Fallbasiertes Schließen in der Finanzwelt: Eine echte Alternative zu Neuronalen Netzen?

Thomas P. Reinartz [1] und Wolfgang Wilke [2]

Zusammenfassung

Fallbasiertes Schließen (engl.: Case-based Reasoning) hat in den vergangenen Jahren zunehmende Bedeutung für den praktischen Einsatz in realen Anwendungsbereichen erlangt. In dieser Arbeit stellen wir zunächst die allgemeine Vorgehensweise und die verschiedenen Teilaufgaben des fallbasierten Schließens vor. Anschließend erörtern wir die charakteristischen Eigenschaften eines Anwendungsbereiches, die einen Einsatz des fallbasierten Ansatzes begünstigen, und demonstrieren an der konkreten Aufgabe der Kreditwürdigkeitsprüfung die Realisierung eines fallbasierten Ansatzes in der Finanzwelt. Die abschließende Diskussion zeigt durch einen Vergleich mit verschiedenen Methoden der Klassifikation das große Innovationspotential des fallbasierten Schließens in der Finanzwelt als eine echte Alternative zu bisherigen Methoden.

1 Einleitung

Der Mensch lernt aus Erfahrungen. Mit jeder neuen Erfahrung wächst sein Wissen und die Möglichkeit, sich in Zukunft in ähnlichen Situationen durch Erinnerung an eine entsprechende Erfahrung angemessen zu verhalten. Beispielsweise stellt ein Arzt eine Diagnose häufig aufgrund bereits bekannter Fälle, in denen ein Patient mit vergleichbaren Symptomen seine Praxis besuchte. Dabei wird ein Arzt gegebenenfalls die bekannte Erfahrung geeignet an die neue Situation anpassen, falls die Symptome des früheren Patienten nicht absolut mit der aktuellen Situation übereinstimmen.

Diese Vorgehensweise stellt das Prinzip des *analogen Schließens* dar [Carbonell, 1986]. In Abbildung 1 wird dieses Prinzip illustriert. Zu einer gegebenen aktuellen Problemstellung versucht der Mensch, sich zunächst an eine ähnliche bekannte Problemstellung

[1] Daimler-Benz AG, Forschung und Technik F3S/E, Postfach 2360, 89013 Ulm, e-Mail: reinartz@dbag.ulm.DaimlerBenz.com

[2] Universität Kaiserslautern, Fachbereich Informatik, AG Richter, Zentrum für lernende Systeme und Anwendungen (LSA), Postfach 3049, 67653 Kaiserslautern, e-Mail: wilke@informatik.uni-kl.de

zu erinnern. Anschließend wird die dazu bekannte Lösung auf die aktuelle Situation

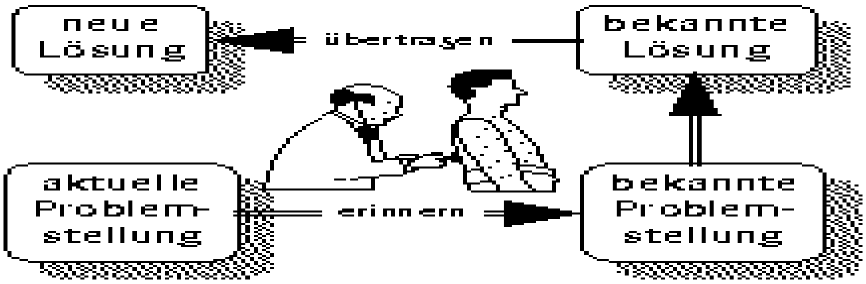

Abbildung 1: Prinzip des analogen Schließens

übertragen. *Fallbasiertes Schließen* (engl.: *Case-based reasoning*) stellt ein Modell dieser menschlichen Schlußweise dar, in der Problemlösen, Verstehen und Lernen in Gedächtnisprozessen integriert wird [Schank, 1982, Kolodner, 1983a, Kolodner, 1993].

Die wesentliche Voraussetzung des fallbasierten Schließens ist eine geeignete Auswahl und Darstellung bereits bekannter Erfahrungen. Jede dieser Erfahrungen wird als *Fall* bezeichnet, die in einer sogenannten *Fallbasis* zusammengefaßt werden. Jeder Fall besteht mindestens aus einer *Problembeschreibung* und einer Darstellung der entsprechenden *Lösung*. Darüberhinaus wird häufig Information zu dem Lösungsweg oder fehlgeschlagenen Lösungsversuchen in einem Fall verwahrt [Veloso, 1992].

Die Lösung einer aktuellen Problemstellung beginnt daraufhin mit einer geeigneten Formulierung der neuen Situation, die einen automatischen Vergleich mit den bereits bekannten Erfahrungen gestattet. Die bereits bekannte Lösung wird anschließend in gegebenenfalls modifizierter Form auf die aktuelle Situation übertragen. Auf diese Weise entsteht ein neuer Fall, der ebenfalls in der Fallbasis abgelegt werden kann, um somit für zukünftige Problemstellungen zur Verfügung zu stehen.

Im folgenden stellen wir zunächst die Technik des fallbasierten Schließens aufgrund eines von Aamodt und Plaza (1994) vorgeschlagenen Prozeßmodells vor. Anschließend diskutieren wir die Einsatzmöglichkeiten des fallbasierten Schließens in der Finanzwelt am Beispiel der Kreditwürdigkeitsprüfung. Schließlich zeigt ein Vergleich zwischen fallbasierten Ansätzen und anderen Methoden - wie beispielsweise *lineare Diskriminanzanalyse* und *Neuronale Netze* - das Innovationspotential des fallbasierten Schließens in der Finanzwelt als Alternative oder Ergänzung zu bisher eingesetzten traditionellen Methoden. Die zusammenfassende Betrachtung der Vor- und Nachteile des fallbasierten

Schließens schließt die Arbeit ab.

2 Fallbasiertes Schließen

Abbildung 2 zeigt das Prozeßmodell [Aamodt and Plaza, 1994] des fallbasierten Schließens. Im Mittelpunkt des fallbasierten Zyklus steht die *Fallbasis*, die eine Menge ausgewählter Fälle enthält. Darüberhinaus kann zusätzliches *Wissen* des entsprechenden Anwendungsbereiches nützlich sein. Solches zusätzliche Wissen dient beispielsweise der Unterstützung der Suche nach einer geeigneten ähnlichen Erfahrung oder kann bei einer gegebenenfalls notwendigen Anpassung einer bekannten Lösung an eine aktuelle Situation verwendet werden. Dieses Wissen ist i.d.R. allgemeinerer Natur als die einzelnen konkreten Erfahrungen einer Fallbasis.

