Fallbasiertes Schließen in Expertensystemen: Welche Rolle spielen Fälle für wissensbasierte Systeme?*

Klaus–Dieter Althoff, Stefan Weß,
Brigitte Bartsch–Spörl, Dietmar Janetzko,
Frank Maurer, Angi Voß,

Zusammenfassung

Fallbasiertes Schließen ist ein derzeit viel diskutierter Problemlösesansatz. Dieser Beitrag gibt einen Überblick über den aktuellen Stand der Forschung auf diesem Gebiet, insbesondere im Hinblick auf die Entwicklung von Expertensystemen (einen ersten Schritt in diese Richtung stellte bereits der Beitrag von Bartsch-Spörl, [BS87] dar). Dazu stellen wir die dem fallbasierten Schließen zugrundeliegenden Mechanismen vor. Ergänzt wird dies durch den Vergleich mit alternativen Verfahren wie z.B. regelbasiertes, analoges und induktives Schließen sowie eine ausführliche Literaturübersicht.

1 Einführung

Regeln, Klauseln und andere Konstrukte bilden die Basis vieler Formalismen zur Wissensrepräsentation in Expertensystemen. Die Verwendung dieser Formalismen zur Wissensakquisition liegt nahe, da Expertenwissen oft in dieser Form vorzuliegen scheint. Aber Expertenwissen ist mehr (vgl. Dreyfus & Dreyfus [DD87] oder Becker [Bec90]). Auch wenn Erfahrung allein noch keine ausreichende Bedingung für Expertentum darstellt, so ist doch die langjährige Vertrautheit mit einem Inhaltsbereich und die anhand bereits bearbeiteter Probleme gewonnene Erfahrung eine ganz wesentliche Quelle des Expertenwissens. Ein Experte kann durch die Betonung immer anderer vorher unbeachteter Aspekte einen bestimmten Sachverhalt auf unterschiedliche Art und Weise interpretieren. Durch die Erinnerung an einen bereits erlebten Fall aus einer neuen Perspektive kann er/sie damit Lösungen für bisher unbekannte Probleme finden bzw. den Lösungsprozeß erheblich beschleunigen.

Dieses sogenannte episodische Wissen (vgl. [Str89]), das auf dem "Erlebt-Haben" vieler konkreter Situationen beruht, kann nicht ohne Verlust in Regeln oder ähnlichen Konzepten dargestellt werden. Mit Hilfe des fallbasierten Schließens *(Case-Based Reasoning)* soll es nun Expertensystemen ermöglicht werden, frühere Fälle zur Lösung neuer Aufgaben heranzuziehen und aus gemachten Erfahrungen für zukünftige Situationen zu lernen.

Fallbasiertes Schließen ist derzeit ein sehr intensiver Forschungsgegenstand, was sich in einer Vielzahl von Publikationen, Workshops, Projekten bzw. Projektanträgen äußert (vgl. Tab. 1). Inzwischen sind auch schon einige kommerzielle Produkte auf dem Markt (vgl. [Hel91, Sys91]). Die Prägung des Begriffes *Case-Based Reasoning* sowie die Einführung grundlegender Ideen gehen dabei auf Janet L. Kolodner und Roger C. Schank zurück (vgl. [Sch82, Kol83a, Kol83b]).

2 Übersicht

Vereinfachend kann unter fallbasiertem Schließen das Lösen von Problemen anhand von bereits bekannten Fällen verstanden werden. Dazu werden Erfahrungen in Form von Fallbeispielen gesammelt und in das bereits vorhandene Erfahrungswissen eingeordnet. Ein neues Problem wird

*erschienen in: *Künstliche Intelligenz - KI* (4/92), 14-21, FBO-Verlag, Baden-Baden, 1992

Name	Anwendung	Literatur
CYRUS	Episodisches Gedächnis	[Kol80]
MEDIATOR	Schlichtung von Streitfällen	[Sim85]
PLEXUS	Adaptives Planen	[Alt86]
JUDGE	Rechtssprechung	[Bai86]
SWALE	Fallbasierte Erklärungen	[Sch86]
CHEF	Chinesische Küche	[Ham89a]
MBRtalk	Aussprache von Wörtern	[SW86]
JULIA	Zusammenstellung von Menüs	[Kol87]
HYPO	Rechtssprechung	[RA87]
COACH	Football Coach	[Col88]
CASEY	Diagnose von Herzerkrankungen	[Kot88]
JOHNNY	Lesen von Texten	[Sta88]
REFINER	Wissensverfeinerung	[SS88]
PROTOS	Wissensakquisition	[Bar89]
PATDEX	adaptives Diagnosesystem	[AdlOM+89]
PRODIGY	Planung	[VC91]
CREEK	Diagnosesystem	[Aam91]
CcC+	Diagnosesystem	[PG91]
PLAKON	Konfiguration	[GP91]
REMIND	CBR-Shell	[Sys91]
CBR-EXPRESS	CBR-Shell	[Hel91]
CABPLAN	Arbeitsplanung	[PPW92]
CASSYS	Klassifikation/Indentifikation	[MCR92]

Tabelle 1: Auswahl fallbasierter Systeme

dann gelöst, indem die Lösung eines ähnlichen, bereits bekannten Problems komplett bzw. teilweise auf die neue Situation übertragen und entsprechend den aktuellen Anforderungen modifiziert wird.

Aus psychologischer Sicht stellen Fälle dabei Abstraktionen von Ereignissen oder Prozessen dar, die über einen konkreten räumlichen und zeitlichen Bezug zu der Situation verfügen, in der das Problem und seine Bearbeitung früher bereits einmal auftrat (vgl. [Str89]). Fälle umfassen dabei zumindest eine Problembeschreibung, eine Lösung sowie eine Erläuterung für die gefundene Lösung.

Es ist interessant zu wissen, was für Fälle in einem fallbasierten Problemlöser vorkommen und welchen Informationsquellen sie entstammen bzw. in welchem Kontext ein fallbasierter Problemlöser eingesetzt wird (z.B. analytische oder synthetische Problematik). Fälle können für vielfältige Aufgaben verwendet werden, z.B. zum Aufbau [Bar89], zur Verfeinerung [Alt92] oder zur Überprüfung einer Wissensbasis [SS88] bzw. zur Erklärung bestimmter Inferenzen [Sch86].

Es ist somit von Interesse, welche Schlüsse aus Fällen gezogen werden sollen, was aus ihnen gelernt bzw. inwieweit Hintergrundwissen zu ihrer Verarbeitung einbezogen werden kann. Zur Präzisierung der zugrundeliegenden Begriffe bzw. zur Konkretisierung des skizzierten Verfahrens des fallbasierten Schließens muß berücksichtigt werden, daß sich unterschiedliche Fachgebiete und -disziplinen mit diesem Thema auseinandersetzen und (somit) eine entsprechende Vielfalt von Sichtweisen und Begriffen existiert.

Während aus kognitionspsychologischer Sicht unter fallbasiertem Schließen ein Modell für menschliches Problemlösen zu verstehen ist, werden aus der Sicht des maschinellen Lernens auch Folgen von Hypothesen erzeugt, was sich als eine Art Induktion verstehen läßt. Für den Bereich des Knowledge Engineering impliziert fallbasiertes Schließen eine dynamische Sicht der Wissensmodellierung, die die scharfe Trennung zwischen Wissensakquisition und -applikation, wie sie z.B. derzeit in der modellbasierten Wissensakquisition (vgl. [WSB92, BS92, JS92]) existiert, aufhebt. Wichtige Begriffe, die mit fallbasiertem Problemlösen verbunden werden sollten, sind somit Lernfähigkeit, Anpassung sowie Integration von Wissensakquisition und -applikation. Die Automatisierung der

Wissensakquisition und -anpassung stellt dabei den Übergang zum Lernen dar. Letzteres wurde bereits von Morik (im Sinne automatischer Wissensmodellierung) [Mor89] vorgeschlagen.

