Maschinelles Lernen Bayes'scher Netze für benutzeradaptive Systeme

Dissertation
zur Erlangung des Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)
der Naturwissenschaftlich-Technischen Fakultät I der Universität des Saarlandes

vorgelegt von

Frank Wittig

Saarbrücken
Dezember 2002

Datum des Kolloquiums:
23.12.2002
Dekan:
Prof. Dr. Philipp Slusallek
Vorsitzender:
Prof. Dr. Gerhard Weikum
Gutachter:
1. Prof. Dr. Dr. h.c. mult. Wolfgang Wahlster
2. Prof. Dr. Anthony Jameson

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet.

Diese Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form in anderen Prüfungsverfahren vorgelegt.

Saarbrücken, den 2. Dezember 2002

Danksagung

Die vorliegende Arbeit entstand im Projekt READY des von der Deutschen Forschungsgemeinschaft (DFG) geförderten Sonderforschungsbereichs 378 „Ressourcenadaptive kognitive Prozesse“ an der Universität des Saarlandes in Saarbrücken. Ich möchte allen Kollegen danken, die es mir ermöglicht haben, die Arbeit in dieser Form zu realisieren.

Mein Dank gilt insbesondere meinem Doktorvater Prof. Dr. Dr. h.c. mult. Wolfgang Wahlster, der es mir mit einer Anstellung an seinem Lehrstuhl möglich gemacht hat, dieses interessante Thema im Rahmen einer Doktorarbeit in einem interdisziplinären Umfeld zu bearbeiten. Ich danke ihm für seine zahlreichen Anregungen und das Interesse, mit der er diese Arbeit begleitet hat.

Prof. Dr. Anthony Jameson danke ich für eine Vielzahl von Vorschlägen und Tipps, die es mir ermöglichten, diese Arbeit in der vorliegenden Form zu erstellen. Seine Fähigkeit, immer wieder interessante Fragestellungen aufzuwerfen, und seine Herangehensweise an wissenschaftliche Problemstellungen im Allgemeinen haben meine Arbeit geprägt. Von den Erfahrungen mit seiner wissenschaftlichen Arbeitsweise, die ich im Verlauf der Zusammenarbeit erlangt habe, werde ich sicherlich in Zukunft profitieren können.

Allen Mitarbeitern und Studenten des Lehrstuhls gilt mein Dank für die angenehme und produktive Arbeitsatmosphäre. Insbesondere danke ich meine beiden Bürokollegen Jörg Baus und Thorsten Bohnenberger, die mich auch während der Zeiträume „ertragen“ haben, in denen ich die verschiedenen Verfahren implementiert und evaluiert habe, was nicht immer ohne erstaunte oder frustrierte Ausrufe meinerseits vonstatten ging. Boris Brandherm gab mir in frühen Stadien Rückmeldung zu einigen meiner Ideen.

Verschiedene Personen haben zur Erhebung und Vorverarbeitung der in dieser Arbeit verwendeten empirischen Daten beigetragen: Barbara Großmann-Hutter, Tore Knabe, Juergen Kiefer und Christian Müller. Björn Decker unterstützte mich bei der Implementation und deren Optimierung. Thorsten Bohnenberger hat eine frühere Version dieser Arbeit Korrektur gelesen. Vielen Dank!

Der wichtigste Dank geht natürlich an meine Familie. Meine Eltern haben haben mich immer und in jeder Form ermutigt und unterstützt. In schwierigen Situationen kann ich immer auf ihre Unterstützung zählen. Danke Anja!

„Das Gewebe dieser Welt ist aus Notwendigkeit und Zufall gebildet.“

Goethe

Kurzzusammenfassung

Das Thema der vorliegenden Arbeit ist die Anwendung existierender sowie die Entwicklung neuer, spezifisch auf den Fall benutzeradaptiver Systeme zugeschnittener, maschineller Lernverfahren für Bayes'sche Netze. Bislang werden die in benutzeradaptiven Systemen eingesetzten Bayes'schen Netze meist manuell—anhand von theoretischen Überlegungen (von Experten)—konstruiert. Es bietet sich an, die im System anfallenden Interaktionsdaten im Rahmen des Konstruktions- bzw. Wartungsprozesses durch die Anwendung entsprechender maschineller Lernverfahren zur Verbesserung der Systemperformanz auszunutzen. Dieser Arbeit liegt eine integrative Konzeption des maschinellen Lernens Bayes'scher Netze für benutzeradaptive Systeme zugrunde, die gemäß den Anforderungen der zu modellierenden Domäne mit alternativen Verfahren instanziiert werden kann. In diesem Rahmen werden in dieser Arbeit neu entwickelte maschinelle Lern- bzw. Adaptionsverfahren für Bayes'sche Netze vorgestellt, die das gemeinsame Ziel verfolgen, die besonderen Eigenschaften und Anforderungen des Benutzermodellierungskontexts während des Lern- bzw. Adaptionsvorgangs zu berücksichtigen. Diese neuen Verfahren werden in vergleichenden Studien mit alternativ einsetzbaren existierenden Methoden des maschinellen Lernens Bayes'scher Netze evaluiert.

Short Abstract

This thesis focuses on the application of existing and the development of new Bayesian network learning methods that are able to deal with or that can exploit the characteristics of domains of user-adaptive systems. So far, Bayesian networks used by user-adaptive systems have typically been specified manually—on the basis of theoretical considerations (of experts). It seems to be a promising approach to exploit the interaction data that can be collected during the systems' use through the application of machine learning methods in the design and maintenance phases. We present an integrative generic framework that can be instantiated with alternative methods according to the demands of the domain to be modeled. To this end, new Bayesian network learning and adaptation methods are presented that jointly aim to address adequately the characteristics and demands of the user modeling context during the learning and adaptation processes. These methods are evaluated in comparative empirical studies relative to alternative existing standard Bayesian network learning procedures.

Zusammenfassung

In der vorliegenden Arbeit wird der Einsatz maschineller Lernverfahren für Bayes'sche Netze in benutzeradaptiven Systemen behandelt. Auf der Grundlage der Definition Bayes'scher Netze sowie wichtigen Verfahren bzw. relevanten Erweiterungen dieses Konzepts wird ein Überblick der aktuellen Forschung zur Anwendung Bayes'scher Netze in benutzeradaptiven Systemen gegeben.

Maschinelle Lernverfahren für Bayes'sche Netze, welche die speziellen Anforderungen des Benutzermodellierungskontexts berücksichtigen bzw. dessen besondere Charakteristika ausnutzen können, wurden bislang nicht entwickelt.

Das allgemeine maschinelle Lernproblem wird auf den Kontext benutzeradaptiver Systeme übertragen. Diesbezüglich werden Kriterien identifiziert, deren Berücksichtigung in der Entwurfsphase eines benutzeradaptiven Systems von entscheidender Bedeutung für einen erfolgreichen Einsatz maschineller Lernverfahren sein können.

Den in der vorliegenden Arbeit entwickelten Methoden liegt eine Gesamtkonzeption des maschinellen Lernens Bayes'scher Netze in benutzeradaptiven Systemen zugrunde. Es handelt sich dabei um einen integrativen Rahmen, der die grundsätzlichen Zusammenhänge zwischen der Art der vorhandenen Daten, dem A-priori-Wissen, der offline stattfindenden Akquisition von Benutzermodellen in Form Bayes'scher Netze sowie der im Laufzeitbetrieb vorgenommen Adaption der Modelle zusammenfasst. Das damit verfolgte Ziel besteht in der Behandlung der angeführten Kriterien eines Einsatzes maschineller Lernverfahren in benutzeradaptiven Systemen im speziellen Fall Bayes'scher Netze. Aus einem Repertoire existierender und in dieser Arbeit neu entwickelter Methoden können bei der Konstruktion benutzeradaptiver Systeme auf der Basis Bayes'scher Netze gemäß den Anforderungen des Einsatzszenarios adäquate Verfahren ausgewählt werden. Sie können in den Rahmen der Gesamtkonzeption eingeordnet werden. Ein benutzeradaptives System, das maschinelle Lernverfahren für Bayes'sche Netze verwendet, bildet in dieser Weise eine Instanziierung der generischen integrativen Konzeption. Typischerweise muss nur ein Teil der Gesamtkonzeption im zu entwickelnden System implementiert werden, um den gestellten Anforderungen zu genügen.

Mit der vorliegenden Arbeit werden folgende konkreten Beiträge geleistet:

- *Identifikation von Kriterien der Anwendung maschineller Lernverfahren in benutzeradaptiven Systemen und deren Behandlung im Fall Bayes'scher Netze mit den entwickelten Methoden*

- *Integration existierender und neu entwickelter Verfahren in einer Gesamtkonzeption des maschinellen Lernens Bayes'scher Netze für und in benutzeradaptiven Systemen*

- *Entwicklung einzelner, speziell auf den Kontext benutzeradaptiver Systeme zugeschnittener maschineller Lernverfahren für Bayes'sche Netze:*
  - Lernen interpretierbarer bedingter Wahrscheinlichkeiten mit qualitativen Constraints
  - Differentielle Adaption bedingter Wahrscheinlichkeiten zur Erfassung und Behandlung individueller Unterschiede zwischen den Benutzern
  - Strukturelle Adaption von Benutzermodellen in Form Bayes'scher Netze mit Meta-Netzen
- *Empirische Fundierung der Benutzermodelle des* READY*-Szenarios:*
  - kognitive Ressourcenbeschränkungen eines Benutzers können mit Hilfe erlernter dynamischer Bayes'scher Netze anhand von Symptomen seiner gesprochenen Sprache erkannt werden
  - Empirisch fundierte Adaption des Präsentationsmodus eines ressourcenadaptiven Dialogsystems anhand eines erlernten Bayes'schen Netzes zur Fehlervermeidung bzw. Beschleunigung der Arbeitsgeschwindigkeit

Mit dem in dieser Arbeit neu entwickelten Verfahren des *Lernens mit qualitativen Constraints* werden wichtige Teile der Gesamtkonzeption bzw. der identifizierten Kriterien behandelt. Das Verfahren ermöglicht das Erlernen interpretierbarer Bayes'scher Netze hinsichtlich der wichtigen Aufgabe des Lernens der bedingten Wahrscheinlichkeiten. Durch das Ausnutzen von vorhandenem A-priori-Wissen über qualitative Zusammenhänge zwischen den im Bayes'schen Netz betrachteten Variablen können gerade bei wenigen, unvollständigen Trainingsdaten die Ergebnisse des Lernvorgangs im Vergleich zu den Standardverfahren sowohl hinsichtlich der (numerischen) Qualität der Modellierung als auch bezüglich des Aspekts der Interpretierbarkeit deutlich verbessert werden.