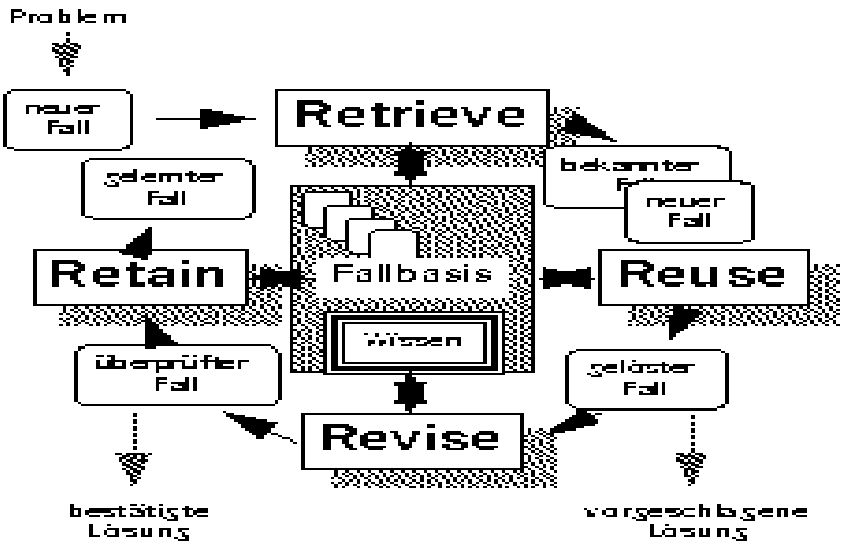

Abbildung 2: Prozeßmodell des fallbasierten Schließens nach [Aamodt and Plaza, 1994]

Der eigentliche Prozeß des fallbasierten Schließens beginnt mit der Beschreibung des aktuellen *Problems*, das einen *neuen Fall* - allerdings ohne Lösung - definiert. Anschließend folgen vier Schritte:

- *Retrieve*: Suche einen bekannten Fall der Fallbasis mit einer hinreichend ähnlichen Problemstellung.
- *Reuse*: Verwende den ähnlichsten solcher Fälle, um eine Lösung für das aktuelle Problem zu erzeugen.

- *Revise*: Überprüfe die vorgeschlagene Lösung des aktuellen Problems. Falls die Lösung fehlerhaft ist, korrigiere diese geeignet.

- *Retain*: Anschließend wird der neue und überprüfte Fall in der Fallbasis abgelegt.

Im folgenden werden wir die Frage nach einer Fallbasis und die einzelnen Prozeßschritte genauer betrachten. Dabei stehen grundsätzlich die Ideen des fallbasierten Schließens im Vordergrund. Zu jedem Schritt werden die entstehenden Teilaufgaben verdeutlicht und Hinweise auf verschiedene existierende Lösungsansätze gegeben. Dabei wird insbesondere weniger Wert auf eine formale Darstellung einzelner Ansätze gelegt (siehe zum Beispiel: [Riesbeck and Schank, 1989, Wess et al., 1992, Kolodner, 1993]).

2.1 Fallbasis

Die wesentliche Voraussetzung des fallbasierten Schließens ist zunächst die Existenz einer hinreichend großen Anzahl konkreter Erfahrungen eines Anwendungsbereiches. Konkrete Erfahrungen liegen häufig lediglich in Form menschlichen Wissens einzelner Experten vor. Demzufolge ist sowohl eine *geeignete* Darstellung der Erfahrungen als auch eine *angemessene* Organisation der Fallbasis notwendig. Die Darstellung eines Falles muß sich für einen automatischen Vergleich zwischen einer bekannten Erfahrung und einer aktuellen Problemstellung sowie für eine gegebenenfalls notwendige Anpassung als nützlich erweisen. Die Organisation der Fallbasis sollte eine möglichst schnelle Suche nach hinreichend ähnlichen bekannten Erfahrungen gestatten.

Um eine effiziente Suche zusätzlich zu unterstützen, muß bereits die Auswahl der Erfahrungen, die in der Fallbasis abgelegt werden, sehr gewissenhaft erfolgen. Einerseits soll eine gewisse Kompetenz des fallbasierten Systems erreicht werden. Andererseits müssen redundante oder wenig neue Information beinhaltende Erfahrungen vermieden werden. Schließlich stellt sich in manchen Anwendungsbereichen sogar prinzipiell die Frage, welches Expertenwissen einen Fall im Sinne des fallbasierten Schließens ergibt.

2.2 Retrieve

Der erste Schritt des Prozeßmodells dient der Suche nach einem geeigneten Kandidaten für die Wiederverwendung in der aktuellen Situation. Ein Kandidat ist insbesondere dann geeignet, wenn der Aufwand für eine Wiederverwendung möglichst gering ist. Dabei lassen sich prinzipiell zwei Strategien unterscheiden:

- *Suche*

Durch geschickte Organisation der Fallbasis - beispielsweise durch Vergabe einer Indexstruktur [Kolodner, 1983b] - wird die explizite Suche nach einem geeigneten Kandidaten für die Wiederverwendung unterstützt. Fälle mit ähnlichen Problemstellungen werden nahe bei einander in der Fallbasis abgespeichert. Dadurch ist ein Vergleich der beiden Problemstellungen nicht unbedingt erforderlich. Ein typisches Beispiel für diese Vorgehensweise sind die sogenannten *kd-Bäume* [Wess et al., 1993, Wess et al., 1994, Wess, 1995]. Mit Hilfe dieser speziellen Datenstruktur wird die Fallbasis in Gruppen ähnlicher Fälle unterteilt. Die Suche in einem solchen Baum wird dann durch die Ausprägungen einzelner Merkmale geleitet.

- *Berechnung*

  Die zweite Alternative vergleicht explizit die aktuelle Problemstellung mit den bekannten Erfahrungen, indem die Eignung eines Kandidaten für die Wiederverwendung berechnet wird. Dazu verwendet das fallbasierte Schließen beispielsweise ein Ähnlichkeits- oder Distanzmaß, das den Aufwand einer Wiederverwendung quantitativ repräsentiert. Der Fall der Fallbasis mit maximaler Ähnlichkeit bzw. minimaler Distanz zu der aktuellen Situation eignet sich demzufolge am besten für eine Wiederverwendung (vgl. Abschnitt 3.2). Solche Maße können sowohl syntaktische als auch semantische Aspekte berücksichtigen.

Beide Möglichkeiten können auch kombiniert eingesetzt werden. Beispielsweise kann durch eine geschickte Organisation der Fallbasis zunächst die Menge der in Betracht zu ziehenden Fälle eingeschränkt werden. Anschließend kann aus dieser ausgewählten Menge geeigneter Kandidaten der beste Fall ausgewählt werden (siehe etwa [Goos, 1995]).

2.3 Reuse

Der nächste Schritt des fallbasierten Schließens besteht nun in der Übertragung der im ersten Schritt gefundenen bekannten Lösung auf die aktuelle Situation. Auch hier unterscheiden wir zwei unterschiedliche Prinzipien:

- *Kopie*

  In der einfachsten Form bedeutet die Übertragung eine Kopie der bekannten Lösung. Falls die bekannte und die aktuelle Problemstellung exakt übereinstimmen, ist diese Methode geeignet. Allerdings sind solche lediglich fallvergleichenden Ansätze in ihrer Kompetenz sehr eingeschränkt. Ein solcher Ansatz kann nur dann eine Lösung für eine neue Problemstellung erzeugen, wenn genau diese Problemstellung bereits erfolgreich bearbeitet wurde. Dies ist leider in vielen Anwendungsbereichen nicht zu erwarten.