Grundsätzlich stellt fallbasiertes Schließen ein Inferenzverfahren zur Behandlung inhomogener Lösungsräume dar. Die einzelnen Fälle korrespondieren dabei zu homogenen[1] Teilräumen. Aufgrund der zugrunde liegenden Inferenzmechanismen kann fallbasiertes Schließen als eine Spezialisierung analogen Schließens aufgefaßt werden (vgl. [Bur89, SD90]). Fallbasierte Schlüsse sind damit, ähnlich wie auch induktive Schlußfolgerungen, nicht notwendigerweise korrekt.

Im folgenden werden wir den Begriff *fallbasiertes Lernen* verwenden, wenn wir besonderen Wert auf die zugrunde liegenden Lernaspekte legen. Der Begriff *fallbasiertes Schließen* (fallbasierte Wissensverarbeitung, fallbasiertes Problemlösen) bezeichnet jeweils den Gesamtprozeß. Spezielle Ausprägungen des fallbasierten Schließens sind in der Literatur auch als *Memory-based Reasoning* [SW86], *Instance-based Learning* [AKA91] *Exemplar-based Learning* [Bar89, PBH90] und *Nearest–Neighbor Classification* [Das90, Aha91] bekannt.

Einen allgemeinen Überblick über fallbasiertes Schließen geben Barletta [Bar91b], Kolodner [Kol91], Riesbeck & Schank [RS89] Rissland, Kolodner & Waltz [RKW89], Slade [Sla91], Schult [Sch92] sowie Wess, Paulokat & Althoff [WPA92, AWBSJ92, AW92]. Eine Beschreibung fallbasierten Schließens aus Sicht der Kognitionspsychologie bzw. des Knowledge Engineering geben Strube & Janetzko [SJ90].

3 Prozeßmodell

Für das fallbasierte Schließen wurden in der Literatur eine Reihe unterschiedlicher Prozeßmodelle vorgeschlagen (vgl. z.B. [Ham89a, Sla91, Wol89]). Das Phasenmodell nach Kolodner et al. [KSS85] ist in der Literatur am weitesten verbreitet. Für den weiteren Verlauf wollen wir jedoch die folgende Beschreibung nach Rissland et al. [RKW89] als Basis für die fallbasierte Wissensverarbeitung verwenden. Der Ablauf nach Rissland zeichnet sich gegenüber dem Phasenmodell nach Kolodner durch eine genauere Spezifikation der einzelnen Phasen aus. Insbesondere fehlt im Modell von Kolodner eine explizite Lernphase.

Eingabe: Aktuelle Problembeschreibung P, Fallbasis FB mit Fällen F_i.

Ausgabe: Lösung S für das aktuelle Problem.

Bereitstellung: Ziel des ersten Schrittes ist es, geeignete Fälle aus der Fallbasis zu wählen. Die Auswahl erfolgt meistens über Eigenschaften der aktuellen Situation, welche sich in der Vergangenheit für die Lösung des Problems als relevant erwiesen haben. Das Retrieval soll zunächst die Anzahl der in den folgenden Schritten zu betrachtenden Fälle einschränken (vgl. dazu [GF91]). Die Organisation der Fallbasis, sie kann hierarchisch oder eher flach aufgebaut werden, und die Realisierung der Zugriffsfunktionen führt zu unterschiedlichen Ansätzen: *Memory Organisation Packages*, z.B. in CYRUS, CASEY, *Explanation-Based Indexing* (vgl. [BM88]), *K-D Bäume*, z.B. in REMIND, *Hypercubes* (vgl. [SHK89]), *Listen*, z.B. in MBRTALK, *Graphen,* z.B. in PATDEX oder CABPLAN.

Auswahl: Im zweiten Schritt muß aus der unter Umständen großen Anzahl von gefundenen, potentiell brauchbaren Fällen der für die spezielle Situation am besten geeignete Fall ausgewählt werden. Ein Fall wurde dann gut gewählt, falls die nachfolgenden Adaptionsschritte leicht ausgeführt werden können. In den bekannten Systemen werden dazu eine Reihe von unterschiedlichen Ranking-Mechanismen verwendet: *Explizite Ähnlichkeitsmaße*, z.B. in MBRTALK, PATDEX, CCC+, *Taxonomien*, z.B. in CHEF, *Goal–Driven Similarity Assessment* (vgl. [WJM92]), *Parameter–Gewichtungen*, z.B. in PROTOS, *Kausale Modelle*, z.B. in CASEY, (vgl. auch [AWBSJ92]).

Anpassung und Interpretation: In diesem Schritt erfolgt, falls nötig, die Anpassung der gefundenen Lösung an die aktuelle Situation. Je nach Anwendungsgebiet existieren unterschiedliche Möglichkeiten (vgl. [RS89]): *Parameter–Änderung*, z.B. in JUDGE, HYPO, *Abstraktion*

[1] Homogen in dem Sinne, daß kleine Änderungen der Problembeschreibung kleine Änderungen der jeweiligen Lösung zur Folge haben.

und Spezialisierung, z.B. in SWALE, *Critic-Based Adaptation*, z.B. in CHEF, *Modifikationsregeln*, z.B. in CHEF, *Reinstantiation*, z.B. in MEDIATOR. Eine allgemeingültige Vorgehensweise, wie bei der Anpassung vorgegangen werden muß, kann jedoch nicht angegeben werden. Die in konkreten fallbasierten Systemen angewandten Methoden sind daher im allgemeinen sehr problemspezifisch.

Test und Kritik: In diesem Schritt soll der Output des letzten Schrittes intern, d.h. vom System selbst, nochmals überprüft werden. Dazu eignen sich unterschiedliche Methoden: *Überprüfung auf existierende Gegenbeispiele*, z.B. in CASEY, *Simulation*, z.B. in CHEF, *Überprüfung auf Verletzungen von gegebenen Randbedingungen*, z.B. in JULIA.

Überprüfung der Ergebnisse: In diesem Schritt wird das Ergebnis der Problemlösung bzw. die Entscheidung des Systems in der realen Welt überprüft. Das System erhält dann ein positives oder negatives Feedback vom Benutzer.

Lernphase: In der Lernphase wird das im vorhergehenden Schritt erhaltene Feedback analysiert. Das Ergebnis der Analyse ist eine Änderung des internen Zustandes des Systems. Falls ein Fehler aufgetreten ist, soll eine Wiederholung des Fehlers vermieden werden. Ansonsten wird die richtige Lösung für die Zukunft positiv vermerkt. Je nach gewähltem Ansatz führt das positive oder negative Feedback zu einer Änderung in der internen Speicherstruktur bei symbolischen Ansätzen oder bei statistischen Ansätzen zu einer entsprechenden Verstärkung oder Abschwächung der jeweiligen Parameter: *Änderung der Indexstruktur*, z.B. in CHEF, *Änderung der Gewichtung*, z.B. in MBRTALK, PATDEX/2, *Generalisierung*, z.B. in CREEK, *Löschen von Fallbeispielen* (vgl. [AKA91]).