Die neu entwickelte Methode der *differentiellen Adaption der bedingten Wahrscheinlichkeiten* nutzt existierende Adaptionsverfahren, um unterschiedliche Aspekte des Benutzermodells mit verschiedenen Adaptionsgeschwindigkeiten anzupassen. Modellbereiche, die sich durch große individuelle Unterschiede auszeichnen, werden schneller anhand der gesammelten Adaptionsdaten modifiziert als Bereiche, in denen die meisten Benutzer größtenteils übereinstimmen. Dazu werden—vereinfacht dargestellt—anhand der Varianzen der individuellen Benutzermodelle Adaptionsparameter in Form von lokalen so genannten ESS-Werten bestimmt, welche die Adaptionsgeschwindigkeiten im Rahmen des Bayes'schen Adaptionsvorgangs festlegen.

Der Ansatz des *strukturellen Lernens mit Meta-Netzen von Hofmann (2000) wird im Kontext benutzeradaptiver Systeme angewendet*, mit dem Ziel, das Verständnis der der modellierten Domäne zugrunde liegenden Struktur zu erhöhen. Meta-Netze bieten die Möglichkeit, die strukturelle Unsicherheit, die insbesondere beim Strukturlernen mit wenigen Trainingsdaten eine Rolle spielt, kompakt zu repräsentieren und auszuwerten. Aufbauend auf dieser Methode wird mit der *strukturellen Adaption mit Meta-Netzen* ein neues Adaptionsverfahren beschrieben, das die Struktur eines Bayes'schen Netzes an Veränderungen des Kontexts anpassen kann.

Abstract

This thesis addresses machine learning techniques for Bayesian networks in the context of user-adaptive systems. On the basis of the definition of a Bayesian network and the discussion of related methods and extensions of this framework, an overview of current research on the application of Bayesian networks within user-adaptive systems is presented. So far, there have been no major efforts to develop Bayesian network learning algorithms that are especially well suited to dealing with the demands of the user modeling context, or that are able to exploit the specific characteristics of this context.

The general formulation of the machine learning problem is transferred to the context of user-adaptive systems. Several crucial criteria are identified that have to be addressed adequately during the planning phase to ensure a successful application of machine learning techniques in a user-adaptive system.

The methods that are developed in this thesis, along with already existing standard learning methods, are integrated into a general framework that can be instantiated according to the demands of the domain to be modeled. This generic framework describes the relationships between available data, prior knowledge, offline acquisition of Bayesian network user models, and the models' online adaptation. The purpose of this framework is to address the identified criteria for the successful application of machine learning methods in user-adaptive systems in the case of Bayesian networks. The system's developer can choose an adequate selection from a repertoire of existing and new methods that are presented in this thesis. This selection can be arranged within the generic framework, thereby yielding a specific instance of the framework. Typically, only a subset of the whole range of possibilities has to be implemented in order for the demands of the domain under consideration to be satisfied.

The following concrete contributions are made by this thesis:

- *Identification of crucial criteria for a successful application of machine learning methods in the context of user-adaptive systems and the discussion of solutions for the case of Bayesian network user models*

- *Integration of existing and newly developed methods into a generic framework for learning Bayesian networks for user-adaptive systems*

- *Development of several machine learning algorithms for Bayesian networks with focus on the demands and characteristics of the user modeling context:*

  - Learning interpretable tables of conditional probabilities using qualitative constraints
  - Differential adaptation of conditional probabilities to take into account individual differences between users
  - Structural adaptation of Bayesian network user models with meta networks

- *Empirical grounding of the* READY*-scenario's user models:*
  - It is shown that it is possible to recognize a user's cognitive resource limitations using learned dynamic Bayesian networks on the basis of symptoms of the user's speech.
  - Empirically grounded adaptation of the presentation of instructions in a resource-adaptive dialog system using a learned Bayesian network, with the goal of avoiding errors and increasing the efficiency of task execution.

The method of learning with qualitative constraints that is presented in this thesis addresses several of the identified crucial criteria. The method provides the opportunity to learn Bayesian networks with interpretable conditional probabilities. By exploiting available qualitative prior knowledge regarding the dependencies between the networks' variables it is possible—especially in situations with limited and/or missing data—to improve the learning results with regard to their inferential performance as well as the interpretability of the learned models.

The differential adaptation method for revising the conditional probabilities uses standard adaptation techniques to adapt different parts of the Bayesian network user model at different rates. Those parts that are characterized by large individual differences between the users are adapted faster than parts that represent user properties that are generally similar across users. To realize such a behavior, the method determines for each part separately an adaptation parameter by computing local equivalent sample sizes, which in turn determine the adaptation rates. In essence, these adaptation parameters are computed on the basis of a comparison of the previously learned user models.

The structural learning with meta networks described by Hofmann (2000) is applied in the user modeling context to increase the understanding of the underlying structure of the modeled domain. Meta networks are an opportunity to model structural uncertainty—which plays an important role when only limited data is available—in a compact and efficient manner. On the basis of this method, a new structural adaptation algorithm is presented in this thesis that is able to cope with temporal changes regarding the structure of the Bayesian network user models.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 EINLEITUNG

Die Bedeutung *Bayes'scher Netze* als Inferenzkomponente in intelligenten Systemen hat in den vergangenen Jahren stark zugenommen. Sie kristallisieren sich zunehmend als das bevorzugte Standardwerkzeug zum Schlussfolgern unter Unsicherheit in der Forschung der Künstlichen Intelligenz heraus. Im Verlauf des letzten Jahrzehnts wurde stark an der Entwicklung *maschineller Lerntechniken*[1] zum automatischen Erlernen Bayes'scher Netze anhand verfügbarer empirischer Daten gearbeitet. Thema der vorliegenden Arbeit ist die Entwicklung neuer und die Anwendung existierender maschineller Lernverfahren für Bayes'sche Netze im Kontext *benutzeradaptiver Systeme*.[2] Bislang wurden die in solchen Systemen eingesetzten Netze meist manuell auf der Basis theoretischer Überlegungen spezifiziert. Durch den Einsatz entsprechender maschineller Lernverfahren können die oft vorhandenen Interaktionsdaten mit in den Modellierungsprozess einfließen. Im Folgenden wird eine weitergehende Einordnung der Arbeit sowie eine ausführliche Diskussion der dieser Arbeit zugrunde liegenden Fragestellungen vorgenommen.

1.1 Einordnung

Eine der Voraussetzungen eines Gesprächs, das von allen Beteiligten als sinnvoll empfundenen wird, ist, dass sich die Gesprächsteilnehmer zu einem gewissen Grad auf den jeweiligen Partner einstellen. Nur so kann im Allgemeinen der Austausch von Informationen und Argumenten in einer für alle Seiten produktiven Weise stattfinden. Dazu gehört auf einer unteren Kommunikationsebene erst einmal, dass man sich auf eine gemeinsame Sprache verständigt. Auf einer „höheren" Ebene der Kommunikation spielt im Verlauf des Gesprächs aber auch beispielsweise die Berücksichtigung der Gesprächssituation eine Rolle. Wird man z.B. auf einem Bahnhof von einer Person, die durch die nahende Abfahrt ihres Zuges unter Zeitdruck steht, in Englisch nach einer Wegbeschreibung zum Ticketautomaten gefragt, so sollte man sich auf die Mitteilung der wichtigsten Informationen mittels einer kurzen englischen Äußerung (in Kombination mit Zeigegesten) beschränken, wie beispielsweise „20 meters this direction, then turn left!". In einer anderen Situation, in der der Gesprächspartner nicht unter Zeitdruck steht, könnte man ausführlicher antworten und gegebenenfalls zusätzliche Informationen mitteilen—beispielsweise, dass es etwas weiter entfernt auch einen Ticketschalter gibt, wo man Platzreservierungen vornehmen kann.

[1] Eine ausführliche Einführung in dieses Forschungsgebiet gibt beispielsweise Mitchell (1997).

[2] Häufig wird gerade bei webbasierten Systemen auch der Begriff *personalisierte* Systeme verwendet. Jameson (2002) gibt einen ausführlichen Überblick über die aktuelle Forschung in diesem Gebiet.

1.1.1 Benutzeradaptive Systeme

Es ist wünschenswert, dass interaktive Softwaresysteme möglichst in ähnlicher Weise auf ihren „Gesprächspartner“ am Bildschirm, d.h., den Benutzer,[3] reagieren und sich automatisch auf dessen Bedürfnisse einstellen. Dabei sollten soweit wie möglich die individuellen (Interaktions-)Vorlieben, die aktuell verfolgten Ziele des System-Benutzers und / oder der aktuelle Kontext bei der Anpassung beachtet werden. Anhand typischer Beispiele aus drei Bereichen der aktuellen Forschung können die Problemstellungen anpassungsfähiger Softwaresysteme verdeutlicht werden.