- *Anpassung*

  Die zweite Alternative erlaubt eine Anpassung der bekannten Erfahrung an die neue Situation. Ein Teil der bekannten Lösung wird beispielsweise geeignet modifiziert. Dabei verwenden einige Ansätze nicht nur die Lösung einer Problemstellung selbst, sondern auch Informationen aus dem gesamten Problemlösungsprozeß. Beispielsweise versucht ein Ansatz, eine Transformationsabbildung zwischen der bekannten und der neuen Problemstellung zu generieren und mit Hilfe dieser Abbildung angewendet auf die bekannte Lösung die neue Lösung zu erzeugen [Carbonell, 1986].

2.4 Revise

Die auf diese Weise erzeugte Problemlösung für die aktuelle Situation ist nicht notwendigerweise korrekt. Aus diesem Grund wird die vorgeschlagene Lösung im dritten Schritt des fallbasierten Zyklus überprüft und gegebenenfalls korrigiert:

- *Evaluation*

  Zunächst wird die vorgeschlagene Lösung getestet. Dies kann beispielsweise durch eine wirklich durchgeführte oder simulierte Anwendung der Lösung in dem jeweiligen Bereich erfolgen. Gegebenenfalls kann an Stelle eines Tests auch lediglich die Beurteilung der Güte einer Lösung durch einen Experten des Anwendungsbereiches treten.

- *Korrektur*

  Je nach Ausgang der Evaluation muß eine Korrektur der vorgeschlagenen Lösung erfolgen. Dabei wird aufgrund der bei der Evaluation festgestellten Fehler der Lösung eine geeignete Modifikation vorgenommen, so daß schließlich eine korrekte Lösung entsteht. Eine Möglichkeit der Modifikation beginnt beispielsweise mit der Erklärung der beobachteten Fehler. Diese Erklärungen werden dann als Hinweis auf die Art der Korrektur betrachtet [Koton, 1988].

2.5 Retain

Im letzten Schritt des fallbasierten Prozeßmodells wird schließlich die neu erzeugte Erfahrung für zukünftige Problemlösungen zur Verfügung gestellt. Dabei muß zunächst entschieden werden, ob der neue Fall insgesamt oder gegebenenfalls nur Teile des neuen Falles neue nützliche Information enthält. Anschließend muß eine geeignete Darstellung der neuen Information erzeugt und diese in die Fallbasis integriert werden. Auch hier unterscheiden wir zwei Aspekte:

- *Wissensakquisition*

  Einerseits liefert die Verwahrung neuer Erfahrungen einen Beitrag zu der Aufgabe der Wissensakquisition. Neue Fälle stellen neues problemspezifisches Wissen dar, das durch Aufnahme des neuen Falles in die Fallbasis auf natürliche Weise erhoben werden kann. Im Prinzip kann auf diese Weise auch der Aufbau einer Fallbasis erfolgen.

- *Lernen*

  Andererseits stellt die Hinzunahme einer neuen Erfahrung einen Lernprozeß dar. Dabei kann zwischen vollständig neu gelernter Information und der Anpassung existierenden Wissens unterschieden werden. Wenn ein neuer Fall beispielsweise nur geringe Information gegenüber einer bereits existierenden Erfahrung beinhaltet, so kann dieser gänzlich verworfen oder der bekannte Fall geeignet ergänzt werden. Dazu kann auch die Generalisierung einzelner vergleichbarer Erfahrungen zu einem allgemeineren Lösungsprinzip beitragen [Michalski et al., 1986].

3 Fallbasiertes Schließen in der Finanzwelt

Nachdem wir im zweiten Kapitel die einzelnen Schritte und Teilaufgaben des fallbasierten Schließens im allgemeinen vorgestellt haben, werden wir nun auf die charakteristischen Eigenschaften einiger konkreter Anwendungsfelder eingehen und am Beispiel der Kreditwürdigkeitsprüfung (engl.: *Credit Scoring*) zeigen, wie eine fallbasierte Problemlösung gestaltet werden kann.

3.1 Anwendungsbereiche des fallbasierten Schließens

Die Anwendungsgebiete des fallbasierten Schließens sind in den letzten Jahren sehr vielfältig geworden und der Einsatz kommerzieller fallbasierter Systeme wächst zunehmend (vgl. [Althoff et al., 1995]). Abbildung 3 zeigt eine mögliche Einteilung einer Auswahl verschiedener Anwendungsgebiete. Zunächst können wir die Anwendungsbereiche des fallbasierten Schließens in Abhängigkeit des Anpassungsaufwandes bei der Übertragung einer bekannten Lösung auf eine neue Situation in zwei verschiedene Klassen einteilen. Anwendungen mit einem geringen Anpassungsaufwand werden als *Klassifikationsaufgaben* bezeichnet und Anwendungen mit einem hohen Anpassungsaufwand als *synthetische Aufgaben* [Althoff et al., 1995]. Je nach Art der zu lösenden Problemstellung werden diese allgemeinen Klassen weiter unterschieden:

- *Planung*:

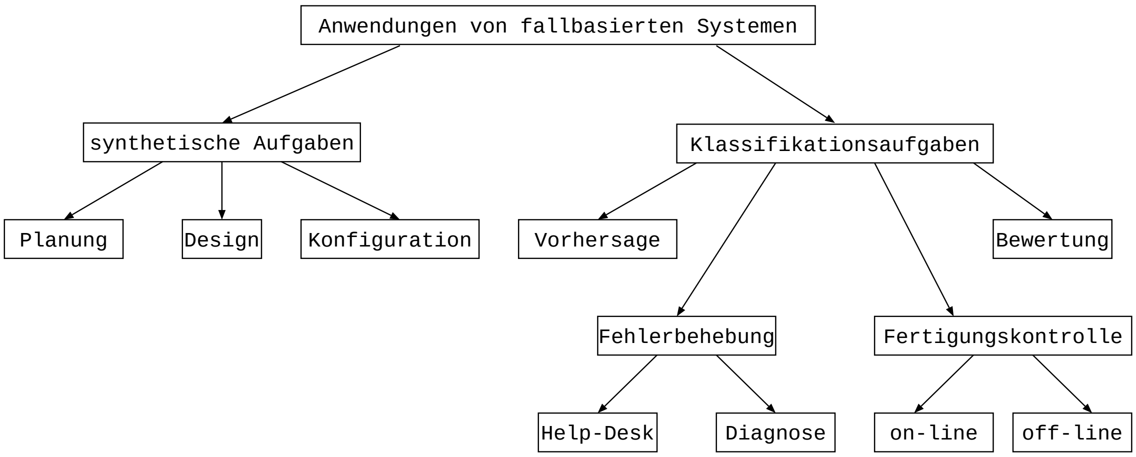

Abbildung 3: Mögliche Einteilung einer Auswahl verschiedener Anwendungsgebiete des fallbasierten Schließens

Zu einem gegebenen Anfangszustand und einem zu erreichenden Zielzustand wird bei der Planung eine Folge von Operatoren - kurz: ein Plan - erzeugt, der den Anfangszustand in den Zielzustand überführt. Als Fälle dienen hier bereits gelöste Planungsprobleme, die aus einem Anfangszustand, einem Zielzustand und dem entsprechenden Plan bestehen. Ein neues Planungsproblem beinhaltet lediglich einen Anfangszustand und einen Zielzustand. Ein fallbasierter Planer sucht nun einen oder mehrere ähnliche bereits gelöste Fälle und konstruiert den neuen Plan mit Hilfe bereits bekannter Pläne. Solche fallbasierten Systeme werden zum Beispiel bei der Fertigung rotationssymmetrischer Drehteile auf einer CNC-Maschine eingesetzt. Prototypische Beispiele für solche fallbasierten Planer sind CAPLAN/CBC [Muñoz et al., 1994, Paulokat and Wess, 1994] oder PARIS [Bergmann and Wilke, 1995].