Aus einer abstrakten Sicht sind die Begriffe *Lösungstransfer* und vor allem *Ähnlichkeit* für fallbasiertes Schließen zentral. Die Ähnlichkeit wird gewöhnlich (oft auch in der Implementierung versteckt) mit Hilfe von Ähnlichkeitsmaßen modelliert. Zur inhaltlichen Bedeutung ist zu sagen, daß es keinen Sinn macht, von einer absoluten Ähnlichkeit zu sprechen. Ähnlichkeit von Problemstellungen besteht stets nur *bezüglich bestimmter Aspekte* und diese sind ganz wesentlich von dem jeweiligen Verwendungszweck, d.h. von der Pragmatik (vgl. [WJM92]) bestimmt. Wird eine Ähnlichkeitsbewertung zum Problemlösen benötigt, so hat man meist eine gute Vorstellung davon, ob die erhaltene Lösung brauchbar ist. Dies läßt einen a posteriori Rückschluß auf die Güte des zur Lösungsfindung verwendeten Ähnlichkeitsbegriffes zu. Diese a posteriori Information ist aber zum Zeitpunkt der Problemstellung, also a priori, nicht direkt erhältlich. Die Suche nach einem geeigneten Ähnlichkeitsbegriff muß also als ein Lernprozeß aufgefaßt werden.

Ähnlichkeitsmaße sind - meist in der Form ihres Duals, des Abstandsmaßes - Gegenstand ausgedehnter Untersuchungen in verschiedenen Disziplinen. Das Maß bestimmt die abstrakte Klasse der Probleme, die gelöst werden können und somit ist die Menge der zugelassenen Maße von entscheidender Bedeutung. Grundsätzlich sind vor allen Dingen Maße aus der Psychologie (z.B. [Tve77]), der Cluster-Analyse (z.B. [Boc74]) sowie aus der Mustererkennung (z.B. [Fu76]) oder auch aus der Biologie (z.B. [Dic45]) wichtig. Verfahren aus der Mustererkennung bzw. des *Nearest–Neighbor Classification* und das erwähnte *Instance-based Learning* treten auch in der einen oder anderen Form im fallbasierten Schließen auf.

4 Einordnung

4.1 Fälle und Beispiele

Es existieren unterschiedliche Auffassungen über die formale Beschreibung von Fällen. Jedes System verwendet meist eine eigene, sehr implementierungsabhängige Repräsentation. Wir wollen hier von der folgenden (sehr allgemeinen) Definition ausgehen, die für unsere Zwecke jedoch ausreichend ist. Formal wollen wir unter einem Fall (mindestens) ein Tripel aus einer Problembeschreibung P, der zugehörigen Lösung L und einer Erläuterung E der Lösung verstehen (vgl. hierzu z.B. [VC89]):

Definition 1 (Fall) *Ein Fall F ist gegeben als ein Tripel (P, E, L) mit einer Problembeschreibung P, einer Erläuterung E und einer Lösung L.*

Häufig wird dabei der Lösungsweg als Erläuterung angegeben. Für die Diagnose kann ein Fall dann z.B. das folgende Aussehen haben: die gegebene Symptomatik beschreibt das aktuelle Problem, die gestellte Diagnose ist die Lösung und die real aufgetretene Folge der Symptomerhebungen, die zum Stellen der Diagnose führten, liefert die (empirische) Erläuterung.

4.2 Fallbasiertes und analoges Schließen

Aufgrund der zugrunde liegenden Inferenzmechanismen kann fallbasiertes Schließen als eine Spezialisierung analogen Schließens aufgefaßt werden (vgl. [Bur89, SD90, Sei89]). Auf die Problematik, daß fallbasiertes bzw. analoges Schließen bislang hauptsächlich von unterschiedlichen Forschergruppen bearbeitet wurden, verwies z.B. Kolodner [Kol91]. Dabei lag der Forschungsschwerpunkt beim fallbasierten Schließen auf der Fallrepräsentation und -bereitstellung, bei der Analogieforschung dagegen auf dem Lösungstransfer (vgl. z.B. die Arbeiten von Genter [Gen83] oder Kedar–Cabelli [KC85]). Dies ist u.a. in unterschiedlichen Grundannahmen begründet, nämlich ob beim Lösungstransfer ein Domänenwechsel stattfindet (analoges Schließen) oder nicht (fallbasiertes Schließen). Beim fallbasierten Schließen wird im allgemeinen versucht, die Probleme einer Domäne durch das Heranziehen von ähnlichen Beispielen aus dieser Domäne zu lösen. Im Gegensatz dazu ist das Ziel des analogen Schließens, die Transformation und Extension existierenden Wissens aus einer bekannten Domäne in eine Domäne mit noch unbekannter Struktur.

4.3 Fälle und Regeln

Mit dem zentralen Begriff des Falls sind nun einige, das fallbasierte Schließen charakterisierende Eigenschaften verbunden. Da Regeln einen sehr gebräuchlichen Wissensrepräsentationsformalismus darstellen, wollen wir im folgenden eine Unterscheidung zwischen Fällen und Regeln treffen (vgl. auch [RS89]), die uns für eine zunehmende Präzisierung der Begriffsbildung im fallbasierten Schließen (aber auch im maschinellen Lernen) von zentraler Bedeutung zu sein scheint. Eine Regel ist für uns in diesem Zusammenhang (z.B. in Sinne der Regelsprache Ops5) ein Paar aus einer Bedingung und einer Aktion. Im Gegensatz zu Fällen handelt es sich bei Regeln normalerweise nicht um episodisches Wissen, d.h. Regeln korrespondieren nicht mit einem konkreten Ereignis, sondern man kann sie sich durch einen Generalisierungsprozeß aus vielen Ereignissen entstanden denken. Fälle und Regeln unterscheiden sich somit grundsätzlich auf der kognitiven Ebene. Dies muß auf der Ebene der Repräsentation bzw. Implementierung natürlich nicht unbedingt gelten. So kann man sich z.B. (im Extremfall) jeden Fall als eine Regel mit dem entsprechenden Kontext als Vorbedingung denken bzw. Regeln können als Fälle interpretiert werden (partieller Match der Vorbedingung). Zur Gegenüberstellung von Fällen und Regeln vgl. Abb. 1.

	Realitätsbezug	**Art der Heuristik**	**Interpretationsmöglichkeiten**
Regeln	abstrakt	explizit	einfach
Fälle	konkret	implizit	mehrfach

Abbildung 1: Fälle und Regeln

Im Unterschied zu Fällen ist die intendierte Verwendung bei Regeln im Normalfall klar (Vorbedingungen). Entsprechend stellen sie eine explizite Repräsentation von (u.U.) heuristischem Wissen dar. Fälle können dagegen auf unterschiedliche Weise interpretiert werden. Die durch sie realisierten Heuristiken sind daher eher impliziter Natur. Aufbauend auf der Unterscheidung zwischen Regeln und Fällen ergibt sich somit in Abhängigkeit vom gewählten Mustervergleich (exakt oder teilweise; *exact matching* versus *partial matching*) die in Abb. 2 vorgestellte Einordnung für fall-, analogie- und regelbasierte Systeme sowie Standarddatenbanksysteme. Letztere können somit gewissermaßen als ein Spezialfall fallbasierter Systeme betrachtet werden.