1.1.1.1 Funktionalitäten

Adaptive Webseiten Die erfolgreiche Entwicklung des World Wide Webs (WWW) zu einem alltagstauglichen Medium während der letzten Jahre macht es immer wichtiger, die große Menge der verfügbaren Information innerhalb webbasierter Systeme für den Benutzer individuell aufzuarbeiten und zu präsentieren. Beispielsweise sollten von einem WWW-Shopsystem im Rahmen der Erstellung von Produktempfehlungen und -präsentationen sowohl die variierenden Interessen zwischen den einzelnen potenziellen Kunden als auch deren unterschiedlich stark ausgeprägten Kenntnisse—insbesondere im Fall technischer Produkte—berücksichtigt werden. So sollen einerseits aus Verkäufersicht die Aussichten auf einen erfolgreichen Geschäftsabschluss erhöht werden, andererseits soll aus Kundensicht auf diese Weise der möglichst schnelle Zugang zu den gewünschten produktspezifischen Informationen—möglichst ohne eine aufwendige Suche—gewährleistet werden. Dazu muss ein solches anpassungsfähiges WWW-Shopsystem über Verfahren und Datenstrukturen verfügen, um—unter Berücksichtigung von Datenschutzaspekten (Kobsa, 2001b)—relevante Informationen über die Interessen der Benutzer zu sammeln, zu verwalten und entsprechend auszunutzen. Gerade in einem solchen E-Commerce-Szenario, das sich durch gute Möglichkeiten zur Erhebung von Interaktions- und Benutzerdaten auszeichnet, bietet sich die Anwendung von Verfahren des maschinellen Lernens an. Mit ihnen kann die große Menge sowohl impliziter als auch expliziter Rückmeldungen des Benutzers, beispielsweise in Form bereits getätigter Einkäufe, der vom Benutzer innerhalb des Online-Shops besuchten Seiten, oder explizite Angaben über gewünschte Produkte, ausgewertet werden, um die Interessen der potenziellen Kunden zu ermitteln und entsprechende „maßgeschneiderte“ Angebote automatisch unterbreiten zu können. Das wohl bekannteste sich im erfolgreichen kommerziellen Einsatz befindliche System dieser Art ist der WWW-Shop der Online-Buchhandlung AMAZON.[4]

Komplexe Softwaresysteme Auch der Bedienkomfort der an Komplexität ständig zunehmenden Desktop-Softwaresysteme, wie es insbesondere bei Office-Anwendungen zu beobachten ist, kann potenziell durch eine Erweiterung um Funktionalitäten zur Anpassung (*Adaption*) an die individuellen—möglicherweise situationsbedingten—Anforderungen und Arbeitsmethoden gesteigert werden. Dies kann etwa durch Anpassung von Menüstrukturen erreicht werden, wobei z.B. die am häufigsten genutzten oder aktuell benötigten Einträge automatisch an prominente Stellen platziert werden. Ein anderes Beispiel sind kontextbezogene Hilfesysteme, die—ohne explizite Anforderung durch den Benutzer—anhand des aktuellen Systemzustands, des Wissens des Systems über die Benutzerkenntnisse, und der letzten durchgeführten Aktionen adäquate, unterstützende

[3] Die männliche Form des Wortes ‘Benutzer’ soll in dieser Arbeit der Einfachheit halber auch Benutzerinnen mit einschließen.

[4] www.amazon.com

Vorschläge oder weiterführende Informationen optional anbieten. Auch hier spielt der Einsatz von Techniken des maschinellen Lernens eine wichtige Rolle. Ergebnisse der Forschung sind in diesem Bereich bereits in kommerzielle Systeme eingeflossen. Die MS OFFICE 97 ASSISTENTEN (Horvitz, Breese, Heckerman, Hovel & Rommelse, 1998) gehören diesbezüglich zu den bekanntesten kommerziellen Projekten.

Intelligente Lehr-/Lernsysteme Weiterer Schwerpunkt der aktuellen Forschung solcher anpassungsfähiger Softwaresysteme sind intelligente Lehr-/Lernsysteme.[5] Gerade hier erscheint es hinsichtlich des Lernerfolgs intuitiv vielversprechend, dass sich solche Systeme auf der Basis von Informationen über die (nicht) vorhandenen Kenntnisse, sowohl bei der Auswahl der Lehrstrategie als auch bei der Bereitstellung entsprechender Funktionalitäten zur Bedienung der Systeme, automatisch individuell an die Lernenden anpassen—ebenso wie ein guter Lehrer in der Schule versuchen sollte, die unterschiedlichen Fähigkeiten und Lerngewohnheiten seiner Schüler zu berücksichtigen, aber in der Praxis meist verständlicherweise an der relativ großen Anzahl an Schülern einer Klasse scheitert. Hier besitzen intelligente Lehr-/Lernsysteme prinzipiell ein großes Potenzial, dem Lernenden gezielt *die richtigen Inhalte zur richtigen Zeit in der richtigen Art und Weise zu präsentieren.*

Wachsender Bedarf an benutzeradaptiven Systemen Jameson (2002) führt die folgenden Gründe an, warum aufgrund der aktuellen Entwicklungen der Informationstechnologie und der Durchdringung unseres Alltags mit entsprechenden Geräten ein wachsender Bedarf an solchen Systemen, die sich an den aktuellen Benutzer anpassen können—man spricht von benutzeradaptiven Systemen—, zu erwarten ist:

1. *Vielzahl unterschiedlicher Benutzer und Anwendungskontexte:* Wie die Beispiele der vorangehenden Abschnitte deutlich machen, werden Softwaresysteme in verstärktem Ausmaß in Kontexten eingesetzt, die durch eine große Anzahl von Benutzern mit unterschiedlichen Kenntnissen und Interessen charakterisiert sind. Es wird damit immer schwieriger, Programme zu entwickeln, die ohne Benutzeradaptivität vielen Benutzern und Kontexten gerecht werden können.

2. *Anzahl und Komplexität interaktiver Systeme:* Immer mehr technische Geräte—und damit auch Software—nimmt Einzug in den Alltag unserer Gesellschaft. Es kann nicht erwartet werden, dass sich die Benutzer mit jedem neuen Gerät oder Softwareprogramm vor dem Einsatz intensiv beschäftigen, um die individuell benötigte Funktionalität zu erlernen. In ähnlicher Weise kann nicht erwartet werden, dass jeder (Gelegenheits-)Benutzer alle von komplexen Systemen angebotenen Optionen kennt.

3. *Vielzahl unterschiedlicher Informationstypen:* Die Entwicklung der letzten Jahre hat dazu geführt, dass Benutzer von Informationssystemen einer (oft unübersichtlichen) Menge unterschiedlicher Informationseinheiten und -objekten gegenüber stehen, z.B. Texten, Produkten oder auch menschlichen, potenziellen Kommunikationspartnern. Es erscheint vielversprechend, den Umgang mit diesen Informationseinheiten—zumindest teilweise—von Systemen bearbeiten zu lassen. Ein typisches Beispiel ist hier das Auswählen einzelner Dokumente aus einer großen Sammlung, die von Nutzen für den Benutzer sein können, bzw. das Unterdrücken solcher Dokumente, die nur einen geringen Wert besitzen.

[5] Brusilovsky (2001) gibt einen Überblick der aktuellen Forschung zu adaptiven Lehr-/Lernsystemen.

1.1.1.2 Benutzermodelle

Grundlage solcher benutzeradaptiver Systeme ist eine Wissensquelle, die Informationen über das Verhalten, die Eigenschaften, Vorlieben, Ziele usw. des Benutzers integriert: das *Benutzermodell*[6] (Wahlster & Kobsa, 1989). Es kann im Verlauf der Interaktion zwischen Benutzer und System um neue Informationen erweitert und an veränderte Sachverhalte angepasst werden. Auf dieser Basis ist das benutzeradaptive System prinzipiell in der Lage, die Interaktion (in einem gewissen Ausmaß) individuell für jeden seiner Benutzer zu gestalten (vgl. Abbildung 1.1).

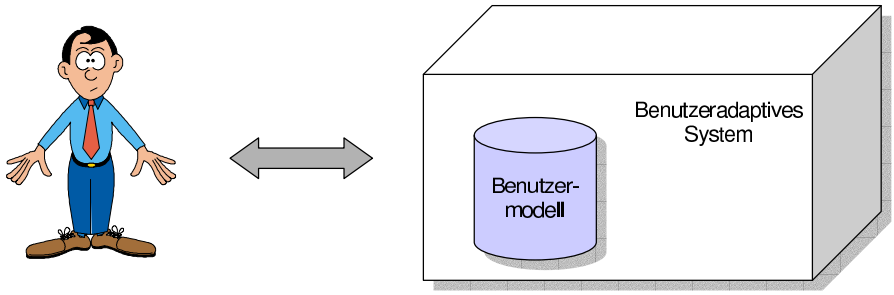


***Abbildung 1.1**: Prototypischer Aufbau eines benutzeradaptives System*

Als Beispiel kann einer der ersten Ansätze zur Repräsentation von Benutzermodellen dienen: der *Stereotypen*-Ansatz von Rich (1979, 1989). Stereotypen modellieren die Benutzer, indem sie Systemnutzer zu prototypischen Klassen zusammenfassen (z.B. zu Anfängern, Durchschnitts-Nutzern und Experten) und den aktuellen Benutzer des Systems dann einer dieser Klassen zuordnen. Damit werden ihm alle klassenspezifischen Eigenschaften zugeschrieben. Ein solches benutzeradaptives System ist somit in der Lage, den Stereotypen entsprechend, die Interaktion mit einem Benutzer zu behandeln: Anfänger erhalten beispielsweise ausführliche Zusatzinformationen, wohingegen Experten nur die zur Lösung tatsächlich benötigten Hilfsmittel und Informationen—ohne weitere Erläuterung—bereitgestellt bekommen.

Es existiert eine Vielzahl weiterer Formalismen aus dem Bereich der Künstlichen Intelligenz, die zur Repräsentation von Benutzermodellen erfolgreich eingesetzt wurden, z.B. regel-basierte Ansätze, künstliche neuronale Netze und Entscheidungsbäume. Die Entscheidung für eine bestimmte Repräsentationsform des Benutzermodells ist in der Regel von den vorgegebenen Anforderungen an das benutzeradaptive System und der Verfügbarkeit von Informationen zu den Benutzern bestimmt.

Abhängig von der Domäne des benutzeradaptiven Systems müssen relevante Benutzerinformationen modelliert werden, um die Adaptivitätseigenschaft realisieren zu können. Die Auswahl, welche Informationseinheiten in ein Benutzermodell aufgenommen werden, stellt eine kritische Entscheidung hinsichtlich des Erfolgs eines Systems dar. U.a. werden häufig die folgenden Arten von Informationen berücksichtigt (nach Kobsa et al., 2001):

- *Demographische Daten zum Benutzer*
- *Wissen des Benutzers*

[6] Manchmal wird bei der Modellierung des Verhaltens von einem „Verhaltensmodell“ (engl. *usage model*) zusätzlich zum Benutzermodell (engl. *user model*) gesprochen (vgl. Kobsa, Koenemann & Pohl, 2001). In dieser Arbeit wird eine solche Unterscheidung keine Rolle spielen, sondern in beiden Fällen von einem Benutzermodell gesprochen werden.