- *Design / Konfiguration*:

  Die Anwendungsbereiche des Designs und der Konfiguration sind prinzipiell gleich. In beiden Bereichen wird eine Kombination einzelner Komponenten gesucht, die ein technisches System ergeben, das möglichst genau den Anforderungen verschiedener Benutzer genügt. Ein Beispiel ist die Konfiguration von Rechnern. Die Fälle der Fallbasis sind durch bereits zusammengestellte Rechner gegeben. Wenn neue Anforderungen eines Kunden gestellt werden, sucht das fallbasierte System nach einem Rechner der Fallbasis, der bereits möglichst viele der Anforderungen erfüllt. Einzelkomponenten können gegebenenfalls an die neue Situation angepaßt werden. Beispielsysteme zur fallbasierten Konfiguration werden in [Vietze, 1993, Steuernagel, 1989, Pfitzner, 1993] dargestellt.

Eine zusätzliche Schwierigkeit des Designs besteht darin, daß unter Umständen neue Komponenten benutzt werden, die bis dahin dem System noch nicht bekannt waren. Dabei ist i.d.R. weiteres allgemeines Anwendungswissen erforderlich [Brigitte Bartsch-Spörl, 1995].

- *Vorhersage*:

  Bei der Vorhersage wird auf das zukünftige Verhalten eines bestimmten Zielmerkmals aufgrund des Verhaltens dieses und gegebenenfalls weiterer Merkmale in der Vergangenheit geschlossen. Ein Beispiel ist die Vorhersage von Wechsel- oder Aktienkursen. Dabei werden verschiedene Kurswerte mit Parametern, die den Kurswert beeinflussen, als Fälle abgelegt. Bei der Vorhersage einer Kursentwicklung wird dann ein Fall in der Fallbasis gesucht, bei dem die Parameter in der Vergangenheit ähnliche Werte angenommen haben. Der bekannte Kurswertverlauf gilt dann als Lösung der neuen Situation.

Auf die Anwendungsbereiche der *Fehlerbehandlung* und der *Fertigungskontrolle* gehen wir an dieser Stelle nicht weiter ein. Auf den Bereich der *Bewertung* kommen wir im folgenden Abschnitt zurück.

Zusammenfassend nennen wir die folgenden charakteristischen Eigenschaften eines Anwendungsbereiches, die einen Einsatz eines fallbasierten Ansatzes begünstigen:

- Historische Daten (und damit ausreichend Erfahrungen) existieren bereits.
- Experten des Anwendungsbereiches arbeiten mit konkreten Erfahrungen. Allgemeine Zusammenhänge - wie etwa Regeln - existieren nicht, spielen lediglich eine untergeordnete Rolle oder enthalten zahlreiche Ausnahmen.
- Konkrete Erfahrungen sind genauso wertvoll wie theoretisches allgemeines Wissen.
- Die Problemstellungen des Anwendungsbereiches sind nicht vollständig verstanden. Dies ist beispielsweise der Fall, falls kein vollständiges Wissen über die Anwendung vorliegt oder solches Wissen bekannt, aber nicht algorithmisch formulierbar ist.

Darüberhinaus muß der Anwendungsbereich einer gewissen Monotoniebedingung genügen: Die Ähnlichkeit der Problemstellungen überträgt sich auf die Ähnlichkeit der Lösungen. Falls dies nicht vorausgesetzt werden kann, ist die Lösung eines fallbasierten Ansatzes für ein neues Problem entweder falsch oder der Aufwand für die Anpassung der Lösung ist unverhältnismäßig hoch und entspricht nahezu der Konstruktion einer neuen Lösung ohne Erfahrungswissen[Nebel and Koehler, 1993].

3.2 Kreditwürdigkeitsprüfung

Die Kreditwürdigkeitsprüfung ist eine typische Klassifikationsaufgabe. Aufgrund verschiedener Merkmale eines potentiellen Kreditnehmers - wie beispielsweise das Zahlungsverhalten - erfolgt eine Bewertung des Kunden hinsichtlich seiner Kreditwürdigkeit. Der Kunde muß demzufolge einer der beiden Klassen *kreditwürdig* oder *nicht kreditwürdig* zugeordnet werden. In diesem Sinne ergibt sich eine *Bewertung* des potentiellen Kreditnehmers, so daß hier speziell von einer Aufgabe aus dem Anwendungsbereich der *Bewertung* ausgegangen wird.

Wir betrachten hier die Situation, daß der potentielle Kunde ein Wirtschaftsunternehmen ist. Die beiden möglichen Klassen heißen in diesem Bereich *insolvent* oder *nicht insolvent*. Dabei unterscheiden wir zwei prinzipielle Ansätze:

- *fundamentaler Ansatz*:

  Bei dem fundamentalen Ansatz wird versucht, über die Abschätzung aller relevanten außerbetrieblichen Einflußfaktoren (z.B. Branchensituation, Konkurrenzsituation, Marktentwickung) und innerbetrieblichen Einflußfaktoren (z.B. Produktstruktur, Ertragssituation, Managment) ein Bild über die Unternehmensentwicklung zu erhalten. Aufgrund dieses resultierenden Unternehmensprofils wird anschließend die Kreditwürdigkeit bewertet.

- *technischer Ansatz*:

  Dagegen geht der technische Ansatz von einer Jahresabschlußanalyse aus und gelangt über Kennzahlenvergleiche zu Aussagen über die Insolvenzwahrscheinlichkeit und damit zu einer Bewertung der Kreditwürdigkeit. Die technischen Ansätze basieren i.d.R. auf Methoden, die mit Hilfe einer Trainingmenge bereits bekannter Fälle die Kombinationen von Kennzahlen finden, die solvente und insolvente Unternehmen möglichst scharf bei bestimmten Werten trennen.

Wir betrachten im folgenden das fallbasierte Paradigma als eine Form des technischen Ansatzes und vergleichen diese Methode mit verschiedenen weiteren Möglichkeiten, den technischen Ansatz zu verfolgen.

Fallbasis:

Zunächst muß die Voraussetzung des fallbasierten Schließens geschaffen werden. Geeignete Erfahrungen aus der Vergangenheit werden ausgewählt, in eine Falldarstellung transformiert und schließlich in einer Fallbasis abgelegt.