	Vollständiger Match	**Partieller Match**
Regeln	Regelbasierter Ansatz	Analogiebasierter Ansatz
Fälle	Standarddatenbank	Fallbasierter Ansatz

Abbildung 2: Matching von Fällen und Regeln

4.4 Ähnlichkeit: Repräsentation versus Berechnung

Grundlegend für das fallbasierte Schließen sind neben der Modellierung eines einzelnen Falles die Repräsentation der Fallgesamtheit in einem Fallgedächtnis, dem *case memory*, die Bereitstellung (*retrieval*) von Fällen aus dem Gedächtnis, die zu einem gegebenen Problem ähnlich sind, sowie die Lösung dieses Problems auf der Basis von Analogien zu den bereitgestellten Fällen (*Lösungstransfer*). Zur Realisierung eines partiellen Matches von aktuell gegebener Situation und bekannten Fallbeispielen bieten sich zwei grundsätzliche Vorgehensweisen an:

- *Berechnung* durch ein explizit gegebenes Ähnlichkeitsmaß.
- *Repräsentation* von Ähnlichkeit durch ein Fallgedächtnis.

Beide Ansätze werden zur Zeit sehr kontrovers diskutiert (vgl. z.B. [Por89]). Ein typischer Vertreter des ersten Ansatzes, also der Verwendung eines explizit gegebenen Ähnlichkeitsmaßes, ist der *Memory-Based Reasoning* Ansatz nach Stanfill & Waltz [SW86]. Die Verwendung eines Fallgedächtnisses zur Speicherung und Bereitstellung von Fallbeispielen wird hingegen von R.C. Schank und Janet Kolodner mit ihrem *Dynamic Memory* favorisiert. Die Motivation dieser Ansätze ist im Fall der Verwendung von Ähnlichkeitsmaßen eher mathematischer, im Fall des *Dynamic Memory* dagegen eher kognitionspsychologischer Natur. Dabei ist beiden Ansätzen gemeinsam, daß sie ähnliche Fälle zur weiteren Bearbeitung bereitstellen. Da wir dies für einen zentralen Punkt beim fallbasierten Schließen halten, wollen wir diese Problematik ein wenig genauer betrachten.

	Ähnlichkeit der Fälle	**Bestimmung der Ähnlichkeit**
Fallgedächtnis	explizit Nachbarn sind ähnlich	implizit durch Einfüge– und Bereitstellungsverfahren
Ähnlichkeitsmaß	implizit durch Funktionswert	explizit durch Funktion

Abbildung 3: Repräsentation versus Berechnung

Die sich auf der kognitiven Ebene stellende Aufgabe ist die Bewertung der Ähnlichkeit zwischen Fällen. Das zugrundeliegende Ziel ist dabei die Optimierung des Kosten–Nutzen–Verhältnisses für den betroffenen Problemlöser. Grundsätzlich kann nun entweder die *Beschreibung des Bewertungsvorganges* oder die *Beschreibung der Ähnlichkeit* in den Vordergrund gestellt, d.h. explizit gemacht werden (vgl. Abb. 3). Im Falle des Ähnlichkeitsmaßes wird dabei der Bewertungsvorgang beispielsweise durch eine mathematische Funktion bzw. durch eine Regel- oder Constraintmenge explizit repräsentiert und z.B. durch eine Funktion der jeweiligen Implementierungssprache bzw. durch eine entsprechende Inferenzmaschine operationalisiert. Beim Fallgedächtnis wird dagegen die Ähnlichkeit von Fällen explizit durch Listen, Tabellen oder Fallstrukturen (Graphen) dargestellt.

Die Konsequenz daraus ist nun, daß mit Hilfe des Ähnlichkeitsmaßes die Ähnlichkeit zwischen Fällen lediglich zur Laufzeit *berechnet* werden kann. Entsprechendes gilt für den *Bewertungsvorgang* der Ähnlichkeit im Fallgedächtnis. Er wird hier lediglich implizit realisiert über die

Bereitstellungs- und Einfügealgorithmen der jeweiligen Implementierungssprache. Die Quintessenz ist also, daß es für beide Ansätze zutrifft, daß zum einen wichtige Information explizit dargestellt, zum anderen aber auch lediglich implizit kodiert wird. Der Vergleich beider Ansätze offenbart den in der Informatik gut bekannten sog. *Tradeoff* zwischen Speicherplatz und Rechenzeit. Hinsichtlich der Verständlichkeit und Transparenz derartiger Verfahren läßt sich somit unabhängig von einer Anwendungssituation keine allgemeine Aussage machen. So können auf einem Fallgedächtnis basierende Verfahren Vorteile für Problemklassen mit eingeschränkter Komplexität bieten, in denen sich die Ähnlichkeitsbewertung auf einfache Weise aus der repräsentierten Ähnlichkeit ableiten läßt. Ähnlichkeitsmaße können dagegen aufgrund ihres hohen Abstraktionsgrades auch für komplexe Anwendungen eingesetzt werden. Die Problematik des Findens eines guten Ähnlichkeitsmaßes kann dabei durch adaptive Lernverfahren erheblich vereinfacht werden (vgl. [SW86, AW91, RW91]).

Entsprechend der Gestaltung der Fallbereitstellung ergeben sich unterschiedliche Möglichkeiten zur Beurteilung der Anwendbarkeit eines Falles bzw. zur Übertragung des in ihm enthaltenen Wissens auf das gegebene aktuelle Problem. Realisiert werden kann dies durch die Partitionierung des Wertebereiches eines Ähnlichkeitsmaßes mit Hilfe von Schwellwerten, durch die explizite Repräsentation zusätzlichen Hintergrundwissens (z.B. Constraints) oder eine Kombination von beidem. Die Möglichkeiten zur Beurteilung der Anwendbarkeit von Fällen bzw. der Übertragung des in ihnen enthaltenen Wissens hängen natürlich direkt von der Reichhaltigkeit der gewählten Fallrepräsentation ab. Darüber hinaus ist insbesondere für Plausibilitätsüberprüfungen die Tatsache von Interesse, ob der fallbasierte Problemlöser mit anderen Inferenzmechanismen integriert ist.

4.5 Klassen fallbasierter Systeme

Die Art und Weise, wie Wissen über einen bekannten Fall für ein gegebenes Problem nutzbar gemacht wird, ist ein Klassifikationskriterium für fallbasierte Problemlöser (vgl. Abb. 4). Wir wollen fallbasierte Problemlöser als *fallselektierend, klassifizierend oder fallvergleichend* (classification, interpretive, precedent-based case-based reasoning, case-matching system) bezeichnen, falls sie lediglich Lösungen akzeptieren oder ablehnen, die durch die bereits bekannten Fälle vorgegeben sind (z.B. PATDEX/2 [Wes91]). Entsprechend heißt ein fallbasierter Problemlöser *fallanpassend, transformierend oder modifizierend* (problem solving case-based reasoning, case-adaptation system), wenn er Lösungen finden kann, die über die Menge der durch die bekannten Fälle vorgegebenen Lösungen hinausgeht (z.B. CHEF [Ham89a]).

	Vollständiger Match	**Partieller Match**
Keine Anpassung	Datenbanksysteme	Fallselektierende Systeme
Anpassung der Lösung	?	Fallanpassende Systeme

Abbildung 4: Klassen fallbasierter Systeme

5 Lernen

Fallbasiertes und induktives Lernen haben Gemeinsamkeiten hinsichtlich ihres Ausgangspunktes, der Fälle bzw. Beispiele, sowie hinsichtlich der Tatsache, daß die jeweils generierten Hypothesen unsicher sind. Beide Ansätze können, ebenso wie das erklärungsbasierte Lernen, vorhandenes Hintergrundwissen verwenden, um den Prozeß der Hypothesengenerierung zu verbessern. Erklärungsbasiertes Lernen im engeren Sinne verlangt dabei meist die Vollständigkeit und Korrektheit des Hintergrundwissens. Fälle werden hier lediglich zur Fokussierung des deduktiven Schlußfolgerungsverfahrens verwendet.