- *Fähigkeiten des Benutzers*
- *Interessen und Vorlieben des Benutzers*
- *Ziele und Pläne des Benutzers*

Anhand des Zusammenwirkens dieser Information und des beobachteten Interaktionsverhaltens werden vom System im Rahmen eines Schlussfolgerungsprozesses die Adaptionsentscheidungen getroffen.

1.1.1.3 Maschinelles Lernen von Benutzermodellen

Benutzermodelle werden oft von—oder in Zusammenarbeit mit—Domänenexperten *manuell* im Rahmen von typischerweise aufwendigen Knowledge-Engineering-Prozessen erstellt. Eine Alternative zu dieser Vorgehensweise, die insbesondere in den letzten Jahren in den Fokus der Aufmerksamkeit gerückt ist (siehe Webb, Pazzani & Billsus, 2001), stellen—wie bereits in den Beispielen angesprochen wurde—Verfahren des maschinellen Lernens zur Konstruktion und Pflege der Benutzermodelle dar, um die Modelle automatisch unter Ausnutzung bereits vorhandener bzw. im Systembetrieb erhobener Daten zu erstellen bzw. zu aktualisieren. Oftmals findet auch eine Kombination von maschinellem Lernen und manueller Spezifikation durch Experten statt.

Generell besteht ein enger Zusammenhang zwischen maschinellen Lernverfahren und benutzeradaptiven Systemen. Allgemeines Ziel beim maschinellen Lernen ist es, anhand von Daten Modelle automatisch zu erstellen bzw. zu verbessern, die vom System zur Vorhersage in neuen Situationen genutzt werden können. In analoger Weise versucht ein benutzeradaptives System seine (zukünftige) Interaktion mit dem Benutzer anhand der bislang gemachten Erfahrungen zu optimieren.

Ein wesentlicher Vorteil des maschinellen Lernansatzes in der Benutzermodellierung besteht—neben dem im Vergleich zum manuellen Vorgehen meist vereinfachten, (teilweise) automatisierten Konstruktionsprozess—in der in vielen Fällen besseren Qualität der so konstruierten Benutzermodelle. Durch die Verwendung vorhandener empirischer Daten kann beispielsweise das Einfließen subjektiver Fehleinschätzungen der Experten in die Benutzermodelle vermieden werden. Obwohl die Anwendung maschineller Lernverfahren im Kontext benutzeradaptiver Systeme auf den ersten Blick vielversprechend erscheint, wirft sie eine Reihe von Fragen und Problemen auf (siehe Abbschnitt 1.3), deren Behandlung für eine spezielle Repräsentationsform der Benutzermodelle das zentrale Thema dieser Arbeit ist.

1.1.2 Das READY-Projekt

Die vorliegende Arbeit entstand im Projekt READY[7] des Sonderforschungsbereichs „Ressourcenadaptive kognitive Prozesse“ (SFB 378) der Deutschen Forschungsgemeinschaft (DFG). Im weiteren Verlauf dieser Arbeit werden einige konkrete Beispiele aus dem READY-Szenario zur Diskussion der relevanten Fragestellungen sowie zur empirischen Evaluation der vorgestellten Verfahren herangezogen. Aus diesem Grund wird in diesem Abschnitt ein kompakter Überblick über die Forschung innerhalb des READY-Projekts gegeben. Ausführlichere Informationen finden sich bei Bohnenberger, Brandherm, Großmann-Hutter, Heckmann und Wittig (2002) sowie Jameson et al. (2001).

[7] READY ist das Akronym für „**RE**ssourcen-**A**daptive **D**ialogs**Y**steme“ (http://w5.cs.uni-sb.de/∼ready).

Die zentrale Problemstellung im READY-Projekt besteht in der *Erkennung und Berücksichtigung von Beschränkungen der kognitiven Ressourcen* der Benutzer eines mobilen Assistenzsystems in einem Flughafenszenario. Aufgabe des prototypisch zu entwickelnden Systems ist die adäquate Unterstützung eines Flughafenbesuchers bei typischen Aufgaben, die von ihm vor dem Abflug erledigt werden müssen, wie z.B. das Aufsuchen des Check-In-Schalters, das Besorgen von Mitbringseln für die Familie, die Bedienung eines unvertrauten technischen Gerätes wie einem Kreditkartentelefon, und schlussendlich das rechtzeitige Aufsuchen des Abfluggates zur Abflugzeit.

Die in READY betrachteten *kognitive Ressourcen* sind der subjektive *Zeitdruck*, unter dem der Fluggast—beispielsweise durch die nahende Abflugzeit—steht, und die *kognitive Belastung*, die durch die zu erledigenden Aufgaben, etwa das Aufsuchen bestimmter Ziele im Flughafen, ablenkende Lautsprecherdurchsagen, die Interaktion mit dem mobilen System usw., induziert wird. Das Szenario sieht vor, dass der Benutzer mit dem READY-System per Sprachein- und -ausgabe sowie mit Hilfe einer graphischen Schnittstelle über einen persönlichen digitalen Assistenten (PDA) interagieren kann. Entsprechend der vom System anhand des Benutzerverhaltens ermittelten Einschätzungen der (nicht) verfügbaren kognitiven Ressourcen—etwa anhand von Fehlern, die der Benutzer während der Ausführung einer Aufgabe macht, oder anhand der Art und Weise, mit der er mit dem System interagiert,—präsentiert der READY-Prototyp in adaptiver Weise Informationen bzw. Hilfestellungen.

Ein solches benutzeradaptives System wie der READY-Prototyp muss in der Lage sein, mit *Unsicherheit* in der Domäne umzugehen. In diesem speziellen Fall kann das System lediglich anhand von Symptomen des Benutzerverhaltens unter Zuhilfenahme des Benutzermodells Rückschlüsse über den aktuellen kognitiven Zustand des Benutzers ziehen. Beispielsweise könnte das System aus einer beschleunigten Artikulationsgeschwindigkeit der gesprochenen Sprache des Benutzers auf einen erhöhten Zeitdruck schließen, wobei gleichzeitig noch viele andere Aspekte der aktuellen Situation—möglicherweise widersprüchlicher Art—mit berücksichtigt werden müssen. Allerdings sind solche Schlussfolgerungen des Systems inhärent stark mit Unsicherheit behaftet, stellen also lediglich eine Einschätzung des Systems dar, die aufgrund von probabilistischen Zusammenhängen ermittelt wird.

Abbildung 1.2 zeigt einen Überblick über die READY-Systemarchitektur. Dabei sind die Einzelkomponenten in Kuchenform angeordnet. Herausdriftende Teile repräsentieren Aufgaben oder Module, die (zu einen gewissen Grad) vor der eigentlichen Interaktion des Benutzers mit dem System angesiedelt sind. Ein Schwerpunkt des Projekts ist die empirische Fundierung der angewendeten Techniken durch psychologische Experimente (March, 1999; Müller, 2001; Kiefer, 2002). Diese Vorgehensweise steht in einem gewissen Gegensatz zur üblichen Verfahrensweise, bei der benutzeradaptive Systeme erst nach ihrer (vorläufigen) Fertigstellung in ausführlichen Studien mit Testpersonen evaluiert (und danach gegebenenfalls nochmals modifiziert) werden (vgl. Chin, 2001). In dem in READY verfolgten Ansatz fließen bereits in einem frühen Stadium empirische Ergebnisse in den Entwicklungsprozess ein, indem Techniken des maschinellen Lernens auf die Experimentaldaten zum Erlernen initialer Benutzermodelle angewendet werden, die als Ausgangspunkt der Adaption an den Benutzer dienen können. Die so konstruierten Modelle werden dann zur Laufzeit des READY-Prototyps anhand neuer Beobachtungen im Rahmen der Benutzerinteraktion modifiziert. Dies verfolgt das Ziel, die initialen Modelle, die auf der Basis von Daten einer Vielzahl von Benutzern gelernt wurden, an die individuellen Eigenschaften des aktuellen Benutzers anzupassen. Sowohl das Lernen der initialen Benutzermodelle als auch der anschließende Adaptionsprozess stehen im Mittelpunkt dieser Arbeit. Im READY-Prototyp liefern diese

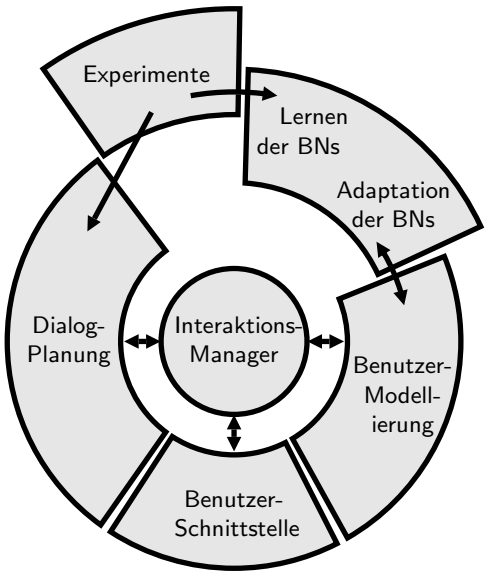


***Abbildung 1.2**: Systemarchitektur des* READY*-Prototyps*
(Die Pfeile repräsentieren den Informationsfluss zwischen den Einzelkomponenten. Weitere Erläuterungen finden sich im Text.)

Benutzermodelle Informationen für die entscheidungstheoretischen Planungsverfahren (Bohnenberger & Jameson, 2001), um die nächsten Dialogschritte unter Berücksichtigung der kognitiven Ressourcenlage des Benutzers zu planen. Der Interaktions-Manager ist als zentrales Modul für die Koordination der Teilmodule zur Bearbeitung der beschriebenen Aufgaben verantwortlich.