In dieser Anwendung eignet sich beispielsweise eine Attribut-Wert-basierte Fallrepräsentation. Jeder Fall wird mit Hilfe der konkreten Ausprägungen verschiedener relevanter Merkmale beschrieben.

Dazu erfolgt zunächst die Auswahl der wichtigen bzw. relevanten Merkmale. Für den hier betrachteten Bereich sind beispielsweise die folgenden Attribute relevant:

- absolute Werte aus der Bilanz oder dem Jahresabschluß
- Kennzahlen über diesen Werten
- bewertete Kennzahlen
- externe Faktoren (wie sie auch beim fundamentalen Ansatz verwendet werden)

Die Auswahl relevanter Attribute kann auch inkrementell erfolgen. Zunächst werden alle zur Verfügung stehenden Attribute zusammen mit der Klassifikation - hier: solvent oder insolvent - in die Fallrepräsentation aufgenommen. Anschließend wird die Klassifikationsgüte getestet und sukzessive Attribute aus der Repräsentation entfernt, die für ein schlechtes Klassifikationsverhalten sorgen. In umgekehrter Weise kann auch mit wenigen Attributen begonnen werden, um anschließend die Attribute sukzessive hinzuzunehmen, die sich darüberhinaus als relevant erweisen bzw. das Klassifikationsverhalten verbessern. Eine genaue Beschreibung dieser beiden Verfahren findet sich beispielsweise in [Weiss and Kulikowski, 1991].

Retrieve:

Für die nun folgende Suche nach einem geeigneten Wiederverwendungskandidaten für die Bewertung eines neuen Unternehmens nehmen wir an, daß lediglich wenig konkrete Erfahrungen vorliegen, so daß wir uns eine lineare ungeordnete Anordnung der Fälle in der Fallbasis erlauben können. Desweiteren entscheiden wir uns, für die Beurteilung des Aufwandes einer Übertragung einer bereits bekannten Lösung auf eine aktuelle Situation ein Ähnlichkeitsmaß zu verwenden. In dieser Anwendung beurteilt dieses Maß treffender, wie gut die Klassifikation eines neuen Kunden aufgrund bereits bewerteter Unternehmen ist. Je grösser die Ähnlichkeit zwischen einer bekannten und einer neuen Situation ist, desto höher sei die Wahrscheinlichkeit, daß sich das Verhalten des früheren Kunden auch bei dem neuen Kunden zeigen wird.

Zunächst definieren wir nun ein *lokales* Ähnlichkeitsmaß, das zwei Fälle lediglich bezüglich eines Merkmals vergleicht. Anschließend müssen diese lokalen Ähnlichkeiten zu einer Gesamtähnlichkeit akkumuliert werden.

Ein Ähnlichkeitsmaß sei ferner eine Abbildung μ des kartesischen Produktes der Menge A der betrachteten Ausprägungen auf das Intervall [0;1]:

$$\mu : A \times A \rightarrow [0, 1]. \qquad (1)$$

Zusätzlich fordern wir sinnvollerweise, daß jeder Kunde zu sich selbst die maximale Ähnlichkeit besitzt und sich bei der Berechnung die Richtung des Vergleichs nicht auf das Maß der Ähnlichkeit auswirkt:

$\forall x, y \in A$ gilt

$$\mu(x, x) = 1 \tag{2}$$

$$\mu(x, y) = \mu(y, x). \tag{3}$$

Ein Beispiel für ein solches Maß auf geordneten Werten ist:

$$\mu(x, y) := 1 - \frac{\mid x - y \mid}{max}, \tag{4}$$

wobei max die Anzahl der Wertausprägungen darstellt und $\mid x - y \mid$ der Abstand der beiden Werte bezuglich der zugrunde liegenden Ordnung ist. An dieser Stelle sind jedoch auch beliebige andere Maße denkbar, die obige Anforderungen (1) bis (3) erfüllen (vgl. [Bock, 1974]).

Um die unterschiedliche Relevanz der verschiedenen Merkmale zu berücksichtigen, führen wir zusätzlich *Gewichte* ein, die ein Maß für die Bedeutung eines Merkmals für die Klassifikation darstellen. Solche Gewichte können beispielsweise durch einen Experten des Anwendungsfeldes festgelegt oder inkrementell durch ein Lernverfahren bestimmt werden (siehe [Wess, 1995]).

Sei also jetzt für jedes Attribut a_i $(i = 1, \ldots, n)$ der Falldarstellung ein lokales Ähnlichkeitsmaß μ_i in obigem Sinne definiert und jedem Attribut ein Gewicht w_i zugeordnet, so daß gilt:

$$\sum_{i=1}^{n} w_i = 1. \tag{5}$$

Die Ähnlichkeit eines Anfragefalles und eines Falles der Fallbasis berechnet sich dann aus der gewichteten Summe der lokalen Attributähnlichkeiten. Sei also $aq = (aq_1, \ldots, aq_n)$ die Falldarstellung des Anfragefalles und $ac = (ac_1, \ldots, ac_n)$ die des Falles aus der Fallbasis. Dann berechnet sich die Gesamtähnlichkeit zu:

$$\mu_F(aq, ac) := \sum_{i=1}^{n} w_i * \mu_i(aq_i, ac_i). \tag{6}$$

Aufgrund eines solchen Maßes können wir nun die Ähnlichkeit zwischen jedem Fall der Fallbasis und einer neuen Situationsbeschreibung berechnen. Ein fallbasiertes System liefert dann eine Menge der ähnlichsten Fälle zu einem Anfragefall zurück. Dabei können nach der Definition der Gesamtähnlichkeit verschiedene Fälle die gleiche Ähnlichkeit zu einem Anfragefall aufweisen.

Reuse:

Die Anpassung einer bereits existierenden Erfahrung besteht bei der Kreditwürdigkeitsprüfung lediglich aus einer Übernahme der entsprechenden Klassifikation bzw. Bewertung. Dazu können entweder nur die ähnlichsten Fälle berücksichtigt oder auch weniger ähnliche Fälle betrachtet werden. Desweiteren kann die Ähnlichkeit der einzelnen Fälle zueinander mit in diese Entscheidung eingehen. Beispielsweise ist die Ähnlichkeit eine Maßzahl, mit welcher *Sicherheit* eine Bewertung akzeptiert wird.

Revise:

Eine Überprüfung der Lösung kann in dem hier betrachteten Anwendungsbereich im Prinzip erst im Nachhinein erfolgen. Die Klassifikation eines Kunden erweist sich erst nach Betrachtung des tatsächlichen Verhaltens als richtig oder falsch. Darüberhinaus ist nicht mehr feststellbar, ob ein abgelehnter Kunde nicht vielleicht ein guter Kunde gewesen wäre.

Es gibt jedoch auch Möglichkeiten, die Klassifikationsgüte des fallbasierten Ansatzes mit den bereits bekannten Fällen zu beurteilen. Dazu kann beispielsweise jeder einzelne Fall aus der Fallbasis entfernt und als Anfrage an das System gestellt werden. Auf diese Weise ist die Klassifikation des Anfragefalles bekannt, und die erstellte Bewertung kann auf Korrektheit überprüft werden [Weiss and Kulikowski, 1991, Wess, 1991b, Wess, 1991a, Wess, 1991c].