Fallbasierte Lernverfahren sind grundsätzlich inkrementell, es gibt allerdings auch eine Reihe von induktiven Lernverfahren, die inkrementell arbeiten. Insbesondere bestehen Parallelen zwischen fallbasierten Lernverfahren und inkrementellen Conceptual-Clustering Ansätzen (vgl. z.B. COBWEB [Fis87] oder UNIMEM [Leb87]).

Während beim fallbasierten Lernen die Fälle unverändert in einem Fallgedächtnis abgelegt werden, vergessen die meisten induktiven Lernverfahren alle Fälle, die zur Bildung einer Hypothese beigetragen haben. Andere Lernverfahren kombinieren das Lernen expliziter Begriffsbeschreibungen mit dem unveränderten Speichern von Fällen (vgl. [SS88]). Darüber hinaus gibt es fallbasierte Ansätze, die ihre impliziten Begriffsbeschreibungen durch gezieltes, sukzessives Löschen von Fällen verbessern (vgl. [AKA91]).

Aus Sicht des maschinellen Lernens ist fallbasiertes Schließen nicht so gut verstanden wie z.B. induktives Lernen. Nach Shavlik und Dietterich [SD90] besteht z.B. Unklarheit, welche Gesamtlernaufgabe durch fallbasiertes Schließen überhaupt behandelt wird. Sie sehen lediglich eine Fokussierung auf bestimmte Mechanismen wie analoges Schließen und Schließen auf der Basis von Fällen. Shavlik und Dietterich begründen dies damit, daß das bisherige Leitmotiv im fallbasierten Schließen eher kognitive Plausibilität war, denn der Wunsch, praktische Systeme zu entwickeln.

Wir sehen in dieser vermeintlichen Schwäche fallbasierter Systeme andererseits auch einen großen Vorteil. Maschinelle Lernsysteme unterscheiden normalerweise streng zwischen Problemlösen und Lernen. Lernen bedeutet dabei die Analyse von Trainingsbeispielen oder Problemlöseerfahrungen, um Funktionen oder Regeln zu extrahieren. Problemlösen heißt dann, die gelernten Funktionen oder Regeln zum Lösen neuer Probleme anzuwenden. Im fallbasierten Schließen heißt Problemlösen dagegen, Trainingsbeispiele direkt zu untersuchen, um neue Probleme dann durch Analogien zu alten Fällen zu lösen. Es ist gerade das Verzahnen von Lernen und Problemlösen, was fallbasiertes Schließen für viele Aufgaben im Bereich der Expertensystementwicklung für reale komplexe Problemstellungen als besonders geeignet erscheinen läßt (ähnliche Ansätze finden sich auch in neueren Arbeiten aus dem Bereich des Maschinellen Lernens).

5.1 Fallbasiertes Lernen

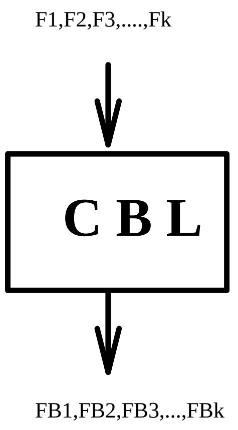
Bekannte Lernverfahren beschäftigen sich überwiegend damit, das Lernen direkt auf die Klassifikation auszurichten. Dazu gehört in der Trainingsphase auch eine geeignete Strukturierung der Fallbasis und eventuell deren Modifikation; trotzdem geht das Gefühl für die Bedeutung des Ähnlichkeitsmaßes meist verloren, insbesondere ist es selten expliziter Gegenstand des Lernverfahrens. Wir wollen hier die Charakterisierung induktiven Lernens aus Jantke und Lange [JL89] aufgreifen und auf fallbasiertes Lernen übertragen (vgl. Abb. 5). Charakteristisch für beide Lernverfahren ist, daß einerseits vorwiegend logische (induktives Lernen), andererseits vorwiegend analytische Begriffsbeschreibungen (fallbasiertes Lernen) gelernt werden (vgl. [Ric92]). Aus einer vorgelegten Folge von Fällen werden inkrementell Hypothesen bestehend aus einer Menge von Fällen (der Fallbasis) generiert. Offen ist dabei, welche Fälle in die Fallbasis aufgenommen werden, inwieweit Fälle wieder aus der Fallbasis ausgeschlossen werden und wie z.B. das Ähnlichkeitsmaß realisiert wird (Bewertungsfunktion versus Fallgedächtnis versus Kombination von beidem). Ansonsten sind die Kriterien aus Jantke und Lange sinngemäß anwendbar.

Wir haben es also aus Sicht dieses abstrakten Szenarios beim fallbasierten Lernen gewissermaßen mit einem Spezialfall induktiven Lernens zu tun. Derartige Betrachtungen können allerdings lediglich diejenigen Aspekte berücksichtigen, die im Rahmen einer formalen Anforderung spezifiziert und damit in ein derartiges Szenario eingebracht werden können.
Sowohl fallbasierte wie auch induktive Lernverfahren erschließen aus lokalen Beobachtungen globale Zusammenhänge. Den Unterschied der einzelnen Ansätze macht aus, welche Zusammenhänge auf welche Art erschlossen werden und was man darüber aussagen möchte. Eine Bearbeitung dieser und ähnlicher Fragen setzt aber zunächst einen gemeinsamen formalen Rahmen voraus, in dem einheitlich fallbasiertes und induktives Lernen behandelt werden können. Dies ist derzeit Gegenstand der Forschung (als einen ersten Schritt in diese Richtung vgl. [Jan92b, Jan92a]).

6 Ausblick

Fallbasiertes Schließen erscheint uns aus den verschiedensten Gründen ein sehr vielversprechendes Forschungsthema zu sein. So sind z.B. Rechtsprechung, Medizin, Wirtschaftswissenschaften oder auch Architektur prominente Bereiche, in denen fallbasiertes Schließen zum Alltag gehört. Aber selbst einer so formalen Disziplin wie der Mathematik ist, beispielsweise bei der Beweisfindung, eine

Eingabe: Folge von Fallbeispielen $F_1, F_2, F_3, ..., F_k$
Ausgabe: Folge von Fallmengen $FB_1, FB_2, FB_3, ..., FB_k$
mit $FB_i \subseteq \{F_1, F_2, F_3, ..., F_i\}$



Ziel: Konvergenz $\exists m \forall n \geq m \, FB_n = FB_{n+1}$

Abbildung 5: Fallbasiertes Lernen

Orientierung an bekannten Fallbeispielen nicht fremd (vgl. Owen [Owe90]). Die Leistungsfähigkeit einer Sammlung von Fallbeispielen sollte jedoch nicht überbewertet werden. Fehler bei der Übertragung von Lösungen auf *neue* Situationen und Fehlklassifikationen lassen sich bei einem fallbasierten Ansatz im allgemeinen nicht vermeiden. Die Kopplung von fallbasierten und anderen Schlußfolgerungsmechanismen scheint in diesem Zusammenhang erfolgversprechend zu sein und ist zur Zeit unter dem Schlagwort *Mixed-Paradigm Reasoning* ein aktuelles Forschungsthema (vgl. Moltke [AMTW91]).