1.2 Bayes'sche Netze in benutzeradaptiven Systemen

Bayes'sche Netze (Pearl, 1988) kristallisieren sich immer stärker als eines der wichtigsten Werkzeuge zur Repräsentation von Benutzermodellen in benutzeradaptiven Systemen, die mit Unsicherheit in ihrer Domäne umgehen müssen, heraus (eine ausführliche Diskussion hierzu erfolgt in Abschnitt 2.6). Dazu tragen insbesondere ihre im Folgenden aufgelisteten, im Rahmen benutzeradaptiver Systeme besonders vorteilhaften Eigenschaften, bei (vgl. auch Schäfer, 1998, Abschnitte 1.1, 2.5 für eine Diskussion dieser Thematik im speziellen Kontext benutzeradaptiver Dialogsysteme):

Repräsentation von und Inferenzen unter Unsicherheit Bayes'sche Netze stellen eine *kompakte Repräsentation einer gemeinsamen Wahrscheinlichkeitsverteilung* über der Menge der relevanten Variablen einer Domäne dar. Mit ihrer Hilfe werden probabilistische Beziehungen zwischen den Variablen modelliert, was in vielen Domänen im Kontext benutzeradaptiver Systeme eine Aufgabe ist, die mit über den Erfolg des Systems entscheidet. Beispielsweise kann so in einem intelligenten Lehr-/Lernsystem ein Zusammenhang zwischen dem Wissensniveau eines Studenten und der Note, die er in der Abschlussklausur voraussichtlich erzielen wird, modelliert werden: Ein hohes Wissensniveau bedingt eine hohe Wahrscheinlichkeit einer guten Note, aber dennoch

kann der Student versagen. Möglicherweise werden gerade solche Wissensgebiete abgefragt, die er nicht beherrscht, oder die Klausur ist zu schwierig konzipiert. Die Wahrscheinlichkeiten bestimmter Sachverhalte werden mit der in den Bayes'schen Netzen kodierten Information durch entsprechende *Inferenzalgorithmen* berechnet (siehe Pearl, 1988; Neapolitan, 1990; Jensen, 1996, 2001). Zwar ist die Inferenz in Bayes'schen Netzen generell ein NP-hartes Problem (Cooper, 1990), es hat sich aber gezeigt, dass in vielen praktischen Anwendungen dies entweder kein Hindernis darstellt, approximative Verfahren eingesetzt werden können, oder aber, dass es möglich ist, individuelle Lösungen für die jeweilige spezifische Anwendungssituation zu entwickeln.

Probabilistische Vorhersagen über beliebige Variablen(-mengen) des Modells Einer der wesentlichen Unterschiede im Vergleich mit anderen Formalismen wie beispielsweise künstlichen neuronalen Netzen oder Entscheidungsbäumen (siehe z.B. Russell & Norvig, 1995) besteht in der Fähigkeit Bayes'scher Netze, Wahrscheinlichkeitsverteilungen über beliebige Teilmengen der betrachteten Variablen konditioniert auf die verfügbare Information zu liefern. Andere Formalismen erlauben oft nur Aussagen über eine festgelegte Menge an Vorhersagevariablen. Im bereits angeführten Beispiel bedeutet dies, dass man etwa ein neuronales Netz hat, das auf der Basis der Eingabevariablen KLAUSURNOTE und SCHWIERIGKEIT DER KLAUSUR die (bedingte) Wahrscheinlichkeit für eine bestimmte Ausprägung der Variablen WISSENSNIVEAU vorhersagt. Mit einem Bayes'schen Netz ist im gleichen Modell auch eine Vorhersage für KLAUSURNOTE bei bekannten Werten von WISSENSNIVEAU und SCHWIERIGKEIT DER KLAUSUR möglich, im Gegensatz zum neuronalen Netz, das durch die Partitionierung in dezidierte Ein- und Ausgabevariablen in seiner Funktionsweise diesbezüglich eingeschränkt ist. Diese Flexibilität Bayes'scher Netze ist ein weiterer wichtiger Pluspunkt hinsichtlich der Anwendbarkeit in benutzeradaptiven Systemen. Oft liegt in solchen Systemen nur eine partielle Beobachtbarkeit der Domäne vor. Mit Bayes'schen Netzen kann es aus dem beschriebenen Grund möglich sein, schon auf der Basis sehr weniger verfügbaren Informationen, brauchbare probabilistische Vorhersagen über eine große Anzahl an Sachverhalten von Interesse zu ermitteln.

Erweiterbarkeit zu Einflussdiagrammen Bayes'sche Netze können leicht zu *Einflussdiagrammen* erweitert werden (siehe z.B. Neapolitan, 1990; Jensen, 1996, 2001). Mit Einflussdiagrammen lassen sich Bewertungen alternativer Optionen unter Berücksichtigung einer Bewertungsfunktion in mit Unsicherheit behafteten Domänen berechnen. Beispielsweise könnte der READY-Prototyp ein Einflussdiagramm nutzen, um abzuwägen, ob unter der mittels eines Bayes'schen Netzes eingeschätzten kognitiven Ressourcenlage des Flughafenbesuchers die nächste Navigationsanweisung sprachlich oder in Form einer Karten-Graphik auf dem Bildschirm des PDAs präsentiert werden sollte. Dabei spielen zusätzliche Faktoren eine Rolle, wie eventuell vorhandene Umgebungsgeräusche, das Vermeiden sich zu verlaufen oder das rechtzeitige Erreichen des Flugsteigs, die ebenfalls im Rahmen des Entscheidungsprozesses im Einflussdiagramm zu berücksichtigen sind. Vorhandene Umgebungsgeräusche sollten beispielweise zu einer deutlich schlechteren Bewertung der Option der sprachlichen Ausgabe führen. Mit Einflussdiagrammen werden benutzeradaptive Systeme in die Lage versetzt, anhand des Benutzermodells in Form Bayes'scher Netze—ohne zusätzliche Methoden und Datenstrukturen zu benötigen—Entscheidungen zu treffen, die die verfügbare Information adäquat berücksichtigen.

Modellierung temporaler Aspekte durch dynamische Bayes'sche Netze Temporale Aspekte einer Domäne können mit *dynamischen* Bayes'schen Netzen behandelt werden (Dagum, Galper & Horvitz, 1992). Damit ist es möglich, explizit zeitlich veränderliche Teile des Benutzermodells zu repräsentieren, wie z.B. ein sich verändernder Zeitdruck eines Flughafenbesuchers, der typischerweise immer mehr zunimmt, je näher die Abflugzeit rückt; oder die Verbesserung des Wissensniveaus eines Studenten im Verlauf des Lernens mit einem benutzeradaptiven Lehr-/Lernsystem. Eine ausführliche Behandlung dynamischer Bayes'scher Netze im Kontext benutzeradaptiver Dialogsysteme geben Schäfer und Weyrath (1997) sowie Schäfer (1998).

Interpretierbarkeit durch kausale Interpretation Die Frage der Interpretierbarkeit und Erklärbarkeit von Benutzermodellen in Form Bayes'scher Netze besteht aus zwei Aspekten:

1. Eine erhöhte Interpretierbarkeit trägt zu einer Vereinfachung des Design- und Konstruktionsprozesses bei. Fehler im Modell können leichter lokalisiert und korrigiert werden. Ein Nachvollziehen des Systemverhaltens wird weitestgehend ermöglicht.

2. Die Nachvollziehbarkeit des Systemverhaltens kann die Akzeptanz auf Benutzerseite steigern (siehe z.B. Wahlster, 1981; Teach & Shortliffe, 1984; Cook & Kay, 1994; Herlocker, Konstan & Riedl, 2000). Entscheidungen können vom System erklärt werden, der Benutzer versteht wie sein Verhalten vom System interpretiert wird und welche Informationen wozu genutzt werden.

Da Bayes'sche Netze die kausalen Zusammenhänge zwischen den betrachteten Variablen in Form eines gerichteten Graphen modellieren, ist üblicherweise ein hoher Grad an Interpretierbarkeit gewährleistet. Es ist häufig sehr einfach, anhand der Kanten des Graphen den Inferenzprozess zumindest qualitativ nachzuvollziehen. In Fällen, in denen das Netz zu komplex ist, um den gesamten Schlussfolgerungsprozess nachzuvollziehen, können meist noch in lokalen Teilbereichen des Modells, die weitestgehend in sich abgeschlossen sind, die lokalen Schlussfolgerungsvorgänge verstanden werden. Diese Eigenschaft Bayes'scher Netze wird von Erklärungskomponenten (siehe hierzu Abschnitt 2.1.7) ausgenutzt, die gegenüber den Benutzern das Zustandekommen der Systementscheidungen transparenter machen soll. So kann es für einen potentiellen Käufer in einem WWW-Shop von Interesse sein, weshalb er ein spezielles Produkt angeboten bekommt. Er kann dann möglicherweise den Wert der Empfehlung besser einordnen. Erscheint ihm die Argumentation des Systems plausibel, so wird er vermutlich eher dazu tendieren, das Produkt (z.B. ein neues Buch) zu kaufen, als in Situationen, in denen er nicht versteht, weshalb das System der Auffassung ist, warum er an diesem Produkt interessiert sein sollte.

Einbringen von Expertenwissen Mit Bayes'schen Netzen lässt sich vorhandenes Expertenwissen über eine Domäne in kompakter Form repräsentieren. Damit ist es auch in vielen Anwendungsszenarien benutzeradaptiver Systeme, in denen keine Daten erhoben werden können,—beispielsweise aus Gründen des Datenschutzes—möglich, Benutzermodelle zu erstellen, die die Grundlage der Adaptionsentscheidungen bilden. Es wurde eine Vielzahl von Methoden entwickelt, die im Rahmen der Konstruktion Bayes'scher Netze mit Domänenexperten angewendet werden können, um deren Wissen zu extrahieren und in entsprechender Form zu modellieren (siehe z.B. van der Gaag, Renoij, Witteman & Aleman, 1999).

Erweiterbarkeit zu probabilistischen relationalen Modellen Eine neue, vielversprechende Entwicklung im Zusammenhang mit Bayes'schen Netzen stellen *probabilistische relationale Modelle (PRMs)* (Koller & Pfeffer, 1998) dar. Sie basieren auf Ideen objekt-orientierter Ansätze und der relationalen Algebra, indem sie dort verwendete Organisationsstrukturen, wie z.B. Klassen mit ihren Attributen oder Relationen zwischen Klassen, auf den Kontext Bayes'scher Netze übertragen. Eine Klasse entspricht in einem PRM beispielsweise einem lokal verwalteten Bayes' schen Netz, das über definierte Schnittstellen mit den Netzen anderer Klassen interagieren kann. Mit PRMs ist es durch die objekt-orientierte Repräsentationsform möglich, komplexe Domänen zu modellieren, deren Handhabung mit „normalen" Bayes'schen Netzen schwierig würde.