Eine zweite Möglichkeit besteht darin, die zur Verfügung stehenden Fälle in eine Trainings- und eine Testmenge zu unterteilen. Die Trainingsmenge dient dazu, einen fallbasierten Klassifikator zu bilden, und die Testmenge wird anschließend verwendet, um konkrete Anfragen zu stellen. Auch hier läßt sich die Klassifikationsgüte des Systems beurteilen, da die Klassifikationen aller Fälle bekannt sind. Um mit diesem Verfahren jedoch einen statistisch relevanten Test durchzuführen, sind zur Beurteilung eines Systems mehrere Durchläufe mit verschiedenen Test- und Trainingsmengen notwendig. Auf die Auswahl dieser Mengen und auf die Beurteilung der Klassifikationsgüte wollen wir hier nicht näher eingehen (siehe z.B. [Weiss and Kulikowski, 1991]).

Retain:

Im vierten Schritt des fallbasierten Prozeßmodells stellt sich nun die Frage, welche neue Erfahrung für die Lösung zukünftiger Probleme in der Fallbasis verwahrt wird. Dieselbe Frage stellt sich ebenfalls bereits bei dem Aufbau der Fallbasis an sich.

Bei einer solchen Entscheidung spielt einerseits die Repräsentativität eines Falles eine Rolle. Ein neuer Fall sollte nur dann aufgenommen werden, wenn dieser tatsächlich neue Information bietet und die Verhältnisse des Anwendungsbereiches geeignet widerspiegelt. Darüberhinaus müssen andererseits Fälle, die für Fehlklassifikationen verantwortlich sind korrigiert oder entfernt werden. Hierfür existieren jedoch Algorithmen, die aufgrund der einzelnen Anfragen an das System während der Testphase solche Fälle extrahieren [Aha, 1989].

3.3 CreditCBR

Im Auftrag einer Großbank wurde an der *Universität Kaiserslautern* in Zusammenarbeit mit dem *Deutschen Forschungszentrum für Künstliche Intelligenz* ein Vergleich verschiedener Verfahren zur Klassifikation von Kreditnehmern an realen Kundendaten durchgeführt. Für die Entwicklung eines Prototyps und dessen Evaluation wurden uns ca. 700 Fälle jeweils bestehend aus 330 Attributen zur Verfügung gestellt. Um diesen Prototyp zu realisieren, wurde das fallbasierte System INRECA eingesetzt.[3] INRECA vereint die Vorzüge des fallbasierten Schließens und induktiver Techniken. Auf diese Weise konnten die Probleme bei der Modellierung eines Anwendungsbereiches, die im vorherigen Abschnitt geschildert wurden, weitgehend gelöst werden.

Nachdem die Fälle mit den vorgegebenen Attributen in das System eingegeben waren, wurden mit der induktiven Komponente automatisch die Gewichte für die Relevanz der einzelnen Attribute gelernt, so daß wichtige und unwichtige Attribute selbständig unterschieden wurden. Das Problem der Attributauswahl wurde somit automatisch gelöst.

Das Verfahren selbst basiert auf einem Clusteransatz mit einer anschließenden Lernphase, die versucht die Interclusterentropie zu erhöhen und die Intraclusterentropie zu erniedrigen. Als Resultat erhält man eine Menge von Gewichtsvektoren, die dann durch das fallbasierte System für die Ähnlichkeitsberechnung verwendet werden. Eine genaue Beschreibung des Verfahrens befindet sich beispielsweise in [Derwand, 1994]. Entsprechend erfolgte auch die Auswahl der Fälle inkrementell.

Im Vergleich zu anderen Verfahren - wie beispielsweise lineare Diskriminanzanalyse, Induktionsbäume und Neuronale Netze - wurden mit INRECA in dieser Studie die besten Klassifikationsgüten erzielt. Daraus läßt sich jedoch nicht die prinzipielle Überlegenheit des fallbasierten Ansatzes schließen. Diese Studie zeigte dennoch, daß in einem festen kurzen Zeitraum eine erfolgreiche Anwendung mit dem fallbasierten System INRECA realisierbar ist.[4]

[3] INRECA wurde im Rahmen des ESPRIT Projektes P6322 entwickelt. Partner des Projektes sind neben der Arbeitsgruppe „Künstliche Intelligenz - Expertensysteme“ an der Universität Kaiserslautern die Firmen AcknoSoft (Frankreich), tecInno (Deutschland) und Irish Medical Systems (Irland).

[4] Aus Geheimhaltungsgründen können wir an dieser Stelle leider auf keine weiteren Einzelheiten

4 Diskussion

Wir haben in dieser Arbeit zunächst die prinzipielle Vorgehensweise des fallbasierten Schließens erläutert und anschließend die verschiedenen Teilaufgaben diskutiert, die bei der Bearbeitung eines konkreten Anwendungsbereiches mit einem fallbasierten Ansatz auftreten. Das Beispiel der Kreditwürdigkeitsprüfung illustrierte, wie diese Teilaufgaben im Einzelfall gelöst werden können, und zeigte darüberhinaus, daß fallbasiertes Schließen im Vergleich zu anderen Ansätzen bei einer Anwendung in der Finanzwelt sehr gute Ergebnisse liefert. Wir wollen nun die verschiedenen Methoden aufgrund eines hypothetischen Datenraums gegenüberstellen und die verschiedenen Vor- und Nachteile des fallbasierten Schließens zusammenfassen.

4.1 Vergleich

In diesem Abschnitt werden die Verfahren der *linearen Diskriminanzanalyse*, der *Induktionsbäume*, der *Neuronalen Netze* und des *fallbasierten Schließens* in ihrer Funktionsweise bei der Lösung von Klassifikationsaufgaben verglichen. Dazu wollen wir zunächst einige Annahmen vereinbaren, die allerdings nichts an der prinzipiellen Vorgehensweise der Verfahren ändern. Wir betrachten hier die Situation eines zweidimensionalen Datenraums. Jeder Fall sei durch zwei Attribute und durch eine von zwei möglichen Klassifikationen des Falles repräsentiert. Die Ausprägung der beiden Attribute bestimmt die Lage eines Falles im Datenraum bezüglich der x- und der y-Achse. Die Klassifikation eines Falles ist in Abbildung 4 durch die hellen und dunklen Kreise dargestellt. Bei einer graphischen Interpretation der Klassifikation ergibt sich die Aufgabe, den Datenraum so zu partitionieren, daß in jedem Teilbereich lediglich Fälle mit der gleichen Klasse liegen.

- *Lineare Diskriminanzanalyse:*

  Um den Datenraum zu partitionieren, sucht die lineare Diskriminanzanalyse eine Gerade, die die Fälle nach den beiden Klassen möglichst gut trennt (vgl. Abbildung 4, a)). Dabei sollen möglichst viele Fälle oberhalb der Geraden zu der einen Klasse gehören und die Fälle der anderen Klasse sollen möglichst unterhalb der Geraden liegen. Ohne an dieser Stelle etwas über die Funktionsweise der anderen Verfahren zu wissen, erkennt man hier bereits, daß sehr viele Konstellationen im Datenraum existieren, bei denen dieses Verfahren nicht ausreichend ist.