Trotz der nicht in allen Situationen gesicherten Kompetenz der zur Zeit bekannten fallbasierten Ansätze zeigen erste positive Reaktionen von Anwendern und der kommerzielle Erfolg von Cbr-Shells aber, daß ein fallbasierter Ansatz für die Lösung von bestimmten Aufgabenstellungen durchaus sinnvoll sein kann. Der Einsatz von fallbasierten Systemen bietet sich dort an, wo Wissen bereits in der Form vorliegt, in der es das System direkt verarbeiten kann, also in der Form von Fallbeispielen. Gerade Klassifikations- und Diagnoseanwendungen bieten sich hier an. Man denke beispielsweise an alle Formen von technischen Prüfständen. Die anfallenden Prüfprotokolle können direkt als Basis für ein fallbasiertes Diagnosesystem dienen (vgl. Patdex bzw. Ccc+). Interessant ist zudem der Einsatz fallbasierter Problemlöser als entscheidungsunterstützende Systeme [Kol91, Sla91]. Hierbei bieten sich Anwendungen als sog. *HelpDesk Applications*, also Anwendungen zur Serviceunterstützung, an (vgl. [SM91]). Kommerzielle Systeme wie Cbr-Express und ReMind sind auf diesen Markt ausgerichtet.

Mit Blick auf das fallbasierte Schließen ergeben sich eine Vielzahl von Forschungsperspektiven. Dies sind z.B. die Entwicklung fallbasierter Planungssysteme, die Integration von modell- und fallbasierter Wissensakquisition, die Integration induktiver und fallbasierter Lernverfahren sowie das Bereitstellen eines formalen Rahmens zur Analyse fallbasierten Schließens sowie dessen Vergleich mit anderen Lernverfahren.

Ein wichtiges Forschungsziel im Bereich des fallbasierten Schließens ist die Definition von "Rollen", die ein fallbasierter Problemlöser im Rahmen einer Gesamtarchitektur (z.B. einer Expertensystemwerkbank) "spielen" kann. Interessante Fragestellungen sind hier, welche Mechanismen innerhalb einer Architektur sinnvollerweise integriert werden sollten, wie fallbasiertes Schließen mit anderen Schlußfolgerungsmechanismen kombiniert, bzw. wie eine realistische Kosten–Nutzen-Abschätzung hierfür gefunden werden kann. Die Tragfähigkeit fallbasierter Problemlösemechanismen wird sich daran messen lassen müssen, inwieweit es gelingt, Hintergrundwissen in die einzelnen Teilverarbeitungsschritte einzubringen, das einer "vernünftigen" Kosten-Nutzen-Analyse standhält. Erste Ansätze für den Bereich der Diagnose sind die Einbeziehung taxonomischen bzw. kausalen Hintergrundwissens. Durch derartiges Hintergrundwissen werden einfache Anpassungen der Lösungen bzw. Lösungswege bereits bekannter Fälle an neue Gegebenheiten durchführbar (vgl. für PATDEX [PWW92]).

Eine weitere wichtige Forschungsaufgabe ist die Untersuchung von Gemeinsamkeiten und Unterschieden von fallbasierter Diagnose bzw. fallbasierter Planung, insbesondere für bestimmte (noch zu definierende) Problemklassen. Querbezüge sind z.B. gegeben durch die Strategieaufgabe innerhalb des Diagnoseprozesses, durch Klassifikationsteilaufgaben innerhalb des Planungsprozesses sowie durch die Einbeziehung von Modellen des qualitativen Schließens als Hintergrundwissen.

Es wurde bereits in der Einführung dargelegt, daß sich eine ganze Reihe unterschiedlicher Disziplinen aus einer spezifischen Sichtweise mit dem Thema des fallbasiertem Schließens auseinandersetzen. Dies läßt eine Vielzahl interessanter Ergebnisse erhoffen. Zum anderen sehen wir hier die Möglichkeit zur Integration von Ansätzen aus Forschungsgebieten wie Knowledge Engineering/Wissensakquisition, Kognition, Maschinelles Lernen, Wissensrepräsentation, Statistik und Mustererkennung. Grundsätzlich sind wir der Meinung, daß die weitgreifendsten Fortschritte für fallbasiertes Schließen im Rahmen einer interdisziplinären Zusammenarbeit zu erwarten sind, wo sich die Wissenschaftler der beteiligten Gebiete weniger an einem "Input-Output"-Denken orientieren, sondern an interaktiver Zusammenarbeit.

7 Danksagung

Die Idee zu dem vorliegenden Beitrag entstand auf dem GWAI-91 Workshop *„Welche Rolle spielen Fälle für wissenbasierte Systeme?"* der von A. Voß, A. Günter, F. Puppe und G. Strube organisiert wurde. Auf dem Workshop trafen sich zum ersten Mal in Deutschland am Thema des fallbasierten Schließens interessierte WissenschaflerInnen.

Die Autoren danken den Teilnehmern und den Organisatoren des GWAI-91 Fälle-Workshops, sowie Klaus P. Jantke, Katharina Morik, Stefan Wrobel und Michael M. Richter für hilfreiche Anmerkungen zu früheren Versionen dieses Beitrages.

Literatur

[Aam91] Agnar Aamodt. *A Knowledge-Intensive, Integrated Approach to Problem Solving and Sustained Learning.* Dissertation, University of Trondheim, 1991.

[AdlOM+89] Klaus-Dieter Althoff, Alvaro de la Ossa, Frank Maurer, Michael Stadler und Stefan Wess. Adaptive Learning in the Domain of Technical Diagnosis. In P. Bock, F. J. Radermacher und M. M. Richter and, Hrsg, *Proceedings of the FAW Workshop on Adaptive Learning,* FAW-B-89020, Ulm, Germany, Juli 1989. Forschungsinstitut für anwendungsorientierte Wissensverarbeitung (FAW).

[Aha91] David W. Aha. Case-Based Learning Algorithms. In Bareiss [Bar91a]. Washington, D.C., USA, May 8–10, 1991.

[AKA91] David W. Aha, Dennis Kibler und Marc K. Albert. *Instance-Based Learning Algorithms. Machine Learning,* 6:37–66, 1991. March 1991.

[Alt86] R. Alterman. An Adaptive Planner. In *Proceedings of the 5th Annual National Conference on Artificial Intelligence AAAI-86.* AAAI86, Morgan Kaufmann Publishers, 1986. Philadelphia, Pennsylvania, USA.

[Alt92] Klaus-Dieter Althoff. *Eine fallbasierte Lernkomponente als integrierter Bestandteil der* MOLTKE-*Werkbank zur Diagnose technischer Systeme.* Dissertation, Dept. of Computer Science, University of Kaiserslautern, Germany, 1992. (in Vorbereitung).

[AMTW91] Klaus-Dieter Althoff, Frank Maurer, Ralph Traphöner und Stefan Wess. *Die Lernkomponente der* MOLTKE-*Werkbank zur Diagnose technischer Systeme. KI - Künstliche Intelligenz,* 91(1), 1991. January 1991. Also appeared as SEKI-Working Paper SWP-90-08 (SFB).

[AW91] Klaus-Dieter Althoff und Stefan Wess. Case-Based Knowledge Acquisition, Learning and Problem Solving for Diagnostic Real World Tasks. In Smeed [Sme91]. SEKI-Report SR-91-07 (SFB).

[AW92] Klaus-Dieter Althoff und Stefan Wess. Case-Based Reasoning and Expert System Development. In Schmalhofer et al. [SSW92]. In preparation.

[AWBSJ92] K-D. Althoff, S. Wess, B. Bartsch-Spörl und D. Janetzko, Hrsg. *Workshop: Ähnlichkeit von Fällen beim fallbasierten Schließen*, SEKI-Report, Universität Kaiserslautern, SFB 314, 25.-26. Juni, 1992.