Maschinelle Lernverfahren Es existieren maschinelle Lernverfahren für Bayes'sche Netze,[8] die es ermöglichen, gesammelte Daten zur Konstruktion und Pflege eines Benutzermodells in Form eines (teilweise) gelernten bzw. permanent aktualisierten Bayes'schen Netzes auszunutzen (siehe z.B. Lau & Horvitz, 1999; Albrecht, Zukerman & Nicholson, 1998). Allerdings gibt es nach Kenntnisstand des Autors bislang keine Ansätze, die sich explizit mit der Anpassung existierender Verfahren oder der Entwicklung neuer, speziell auf den Benutzermodellierungskontext zugeschnittener Methoden beschäftigen. Dies ist das übergeordnete Ziel der vorliegenden Arbeit.

1.3 Ziele

Obwohl Bayes'sche Netze in immer stärkerem Maße als Inferenzmechanismus in benutzeradaptiven Systemen eingesetzt werden und auch die üblichen Lerntechniken Verwendung finden, existieren bislang keine maschinelle Lernmethoden, die speziell auf die charakteristischen Eigenschaften des Benutzermodellierungskontextes zugeschnitten sind. Im Rahmen dieser Arbeit werden die entsprechenden Fragestellungen identifiziert und formuliert sowie verschiedene Lernverfahren für Bayes'sche Netze entwickelt, die in einer modularen, generischen Konzeption zum maschinellen Lernen Bayes'scher Netze in benutzeradaptiven Systemen integriert werden. Insbesondere werden die folgenden in der Benutzermodellierung relevanten Fragestellungen in dieser Arbeit im Vordergrund der Diskussion stehen (vgl. auch Wittig, 1999):

- *Wie können gute Benutzermodelle auf der Basis geringer Datenmengen gelernt werden?*

  Oft finden nur wenige Interaktionen zwischen dem Benutzer und dem benutzeradaptiven System statt. Im Extremfall interagiert ein Benutzer nur ein einziges Mal mit einem System, beispielsweise einem Online-CD-Shop. Ein allgemeines Problem besteht bei benutzeradaptiven Systemen in der Problematik wie ein neuer Benutzer, der zuvor noch nicht mit dem System in Kontakt war, behandelt werden soll. Auf welcher Basis sollen die Adaptionsentscheidungen getroffen werden? In solchen Situationen sind—wenn überhaupt—nur in sehr begrenztem Umfang Daten zu den Benutzern vorhanden, die als Eingabe für maschinelle Lernverfahren dienen können. In der vorliegenden Arbeit wird untersucht, inwieweit auch mit wenigen verfügbaren Lerndaten brauchbare Benutzermodelle in Form Bayes'scher Netze erlernt werden können. In manchen Domänen wie z.B. im Online-CD-Shop können Empfehlungen etwa auf der Basis eines allgemeinen Benutzermodells bestimmt werden, das Daten von vielen anderen Benutzern zusammenfasst statt nur die Daten des einzelnen Nutzers zu berücksichtigen. Auf dieser Grundlage können in vielen Fällen brauchbare Inferenzen

[8] Überblicke zum maschinellen Lernen Bayes'scher Netze geben Buntine (1996) und Heckerman (1998).

auch über das Verhalten, die Interessen usw. neuer Benutzer gezogen werden. Neben dem Einsatz solcher bekannten Ansätze zur Behandlung des Problems geringer Datenmengen, die auch im Zusammenhang mit anderen maschinellen Lernverfahren angewendet werden, sollen in dieser Arbeit neue, auf den Benutzermodellierungskontext fokussierte Methoden für Bayes'sche Netze entwickelt werden.

- *Wie können die potenziell großen individuellen Unterschiede zwischen den Benutzern erkannt und im Benutzermodell berücksichtigt werden?*

  Typischerweise sind benutzeradaptive Systeme so konzipiert, dass sie sich möglichst optimal an den einzelnen Benutzer anpassen, um ihm die Interaktion mit dem System zu erleichtern oder die für ihn aufbereitete Information zu vermitteln. Der Erfolg eines benutzeradaptiven Systems ist maßgeblich davon abhängig, inwieweit das System auf die individuellen Bedürfnisse, Vorlieben, Ziele usw. eingehen kann. Diese Fähigkeit des Systems ist wesentlich in der Modellierung der individuellen Unterschiede in seinen Benutzermodellen begründet. Deshalb spiegelt sich die Problematik der individuellen Unterschiede zwischen den Benutzern auch beim Einsatz maschineller Lernverfahren wider: Sind große Datenmengen einer Vielzahl von Benutzern für die Lernverfahren verfügbar, dann stellt sich die Frage, wie die individuellen Unterschiede in den Verhaltensweisen, Interessen, Zielen usw. in den Daten identifiziert werden können. Nur so ist es möglich, Benutzermodelle zu erlernen, die in der Lage sind, inter-individuelle Unterschiede auch zur Laufzeit des Systems zu erkennen und während des Interaktionsprozesses zu berücksichtigen.

- *Wie können zeitliche Veränderungen der Interessen bzw. Eigenschaften der Benutzer erkannt und berücksichtigt werden?*

  Während der Interaktion zwischen System und Benutzer können sich die Ziele oder Eigenschaften des Benutzers verändern. Im READY-Szenario kann sich z.B. die kognitive Ressourcenlage des Benutzers verändern. Besonders deutlich wird das Problem beim Lehr-/Lernsystem. Schließlich ist es gerade das Ziel des Systems, die Benutzereigenschaften zu verändern: Der Schüler soll sein Wissen verbessern. Solche Sachverhalte sollten im Benutzermodell und beim Erlernen desselben erkannt und berücksichtigt werden. Im weiteren Verlauf der Arbeit werden Verfahren vorgestellt, die Veränderungen der Benutzermodelle anhand von Interaktionsdaten automatisch erkennen können und entsprechende Anpassungen der Bayes'schen Netze vornehmen.

- *Wie kann die Interpretierbarkeit der gelernten Benutzermodelle gewährleistet bzw. verbessert werden?*

  Wie in Abschnitt 1.2 angeführt, existieren im Wesentlichen zwei Gründe für die Verwendung interpretierbarer Benutzermodelle: (a) die Vereinfachung des Konstruktionsprozesses und (b) die Erhöhung der Akzeptanz auf Benutzerseite. Bei der Entwicklung der in dieser Arbeit vorgestellten Verfahren wurde auf die Interpretierbarkeit der gelernten Modelle einer der Schwerpunkte gesetzt. Die existierenden Lernverfahren bieten nur in geringem Maß Möglichkeiten, die Interpretierbarkeit der Resultate zu verbessern bzw. zu gewährleisten.

- *Wie kann vorhandenes Hintergrundwissen über die zu modellierende Domäne in den Lernprozess eingebracht werden?*

In vielen Situationen sind bestimmte Aspekte des Benutzermodells bereits bekannt, bevor maschinelles Lernen durchgeführt wird. So kann meist von einem Domänenexperten relativ einfach (ein Teil der) kausalen Zusammenhänge zwischen den betrachteten Variablen sowie die Qualität der Zusammenhänge spezifiziert werden. Beispielsweise besteht ein positiver kausaler Zusammenhang zwischen der kognitiven Belastung des Benutzers und seiner Anfälligkeit für Fehler bei der Ausführung einer Aufgabe, d.h., ist der Benutzer kognitiv belastet, dann erhöht sich die Wahrscheinlichkeit, dass er einen Fehler begeht. Es wird untersucht, inwieweit es möglich ist und ob es sich lohnt, solche Informationen in den Lernprozess mit einzubringen.

- *Wie können unterschiedliche Arten von Daten im Lernprozess gemeinsam genutzt werden?*

  Es existieren zumindest zwei Arten von Daten, die von besonderer Bedeutung in benutzeradaptiven Systemen sind: (a) *Experimentaldaten* und (b) *Gebrauchsdaten*. Experimentaldaten werden in kontrollierten Umgebungen, wie z.B. bei der Durchführung von psychologischen Experimenten, gesammelt. In dieser Weise erhobene Daten zeichnen sich typischerweise durch eine hohe Qualität aus, d.h., man ist beispielsweise durch die Kontrolle über die Experimentalsituation in der Lage, die Werte der Variablen von Interesse in allen Fällen zu beobachten. Außerdem existieren bei der Erhebung solcher Daten typischerweise weniger störende Einflüsse der Umgebung, so dass die gesammelten Daten weniger Rauschen aufweisen. Andererseits spiegeln Experimentaldaten oftmals nicht die typische Anwendungsituation wider. Im Gegensatz dazu bestehen Gebrauchsdaten, die beim Einsatz eines Systems in der Anwendungssituation erhoben werden, oft aus unvollständigen Datensätzen, da die Werte einzelner Variablen in bestimmten Situationen nicht beobachtet oder aufgezeichnet werden (können). Durch die Einflüsse der nicht kontrollierbaren Umgebung erhält man oft verrauschte Daten, was zu schlechten Lernergebnissen führen kann.

- *Wie können die kausalen Zusammenhänge verschiedener Aspekte der Benutzermodelle mit Hilfe maschineller Lernverfahren ermittelt werden?*

  Beim Erlernen von Benutzermodellen in Form Bayes'scher Netze wurde sich bislang auf die Behandlung einer der beiden Teilaufgaben des Lernproblems fokussiert—das Erlernen der bedingten Wahrscheinlichkeiten bei Vorgabe der kausalen Beziehungen zwischen den modellierten Variablen (siehe hierzu Abschnitt 2.6). In der vorliegenden Arbeit soll untersucht werden, ob und gegebenenfalls mit welchen Verfahren die zweite Lernaufgabe—das so genannte *Strukturlernen* Bayes'scher Netze—im Benutzermodellierungskontexts mit Erfolg eingesetzt werden kann.