- *Induktionsbäume:*

  Induktionsbäume versuchen achsenparallele Geraden durch den Datenraum zu legen, so daß in jedem entstehenden Rechteck nur Fälle einer Klasse liegen. Jede Ge-

dieser Studie eingehen.

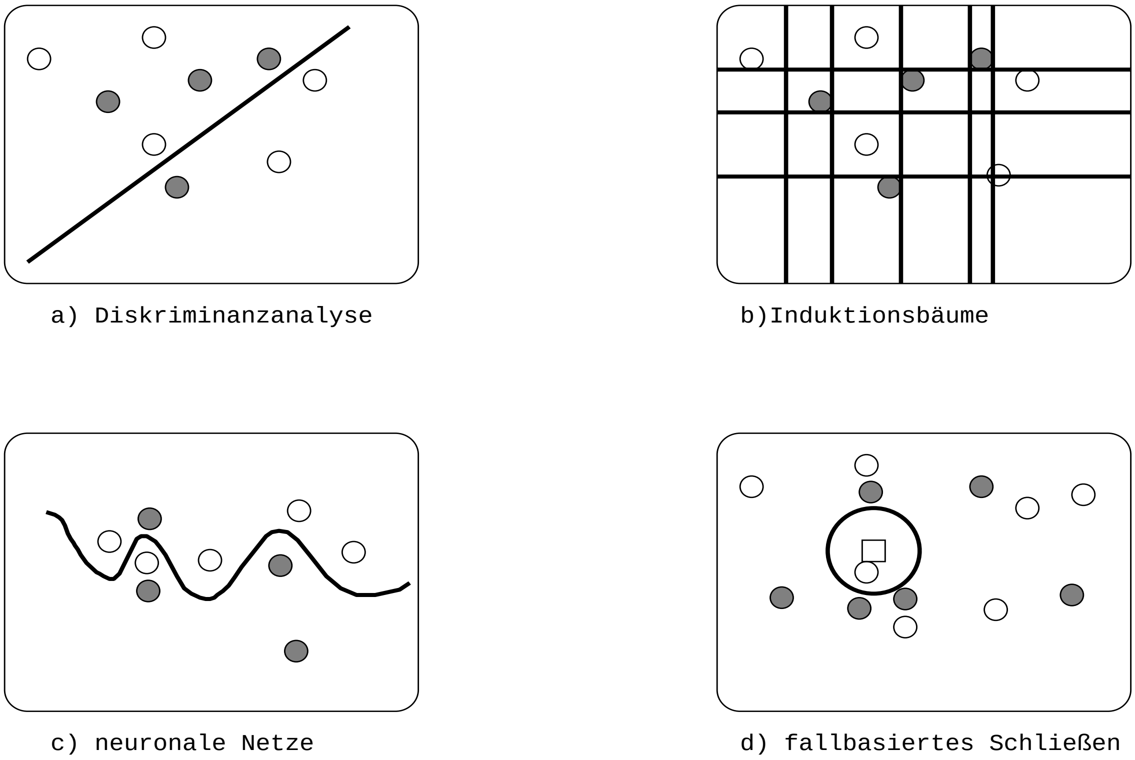

Abbildung 4: Verschiedene Methoden der Klassifikation im Vergleich

rade entspricht einem Knoten des Induktionsbaums, der die Fälle nach einem Attributwert partitioniert. Obwohl hier eine mehrfache Unterteilung in einer Dimension erfolgen kann, existieren dennoch Konstellationen, bei denen dieses Verfahren nicht funktioniert (z.B. den hellen Fall rechts unten in Abbildung 4, b)). Ein präziser Vergleich der linearen Diskriminanzanalyse und der Klassifikation mit Induktionsbäumen befindet sich beispielsweise in [Huber and Nakhaeizadeh, 1993].

- *Neuronale Netze:*

  Neuronale Netze versuchen eine stetige Funktion durch den Datenraum zu legen, so daß die Fälle der einen Klasse oberhalb dieser Funktion liegen und die Fälle der anderen Klasse unterhalb angeordnet sind. Diese Funktion wird durch Interpolation aus den Koordinaten der Falldaten angenähert (vgl. Abbildung 4, c)). Dieses Verfahren ist bereits mächtiger als die beiden zuvor betrachteten Methoden. Dennoch existieren Konstellationen des Datenraums, die nicht durch Neuronale Netze erfaßt werden können. Beispielsweise können drei Fälle, die bei einem Attribut die gleiche Ausprägung aufweisen und zwei verschiedenen Klassen angehören, nicht mit einer linearen Funktion getrennt werden. Ein weiterer Nachteil dieses Verfahrens besteht darin, daß meist sehr viele Beispiele benötigt werden, um die

Funktion zu interpolieren.

- *Fallbasiertes Schließen:*

  Fallbasiertes Schließen bildet zunächst den neuen zu klassifizierenden Fall in den Datenraum ab (siehe das helle Quadrat in Abbildung 4, d)). Dann werden die Klassifikationen betrachtet, die in der Umgebung des Falles liegen. Aus diesen Klassifikationen wird anschließend auf die Klassifikation des Anfragefalles geschlossen. Unter der Voraussetzung, daß die Daten in der Fallbasis richtig akquiriert sind und Fälle derselben Klasse dieselben Attributausprägungen aufweisen, ist dieses Verfahren sehr mächtig. Insbesondere sind hier Fehlklassifikationen nicht dem Verfahren immanent, sondern hängen lediglich von der Qualität der Daten ab, die in die Fallbasis aufgenommen werden. Allerdings liegt eine mögliche Gefahr des Verfahrens darin, daß sogenannte *Ausreißer* ebenfalls gelernt werden, obwohl sie als solche nicht repräsentativ sind. An dieser Stelle sollten zusätzliche Methoden verwendet werden, um solche Ausreißer vor der fallbasierten Problemlösung zu entfernen. Ein präziser Vergleich der Verfahren findet sich in [Nakhaeizadeh, 1993].

4.2 Vor- und Nachteile des fallbasierten Schließens

An dieser Stelle fassen wir die Vor- und Nachteile des fallbasierten Schließens zusammen. Die Vorteile des fallbasierten Schließens ergeben sich nach den vorangegangenen Betrachtungen zu den folgenden Punkten:

- Integration numerischer und symbolischer Techniken
- Bearbeitung komplexer Anwendungen mit zahlreichen Ausnahmesituationen
- Berücksichtigung konkreter Erfahrungen
- Unterstützung unvollständig verstandener Problemstellungen
- Beitrag zum Wissensakquisitionsproblem

Demgegenüber ergeben sich die folgenden Nachteile des fallbasierten Schließens:

- keine direkte Berücksichtigung existierender Regelzusammenhänge
- ungeeignet für Anwendungen, in denen keine Falldaten unmittelbar vorliegen
- gegebenenfalls relativ hoher Aufwand bei der Entwicklung

In einer konkreten Anwendung müssen diese Vor- und Nachteile gegeneinander abgewägt werden, bevor die Entscheidung für den Einsatz eines fallbasierten Systems getroffen wird.