[Bai86] W. M. Bain. *Case-Based Reasoning: A Computer Model of Subjective Assesment.* Dissertation, Yale University, New Haven, Connecticut, 1986.

[Bar89] Ray Bareiss. *Exemplar-Based Knowledge Acquisition: A unified Approach to Concept Representation, Classification and Learning.* Academic Press, 1989.

[Bar91a] Ray Bareiss, Hrsg. *Proceedings: Case-Based Reasoning Workshop*, San Mateo, California, 1991. DARPA, Morgan Kaufmann Publishers. Washington, D.C., USA, May 8–10, 1991.

[Bar91b] Ralph Barletta. *An Introduction to Case-Based Reasoning. AI Expert*, 6(8):43–49, 1991. August 1991.

[Bec90] Barbara Becker. *Die Veränderung von (Experten-) Wissen durch den Prozeß der Wissensakquistion.* In *KI - Themenheft Wissensakquisition* [Vos90], Seiten 31–34.

[BM88] R. Barletta und W. Mark. Explanation-Based Indexing of Cases. In Kolodner [Kol88], Seiten 50–61. Clearwater Beach, Florida, USA, May 10–13, 1988.

[Boc74] H. Bock. *Automatische Klassifikation.* Göttingen, 1974.

[BS87] Brigitte Bartsch-Spörl. *Ansätze zur Behandlung von fallorientiertem Erfahrungswissen in Expertensystemen. KI - Künstliche Intelligenz*, 1987.

[BS92] Brigitte Bartsch-Spörl. A simple interpretation model for case-based reasoning. In *Proceedings 2. KADS User Meeting.* Siemens AG, 1992.

[Bur89] M. H. Burstein. Analogy vs. CBR: The Purpose of Mapping. In Hammond [Ham89b], Seiten 133–136. Pensacola Beach, Florida, USA, May 31–June 2, 1989.

[Col88] Gregg Collins. *Plan Creation: Using Strategies as Blueprints.* Dissertation, Yale University, New Haven, Connecticut, 1988.

[Das90] Belur Dasarathy. *Nearest Neighbor Norms: NN Pattern Classification Techniques.* IEEE Computer Society Press, 1990.

[DD87] H. L. Dreyfus und S. E. Dreyfus. *Künstliche Intelligenz - von den Grenzen der Denkmaschinen und dem Wert der Venunft.* Hamburg, 1987.

[Dic45] L. R. Dice. *Measures of the amount of ecologic association between species. Journal of Ecology*, 26:297–302, 1945.

[Fis87] D. Fischer. *Knowledge Acquisition via incremental conceptual clustering. Machine Learning*, 2:139–172, 1987.

[Fu76] K. S. Fu, Hrsg. *Digital Pattern Recognition.* Springer Verlag, 1976.

[Gen83] Dedre Gentner. *Structure-mapping: A theorectical framework for Analogy. Cognitive Science*, 7:155–170, 1983.

[GF91] Dedre Gentner und Kenneth D. Forbus. MAC/FAC: A model of similarity-based retrieval. In *Proceedings of the 13th Annual Conference of the Cognitive Science Society*, Seiten 504–509. Cognitive Science Society, 1991. Chicago, Illinois, USA.

[GP91] Andres Günter und Kai Pfitzner. Fallbasiertes Konstruieren mit Bibliothekslösungen. In R. Cunis, A. Günter und H. Strecker, Hrsg, *Das* PLAKON *Buch*, chapter 9. Springer Verlag, 1991.

[Ham89a] Kristian J. Hammond. *Case-Based Planning: Viewing Planning as a Memory Task.* Academic Press, Boston, Massachusetts, 1989.

[Ham89b] Kristian J. Hammond, Hrsg. *Proceedings: Case-Based Reasoning Workshop*, San Mateo, California, 1989. DARPA, Morgan Kaufmann Publishers. Pensacola Beach, Florida, USA, May 31–June 2, 1989.

[Hel91] Thomas Helton. *The need for Case-Based Reasoning:* CBR EXPRESS. *AI Expert*, 6(10):55–57, 1991. Software Review. October 1991.

[Jan92a] Klaus P. Jantke. Case-Based Learning in Inductive Inference. In *Proc. COLT-92*, 1992.

[Jan92b] Klaus P. Jantke. Case-Based Reasoning and Inductive Inference. Gosler Report 08/92, University of Leipzig, Leipzig, Germany, 1992. February 1992.

[JL89] Klaus P. Jantke und Steffen Lange. Algorithmisches lernen. In J. Grabowski, K. P. Jantke und H. Thiele, Hrsg, *Grundlagen der Künstlichen Intelligenz*, Seiten 246–277. Akademie–Verlag, Berlin, 1989.

[JS92] Dietmar Janetzko und Gerhard Strube. Case-based reasoning and model-based knowledge acquisition. In Schmalhofer et al. [SSW92]. To appear.

[KC85] S. Kedar-Cabelli. Purpose-Directed Analogy. In *Proceedings of the Cognitive Science Conference 1985*. CSC, 1985.

[Kol80] Janet L. Kolodner. *Retrieval and Organizational Strategies in Conceptual Memory*. Dissertation, Yale University, 1980.

[Kol83a] Janet L. Kolodner. *Maintaining Organization in a Dynamic Long-Term Memory*. *Cognitive Science*, 7:243–280, 1983.

[Kol83b] Janet L. Kolodner. *Reconstructive Memory : A Computer Model*. *Cognitive Science*, 7:281–328, 1983.

[Kol87] Janet L. Kolodner. Extending Problem Solver Capabilities through Case-Based Inference. In *Proceedings of the 4th International Workshop on Machine Learning*. ML, Morgan Kaufmann Publishers, 1987. Also in [Kol88].

[Kol88] Janet L. Kolodner, Hrsg. *Proceedings Case-Based Reasoning Workshop*, San Mateo, California, 1988. DARPA, Morgan Kaufmann Publishers. Clearwater Beach, Florida, USA, May 10–13, 1988.

[Kol91] Janet L. Kolodner. *Improving Human Decision Making through Case-Based Decision Aiding*. *AI Magazine*, 91(2):52–68, 1991.

[Kot88] P. Koton. *Using Experience in Learning and Problem Solving*. Dissertation, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1988.

[KSS85] J. L. Kolodner, R. L. Simpson und K. Sycara. A Process Model of Case-Based Reasoning in Problem Solving. In *Proceedings of the 9th International Conference on Artificial Intelligence IJCAI-85*, Seiten 284–290. IJCAI, Morgan Kaufmann Publishers, 1985. Los Angeles, California, USA.

[Leb87] M. Lebowitz. *Experiments with incremental concept formation:* UNIMEM. *Machine Learning*, 2:103–138, 1987.

[MCR92] Michel Manago, Noel Conruyt und Jaques Le Renard. Acquiring descriptive knowledge for classification and identification. In Th. Wetter, K.-D. Althoff, J. Boose, B.R. Gaines, M. Linster und F. Schmalhofer, Hrsg, *Current Developments in Knowledge Acquisition – EKAW-92*, Nummer 599 in Lecture Notes in Artificial Intelligence, Heidelberg and Kaiserslautern, Mai 1992. Springer Verlag.

[Mor89] K. Morik. Sloppy modeling. Seiten 107–134. Springer Verlag, 1989.

[Owe90] Stephen Owen. *Analogy for Automated Reasoning*. Academic Press, 1990.