- *Wie kann das aus dem maschinellen Lernen bekannte Problem des „Overfitting" vermieden oder zumindest verringert werden?*

  Das *Übertraining* bzw. *Overfitting* ist ein sehr häufig beobachteter Effekt bei der Anwendung maschineller Lernverfahren. Es bedeutet, dass sich das gelernte Modell im Verlauf des Lernprozesses (zu) stark auf die verwendeten Lerndaten spezialisiert hat. Ein solches Lernergebnis besitzt also nur eine eingeschränkte Generalisierbarkeit hinsichtlich neuer Daten. Im Fall der Benutzermodellierung führt Overfitting beispielsweise dazu, dass das erlernte Benutzermodell zwar gute Vorhersagen über die (oder sehr ähnliche) Benutzer liefern kann, die in den Lerndaten vertreten waren, aber neue Benutzer mit eventuell leicht variierenden

Eigenschaften bereiten ihm Probleme. Dies spielt insbesondere dann eine wichtige Rolle, wenn es sich um Domänen handelt, in denen die Benutzereigenschaften sehr heterogen sind und beispielweise keine Stereotypen identifiziert werden können, die in der Lage sind, alle möglichen Benutzer abzudecken. Ein weiteres Beispiel des Overfitting-Effekts in benutzeradaptiven Systemen sind wechselnde Einsatzkontexte des Benutzermodells. Wird ein Modell in einer gegebenen Situation erlernt und anschließend in einem leicht variierenden Kontext eingesetzt, so kann die Spezialisierung auf die Lernsituation zu einer verminderten Performanz in der neuen Einsatzsituation führen.

Ein weiteres konkretes Ziel, das mit dieser Arbeit verfolgt wird, ist die Verbesserung der empirischen Fundierung der im READY-Prototyp verwendeten Methoden der Benutzermodellierung. Die bisher eingesetzten Benutzermodelle wurden im Wesentlichen manuell auf der Basis theoretischer Überlegungen im Zusammenspiel mit relevanten psychologischen Forschungsergebnissen spezifiziert (siehe Schäfer, 1998 und Großmann-Hutter, Jameson & Wittig, 1999).

Obwohl die vorliegende Arbeit und die in ihr entwickelten Verfahren aus dem Forschungsgebiet der Benutzermodellierung heraus motiviert sind, ist ihre Anwendung nicht auf dieses Gebiet beschränkt. Insgesamt ergibt sich ein Anwendungspotenzial für die (einzelnen) Methoden in Bereichen, in denen Bayes'sche Netze eingesetzt werden und in denen die gleichen bzw. ähnliche Probleme auftreten. Das Lernen mit wenigen Daten, die Modellierung von Unterschieden der betrachteten Objekte, die Anpassung der erlernten Modelle an beobachtete Veränderungen, die Interpretierbarkeit, das Einbringen von Hintergrundwissen in den Lernprozess und die Behandlung des Overfitting-Effekts sind Fragestellung von allgemeinem Interesse beim maschinellen Lernen—und damit auch im Zusammenhang mit dem Erlernen Bayes'scher Netze.

1.4 Gliederung

In **Kapitel 2** wird die Definition Bayes'scher Netze und die zugehörige Notation eingeführt. Anhand zweier Szenarien aus dem READY-Projekt werden die Anwendungsmöglichkeiten Bayes'scher Netze in benutzeradaptiven Systemen veranschaulicht. Dabei werden Arbeiten aus dem Umfeld des Forschungsgebiets der Bayes'schen Netze vorgestellt, die von besonderer Relevanz beim Einsatz im Benutzermodellierungskontext sind. Es handelt sich dabei um Verbalisierungsmethoden zur Erklärung der Schlussfolgerungsprozesse, die Erweiterung zu Einflussdiagrammen, dynamische Bayes'sche Netze zur Modellierung temporaler Aspekte und objekt-orientierte Ansätze der Konstruktion und Verwaltung Bayes'scher Netze. Ein ausführlicher Überblick über den Stand der Forschung benutzeradaptiver Systeme auf der Basis Bayes'scher Netze mit besonderem Augenmerk bezüglich des Einsatzes von maschinellen Lernverfahren beschließt dieses einführende Kapitel.

Kapitel 3 dient der Diskussion und der Übertragung des allgemeinen maschinellen Lernproblems auf den Kontext benutzeradaptiver Systeme—unabhängig von potenziell einsetzbaren Verfahren. Es werden Problemstellungen identifiziert und diskutiert, die typischerweise die direkte Anwendung maschineller Standardlerntechniken in Szenarien benutzeradaptiver Systeme verhindern können. Anschließend werden existierende generische Benutzermodellierungsumgebungen hinsichtlich der Integration maschineller Lernverfahren untersucht, gefolgt von einer kompakten Einführung in kollaborative und inhaltlich-basierte Methoden. Den Abschluss dieses Kapitels bildet eine Diskussion von erfolgreich in benutzeradaptiven Systemen eingesetzten maschinellen Lernverfahren.

Die der vorliegenden Arbeit zugrunde liegende Konzeption zum maschinellen Lernen Bayes' scher Netze für benutzeradaptive Systeme wird in **Kapitel 4** vorgestellt. Es schließt sich ein Überblick über die existierenden maschinellen Lernverfahren Bayes'scher Netze an, die im weiteren Verlauf der Arbeit den Ausgangspunkt der Erweiterung bzw. der Entwicklung neuer, speziell auf den Benutzermodellierungskontext ausgerichteten Verfahren bilden.

In **Kapitel 5** wird ein neues Verfahren zum Erlernen interpretierbarer bedingter Wahrscheinlichkeiten in Bayes'schen Netzen vorgestellt. Grundlage bilden die Standardlernverfahren Bayes' scher Netze, die um die Funktionalität erweitert werden, qualitative Information in den Lernprozess einzubringen. Die Methode wird empirisch sowohl anhand synthetisch erzeugter Daten als auch mit Daten der beiden READY-Szenarien evaluiert.

Kapitel 6 beinhaltet zwei Schwerpunkte: Einerseits wird ein neues Verfahren entwickelt, um individuelle Unterschiede zwischen Benutzern in Bayes'schen-Netz-Benutzermodellen zur Laufzeit zu erkennen und in den Modellen zu berücksichtigen, andererseits wird dieses Verfahren ausführlich im Rahmen eines Vergleichs mit alternativ einsetzbaren Vorgehensweisen in den beiden READY-Szenarien untersucht—auch hinsichtlich praktischer Kriterien, die bei der Entscheidung zur Auswahl einer der Alternativen eine Rolle spielen.

Nachdem in den beiden Kapiteln 5 und 6 der wichtige Fall des Erlernens der bedingten Wahrscheinlichkeiten im Fokus gestanden hat, wird in **Kapitel 7** der Strukturfall, d.h., das Erlernen der vorhandenen direkten kausalen Zusammenhänge zwischen den modellierten Aspekten der Domäne untersucht. Es werden Studien vorgestellt, die sich mit dem potenziellen Mehrwert des Strukturlernens Bayes'scher Netze in benutzeradaptiven Systemen beschäftigen. Weiterer zentraler Punkt des Kapitels ist die Präsentation und Evaluation eines im Rahmen der vorliegenden Arbeit entwickelten Verfahrens zur Adaption der Struktur Bayes'scher Netze an Veränderungen der Systemumgebung bzw. der Benutzereigenschaften.

Kapitel 8 beschließt die Arbeit mit einer Zusammenfassung der erzielten Ergebnisse und spricht offene Fragestellungen an, die sich aus der vorliegenden Arbeit ergeben und in weiteren Arbeiten untersucht werden können.

2 BAYES'SCHE NETZE ALS INFERENZMECHANISMUS IN BENUTZERADAPTIVEN SYSTEMEN

In diesem Kapitel wird die formale Definition Bayes'scher Netze eingeführt und dieses Werkzeug zur Repräsentation und Behandlung von Unsicherheit in intelligenten Systemen anhand von Beispielen erläutert. Nach der Vorstellung der Grundidee des Schlussfolgerns mit Bayes'schen Netzen werden empirische Studien des READY-Szenarios besprochen und mit Bayes'schen Netzen beispielhaft modelliert. Diese Modelle werden im weiteren Verlauf dieser Arbeit zur Illustration und Evaluation der entwickelten Lernverfahren verwendet. Die Erweiterung Bayes'scher Netze zum entscheidungstheoretischen Hilfsmittel der Einflussdiagramme wird anhand dieser Beispielszenarien beschrieben. Nach einer kompakten Einführung in die Thematik dynamischer Bayes'scher Netze zur Behandlung temporaler Aspekte der zu modellierenden Domänen folgt ein ausführliches Anwendungsbeispiel in Form einer empirisch basierten Studie zur Erkennung kognitiver Ressourcenbeschränkungen eines Systembenutzers anhand von Symptomen seiner gesprochenen Sprache. Anschließend werden objekt-orientierte Ansätze diskutiert, die einen Einsatz Bayes'scher Netze in komplexen Domänen ermöglichen sollen. Insbesondere werden hier probabilistische relationale Modelle betrachtet. Den Abschluss dieses einführenden Kapitels bildet eine Diskussion neuerer Forschungsprototypen benutzeradaptiver Systeme, die Bayes'sche Netze als Inferenzmechanismus nutzen.

2.1 Bayes'sche Netze

Bayes'sche Netze (Pearl, 1988)[1] stellen einen Ansatz der Repräsentation probabilistischer Zusammenhänge mit graphischen Modellen dar. Deshalb werden im Folgenden einige Begriffe und Notationen aus dem Bereich der Graphen- und der Wahrscheinlichkeitstheorie in kompakter Form eingeführt, um auf dieser Basis anschließend in der Lage zu sein, die formale Definition Bayes'scher Netze vorzustellen und zu erläutern. Für ausführliche Einführungen in die beiden genannten Themengebiete wird auf die entsprechende Standardliteratur verwiesen (eine kompakte Einführung, die die in dieser Arbeit benötigten Begriffe und Inhalte abdeckt, bieten die Anhänge A bzw. B von Beierle & Kern-Isberner, 2000).

[1] Weitere Einführungen in die Thematik Bayes'scher Netze bieten u.a. Neapolitan (1990), Jensen (1996), Castillo, Gutierrez und Hadi (1997) sowie Jensen (2001).