4.3 Fazit

Fallbasiertes Schließen stellt ein sehr mächtiges Verfahren dar. Die speziellen Eigenschaften eines Anwendungsbereiches müssen letztlich darüber entscheiden, ob der Einsatz eines fallbasierten Ansatzes sinnvoll ist. In der Finanzwelt kann fallbasiertes Schließen eine echte Alternative zu traditionellen Ansätzen - wie beispielsweise der Einsatz Neuronaler Netze - bieten.

Literatur

[Aamodt and Plaza, 1994] Aamodt, A. and Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications*, 7(1):39–59.

[Aha, 1989] Aha, D. W. (1989). Incremental, instance-based learning of independend and graded concepts. In *Proceedings of the 6th international Workshop on Machine Learning*, pages 387–391.

[Althoff et al., 1995] Althoff, K.-D., Auriol, E., Barletta, R., and Manago, M. (1995). *A Review of Industrial Case - Based Reasoning Tools.* AI Perspectives Report, AI Intelligence PO Box 95 OXFORD OX2 7XL United Kingdom.

[Bergmann and Wilke, 1995] Bergmann, R. and Wilke, W. (1995). Building and refining abstract planning cases by change of representation language. *Journal of Artificial Intelligence Research*, 3:53–118.

[Bock, 1974] Bock, H. H. (1974). *Automatische Klassifikation.* Vandenhoeck & Ruprecht in Göttingen.

[Brigitte Bartsch-Spörl, 1995] Brigitte Bartsch-Spörl (1995). KI-Merthoden für inovative Design Domänen. In Richter, M. M. and Maurer, F., editors, *Proceedings of the third German Conference on Expert Systems*, pages 137–151, infix Verlag.

[Carbonell, 1986] Carbonell, J. G. (1986). Derivational Analogy : A Theory of Reconstructive Problem Solving and Expertise Acquisition. In Michalski, R., Carbonell, J. G., and Mitchell, T., editors, *Machine Learning: An Artificial Intelligence Approach*, volume 2. Morgan Kaufmann, Palo Alto, California.

[Derwand, 1994] Derwand, G. (1994). Effizientes retrieval und extraktion von entscheidungswissen mit kd-bäumen. Master's thesis, Universität Kaiserslautern, Fachbereich Informatik.

[Goos, 1995] Goos, K. (1995). Fallbasiertes Klassifizieren: Methoden, Integration und Evaluation. *Diss. Universität Würzburg Fakultät für Mathematik und Informatik.*

[Huber and Nakhaeizadeh, 1993] Huber, K.-P. and Nakhaeizadeh, G. (1993). *Maschinelle Lernverfahren als Unterstützung beim Wissenserwerb von Diagnose-Expertensystemen*, pages 167–180. Springer Verlag.

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

[Kolodner, 1983b] Kolodner, J. L. (1983b). Reconstructive Memory: A Computer Model. *Cognitive Science*, 7:281–328.

[Kolodner, 1993] Kolodner, J. L. (1993). *Case-Based Reasoning.* Morgan Kaufmann.

[Koton, 1988] Koton, P. (1988). Reasoning about Evidence in Causal Explanation. pages 260–170.

[Michalski et al., 1986] Michalski, R., Mozetic, I., Hong, J., and Lavrarc, N. (1986). The multi-purpose incremental learning system AQ15 and its testing application on three medical domains. In *Proceedings of the American Association of Artificial Intelligence Conference, Philadelphia.* AAAI.

[Muñoz et al., 1994] Muñoz, H., Paulokat, J., and Wess, S. (1994). Controlling nonlinear hierarchical planning by case replay. In *Proceedings of the 2nd EWCBR*, Paris.

[Nakhaeizadeh, 1993] Nakhaeizadeh, G. (1993). Learning prediction of time series. a theoretical and empirical comparison of cbr with some other approaches. In *Proceedings First European Workshop on Case-Based Reasoning (EWCBR-93)*, pages 149–153.

[Nebel and Koehler, 1993] Nebel, B. and Koehler, J. (1993). Plan modification versus plan generation: A complexity theoretic perspective. In *Proceedings 7. Workshop Planen und Konfigurieren.*

[Paulokat and Wess, 1994] Paulokat, J. and Wess, S. (1994). Planning for machining workpieces with a partial-order nonlinear planner. In Gil and Veloso, editors, *AAAI-Working Notes 'Planning and Learning: On To Real Applications'*. New Orleans.

[Pfitzner, 1993] Pfitzner, K. (1993). Fallbasierte konfigurierung technischer systeme. *KI- Künstliche Intelligenz*, (1):24–30.

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

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

[Steuernagel, 1989] Steuernagel, R. (1989). Integration einer fallbasierten Lernkomponernte in eine Expertensystemshell zur Konfigurationsplanung. Diplomarbeit, Institut für Werkzeugmaschinen und Betriebstechnik, Universität Karlsruhe.

[Veloso, 1992] Veloso, M. (1992). *Learning by Analogical Reasoning in General Problem Solving.* Phd thesis cmu-cs-92-174, Carnegie Mellon University.

[Vietze, 1993] Vietze, T. (1993). Case-based configuration in technical domains: Combining case selection and modification. In *Proceedings EWCBR-93.*

[Weiss and Kulikowski, 1991] Weiss, S. and Kulikowski, C. (1991). *Computer Systems That Learn – Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems.* Morgan Kaufmann.

[Wess, 1991a] Wess, S. (1991a). PATDEX/2: Ein fallbasiertes System zur technischen Diagnostik. SEKI-Working Paper SWP91/01, Dept. of Computer Science, University of Kaiserslautern, Germany.

[Wess, 1991b] Wess, S. (1991b). Patdex/2 - ein fallbasiertes system zur diagnose. In *Jahrestreffen der Fachgruppe Klassifikation und Diagnostik.*

[Wess, 1991c] Wess, S. (1991c). PATDEX/2 – ein fallbasiertes System zur Diagnose. Technical report, Dept. of Computer Science, University of Kaiserslautern, Germany.

[Wess, 1995] Wess, S. (1995). *Fallbasiertes Problemlösen in wissensbasierten Systemen.* PhD thesis, Dissertation Fachbereich Informatik, Universität Kaiserslautern.

[Wess et al., 1993] Wess, S., Althoff, K.-D., and Derwand, G. (1993). Improving the retrieval step in case-based reasoning. In *Proceedings First European Workshop on Case-Based Reasoning*, pages 83–88.

[Wess et al., 1994] Wess, S., Althoff, K.-D., and Derwand, G. (1994). Using kd-trees to improve the retrieval step in case-based reasoning. In *Topics in Case-Based Reasoning*, pages 83–88.

[Wess et al., 1992] Wess, S., Althoff, K.-D., Maurer, F., Paulokat, J., Präger, R., and Wendel, O. (1992). Fallbasiertes schließen - eine übersicht. Seki reports, University of Kaisersllautern.