[PBH90] B. W. Porter, R. Bareiss und R. C. Holte. *Concept Learning and Heuristic Classification in Weak-Theory Domains*. *Artificial Intelligence*, 45, 1990.

[PG91] Frank Puppe und Klaus Goos. Improving Case-Based Classification with Expert Knowledge. In Thomas Christaller, Hrsg, *Proceedings of the German Workshop on Artificial Intelligence 1991*, Berlin, September 1991. Springer-Verlag.

[Por89] B. W. Porter. Similarity Assessment: Computation vs. Representation. In Hammond [Ham89b], Seiten 82–84. Pensacola Beach, Florida, USA, May 31–June 2, 1989.

[PPW92] Jürgen Paulokat, Reinhard Praeger und Stefan Wess. CABPLAN – fallbasierte Arbeitsplanung. In T. Messer und A. Winklhofer, Hrsg, *Beiträge zum 6. Workshop Planen und Konfigurieren*, Nummer 166 in FR-1992-001, Seite 169, Germany, Maerz 1992. Forwiss.

[PWW92] Gerd Pews, Frank Weiler und Stefan Wess. Bestimmung der Ähnlichkeit in der fallbasierten Diagnose mit simultationsfähigen Maschinenmodellen. In Althoff et al. [AWBSJ92].

[RA87] Edwina Rissland und Kevin D. Ashley. HYPO: A Case-Based Reasoning System. In *Proceedings of the 10th International Conference on Artificial Intelligence IJCAI-87*. IJCAI, 1987. Milan, Italy.

[Ric92] Michael M. Richter. Classification and learning of similarity measures. In *Proc. der 16. Jahrestagung der Gesellschaft für Klassifikation e.V.* Springer Verlag, 1992.

[RKW89] Edwina L. Rissland, Janet L. Kolodner und David Waltz. Case-based reasoning *from* darpa: Maschine learning program plan. In Hammond [Ham89b], Seiten 1–13. Pensacola Beach, Florida, USA, May 31–June 2, 1989.

[RS89] C. K. Riesbeck und R. C. Schank. *Inside Case-Base Reasoning*. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1989.

[RW91] Michael M. Richter und Stefan Wess. Similarity, Uncertainty and Case-Based Reasoning in PATDEX. In Robert S. Boyer, Hrsg, *Automated Reasoning*, Essays in Honor of Woody Bledsoe, Seiten 249–265. Kluwer Academic Publishing, 1991. Also appeared as SEKI-Report SR-91-01 (SFB), University of Kaiserslautern.

[Sch82] Roger C. Schank. *Dynamic Memory: A Theory of Learning in Computers and People*. Cambridge University Press, New York, 1982.

[Sch86] Roger C. Schank. *Explanation Patterns: Understanding Mechanically and Creatively*. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1986.

[Sch92] Thomas Schult. *Orientierung am Konkreten - Expertensysteme mit Gedächnis. c't - Magazin für Computer Technik*, Seite 82, April 1992.

[SD90] J. W. Shavlik und T. G. Dietterich, Hrsg. *Readings in Maschine Learning*. Morgan Kaufmann, 1990.

[Sei89] C. M. Seifert. Analogy and Case-Based Reasoning. In Hammond [Ham89b], Seiten 125–129. Pensacola Beach, Florida, USA, May 31–June 2, 1989.

[SHK89] R. H. Stottler, A. L. Henke und J. A. King. Rapid Retrieval Algorithms for Case-Based Reasoning. In *Proceedings of the 11th International Conference on Artificial Intelligence IJCAI-89*, Seiten 233–237. IJCAI, 1989. Detroit, Michigan, USA.

[Sim85] R. L. Simpson. *A Computer Model of Case-Based Reasoning in Problem Solving: An Investigation in the Domain of Dispute Mediation*. PhD thesis GIT-ICS-85/18, School of Information and Computer Science, Georgia Institute of Technology, Atlanta, Georgia, 1985.

[SJ90] Gerhard Strube und Dietmar Janetzko. *Episodisches Wissen und fallbasiertes Schließen: Aufgaben für die Wissensdiagnostik und die Wissenspsychologie. Schweizerische Zeitschrift für Psychologie*, 49(4):211–221, 1990.

[Sla91] Stephen Slade. *Case-Based Reasoning: A Research Paradigm. AI Magazine*, 91(1):42–55, 1991.

[SM91] Evangelos Simoudis und James S. Miller. The Application of CBR to Help Desk Applications. In Bareiss [Bar91a], Seiten 25–36. Washington, D.C., USA, May 8–10, 1991.

[Sme92] Duncan Smeed, Hrsg. *Proceedings of the 5th European Knowledge Acquisition Workshop EKAW'91*. Springer Verlag, 1992.

[SS88] S. Sharma und D. Sleeman. REFINER: A Case-Based Differential Diagnosis Aid for Knowledge Acquisition and Knowledge Refinement. In *Proceedings of the European Working Session on Learning EWSL-88*, Seiten 201–210. EWSL, 1988. University of Aberdeen, Scotland.

[SSW92] Franz Schmalhofer, Gerhard Strube und Thomas Wetter, Hrsg. *Contemporary Knowledge Engineering and Cognition*. Springer-Verlag, Berlin, 1992. To appear.

[Sta88] Craig Stanfill. Learning to Read: A Memory Based Model. In Kolodner [Kol88], Seiten 402–413. Clearwater Beach, Florida, USA, May 10–13, 1988.

[Str89] Gerhard Strube. Episodisches Wissen. Arbeitspapiere der GMD 385, GMD, St. Augustin, Germany, 1989.

[SW86] Craig Stanfill und David Waltz. *Toward Memory-Based Reasoning. Communications of the ACM*, 29(12):1213–1229, 1986.

[Sys91] Cognitive Systems. REMIND - solutions from prior experience. Technical report, Cognitive Systems, Inc., New Haven, CT, 1991.

[Tve77] A. Tversky. *Features of Similarity. Psychological Review*, 84:327–352, 1977.

[VC89] M. M. Veloso und J. G. Carbonell. Learning Analogies by Analogy – The Closed Loop of Memory Organisation and Problem Solving. In Hammond [Ham89b], Seiten 153–158. Pensacola Beach, Florida, USA, May 31–June 2, 1989.

[Vos90] Angi Voss, Hrsg. *KI - Themenheft Wissensakquisition*. Oldenbourg Verlag, München, Juni 1990.

[Wes91] Stefan Wess. PATDEX/2: Ein System zum adaptiven, fallfokussierenden Lernen in technischen Diagnosesituationen. SEKI-Working Paper SWP91/01, Dept. of Computer Science, University of Kaiserslautern, Germany, Januar 1991.

[WJM92] Stefan Wess, Dietmar Janetzko und Erica Melis. Goal-Driven Similarity Assessment. SEKI-Report (SFB), Dept. of Computer Science, University of Kaiserslautern, Germany, 1992. January 1992.

[Wol89] J. Wolstencroft. *Restructuring, Reminding, Repair: What's missing from Models of Analogy. AI Communications*, 2(2):58–71, 1989.

[WPA92] Stefan Wess, Jürgen Paulokat und Klaus-Dieter Althoff. Fallbasiertes Schließen - ein Überblick. Technical report, Fachbereich Informatik, Universität Kaiserslautern, Januar 1992.

[WSB92] B.J. Wielinga, A.T. Schreiber und J.A. Breuker. *KADS: A Modelling Approach to Knowledge Engineering. Knowledge Acquisition Journal (Special Issue on KADS)*, 1992. (to appear).