2.1.1 Grundlegende Begriffe

Ein *gerichteter Graph* $G = (\boldsymbol{V}, \boldsymbol{E})$ ist ein Paar bestehend aus einer Menge von n *Knoten* $\boldsymbol{V} = \{v_1, \ldots, v_n\}$ und einer Mengen von m *gerichteten Kanten* $\boldsymbol{E} = \{e_1, \ldots, e_m\}, e_{ij} = (v_i, v_j), v_i, v_j \in \boldsymbol{V}$. Ein *Pfad* ist eine Sequenz von Knoten $(w_1, w_2, \ldots, w_k), k \geq 2$, so dass $(w_{i-1}, w_i) \in \boldsymbol{E}$ für $2 \leq i \leq k$. Ein *Zyklus* ist ein Pfad, der mit demselben Knoten sowohl beginnt als auch in ihm endet. Ein *azyklischer gerichteter Graph* (engl. *directed acyclic graph, DAG*) ist ein Graph ohne einen solchen Zyklus. v_i ist ein *Elternteil* von v_j, genau dann, wenn es eine Kante $(v_i, v_j) \in \boldsymbol{E}$ gibt. Die Menge aller Elternteile eines Knotens v_j wird mit $\boldsymbol{Pa}(v_j)$ bezeichnet. v_j ist ein *Kind* von v_i. Ein Knoten v_i ist ein *Nachfolger* eines Knotens v_j, wenn ein Pfad von v_j nach v_i existiert.

Bayes'sche Netze dienen der kompakten Repräsentation einer *gemeinsamen Wahrscheinlichkeitsverteilung* $P(X_1, \ldots, X_n)$[2] über einer Menge $\boldsymbol{X}$ von *Zufallsvariablen* $X_1, \ldots, X_n$. In dieser Arbeit werden nur *diskrete* Zufallsvariablen betrachtet. Das bedeutet, jede dieser Variablen X_i besitzt eine endliche Anzahl n_i sich gegenseitig ausschließender *Zustände* $\boldsymbol{x_i} = \{x_{i1}, \ldots, x_{in_i}\}$, die den gesamten Wertebereich *Val*(X_i) der Zufallsvariable komplett überdecken. Eine *bedingte Wahrscheinlichkeit* $P(A_1 = a_1, \ldots, A_s = a_s \mid B_1 = b_1, \ldots, B_m = b_m)$ oder kurz $P(a_1, \ldots, a_s \mid b_1, \ldots, b_m)$ ist definiert als $P(a_1, \ldots, a_s, b_1, \ldots, b_m)/P(b_1, \ldots, b_m)$, für $P(b_1, \ldots, b_m) > 0$. Dabei ist $A = a$ oder kurz a die Schreibweise dafür, dass die Zufallsvariable A ihren Zustand a annimmt. Zwei Zufallsvariablen X, Y sind *unabhängig*, wenn bezüglich ihrer gemeinsamen Wahrscheinlichkeitsverteilung $P(X, Y) = P(X)P(Y)$ gilt. Analog heißen zwei Variablen X, Y *bedingt unabhängig* bezüglich einer Menge von Zufallsvariablen $\boldsymbol{Z}$, wenn gilt $P(X, Y \mid \boldsymbol{Z}) = P(X \mid \boldsymbol{Z})P(Y \mid \boldsymbol{Z})$ für $P(\boldsymbol{Z}) > 0$.

2.1.2 Definition

Soll eine gemeinsame Wahrscheinlichkeitsverteilung einer größeren Anzahl von Variablen verwaltet werden, so stößt man bei expliziter Repräsentation durch Angabe einer Wahrscheinlichkeit pro Zustandskombination schnell an die Grenze des praktisch Machbaren. Beispielsweise müssen im Fall von 20 binären Variablen, d.h., 20 Variablen mit je zwei Zuständen, bereits $2^{20} = 1.048.576$ Einzelwerte spezifiziert werden. Durch Ausnutzen (bedingter) Unabhängigkeiten zwischen Variablen der zu modellierenden Domäne kann die benötigte Anzahl der anzugebenden Werte oft auf eine handhabbare Größe verringert werden. Einen solchen Ansatz stellen Bayes'sche Netze dar:

Definition 2.1 (Bayes'sches Netz) *Ein* Bayes'sches Netz $B = (G, \boldsymbol{\theta})$ *für eine Menge* $\boldsymbol{X} = \{X_1, \ldots, X_n\}$ *von Zufallsvariablen besteht aus zwei Teilen:*

1. *Einem gerichteten azyklischen Graphen* $G = (\boldsymbol{X}, \boldsymbol{E})$*, dessen Knoten den Zufallsvariablen entsprechen*[3] *und mit dessen Kanten die bedingten Unabhängigkeiten zwischen den Variablen kodiert werden. Man spricht von* G *als der* Struktur *von* B.

2. *Einer Menge* $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n\}$ *von mit den Variablen assoziierten* Tabellen bedingter Wahrscheinlichkeiten *(engl.* conditional probability tables, CPTs*)* $\boldsymbol{\theta}_i = P(X_i \mid \boldsymbol{Pa}(X_i))$,

[2] Das Symbol P wird in dieser Arbeit—wie auch in der entsprechenden Literatur üblich—sowohl für Punktwahrscheinlichkeiten als auch für Wahrscheinlichkeitsverteilungen verwendet. Die jeweilige Bedeutung ergibt sich aus dem Kontext.

[3] Aus diesem Grund werden im weiteren Verlauf dieser Arbeit die beiden Begriffe 'Knoten' und 'Variable' austauschbar verwendet. Aus dem Kontext wird wiederum ersichtlich sein, welche der beiden Bedeutungen im Detail gemeint ist.

$i = 1, \ldots, n$. Sie beinhalten als Einträge die bedingten Wahrscheinlichkeiten $\theta_{ijk} = P(x_{ij} \mid pa_k(X_i))$ der n_i Zustände $x_{ij}, j = 1, \ldots, n_i$, der Variablen X_i konditioniert auf die möglichen Zustandskombinationen $\boldsymbol{pa}(X_i)$ der Eltern $\boldsymbol{Pa}(X_i)$. Mit $pa_k(X_i)$ wird die k-te der Zustandskombination $\boldsymbol{pa}(X_i)$ der Eltern bezeichnet. Besitzt ein Knoten keine Eltern, dann beinhaltet seine CPT unbedingte A-priori-Wahrscheinlichkeiten $P(x_{ij})$, d.h., $\boldsymbol{\theta}_i = P(X_i)$.

Die bedingten Unabhängigkeiten zwischen Variablen einer Domäne werden in der Struktur des Netzes durch das folgende Unabhängigkeits-Kriterium kodiert:

Satz 2.1 (Unabhängigkeits-Kriterium Bayes'scher Netze) *Sind die Zustände der Elternvariablen $\boldsymbol{Pa}(X_i)$ bekannt, dann ist eine Variable X_i unabhängig von den Zuständen ihrer Nicht-Nachfolger in der Struktur des Bayes'schen Netzes.*

Weitere dadurch induzierte Unabhängigkeitsannahmen können unter Anwendung des *d-Separationskriteriums* (Pearl, 1988) aus der Struktur abgelesen werden. Eine ausführliche Erläuterung diese Kriteriums und seiner Anwendung zur Identifikation der bedingten Unabhängigkeiten zwischen Variablen des Bayes'schen Netzes findet sich bei Russell und Norvig (1995). Obwohl es eine zentrale Bedeutung im Zusammenhang mit dem Schlussfolgerungsprozess einnimmt, spielt es bei der Anwendung maschineller Lernverfahren lediglich eine untergeordnete Rolle und wird im weiteren Verlauf dieser Arbeit aus diesem Grund nur am Rande betrachtet, weshalb auf eine detaillierte Diskussion an dieser Stelle verzichtet wird.

Für den weiteren Verlauf der Arbeit wird eine *kausale Interpretation der Kanten* zugrunde gelegt: Man nimmt an, dass die Kanten des Bayes'schen Netzes *direkte* kausale (probabilistische) Zusammenhänge zwischen den entsprechenden Variablen repräsentieren, d.h., dass die Elternvariablen direkte kausale Einflüsse auf ihre Kindvariablen besitzen. Es gilt zu beachten, dass die Definition eines Bayes'schen Netzes nichts über kausale Beziehungen aussagt, sie basiert auf der Modellierung (bedingter) *Un*abhängigkeiten.

Es hat sich gezeigt, dass diese häufig bei der Konstruktion Bayes'scher Netze angewendete Heuristik in den meisten praktisch relevanten Anwendungssituationen zu einer Struktur führt, welche die bedingten Unabhängigkeiten im Sinne des Unabhängigkeitskriteriums bzw. d-Separationskriteriums widerspiegelt (siehe z.B. Heckerman, 1998).[4] Durch die Anwendung dieser Heuristik wird sowohl die Interpretation der Netz-Strukturen als auch der manuelle Konstruktionsprozess vereinfacht, da es oft—nicht nur für Experten—einfach ist, die kausalen Zusamenhänge bzw. die kausale Struktur der Domäne zu spezifizieren.

In einem Bayes'schen Netz wird die gemeinsame Wahrscheinlichkeitsverteilung folgendermaßen kompakt als Produkt lokaler Wahrscheinlichkeitsverteilungen (in Form der CPTs) repräsentiert:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid \boldsymbol{Pa}(X_i)) = \prod_{i=1}^{n} \boldsymbol{\theta}_i. \tag{2.1}$$

Damit genügt es, zur Repräsentation der gemeinsamen Wahrscheinlichkeitsverteilung, die bedingten Wahrscheinlichkeiten θ_{ijk} der CPTs $\boldsymbol{\theta}$ zu verwalten—einer in den meisten Anwendungsdomänen deutlich geringeren Anzahl an Einzelwerten im Vergleich zur expliziten Speicherung der unbedingten Wahrscheinlichkeiten aller Zustandskombinationen. Hinsichtlich der Anwendung

[4] Einen theoretisch fundierten Ansatz, um zu entscheiden, ob die Struktur eines Bayes'schen Netzes ein kausales Modell eines Systems widerspiegelt, beschreiben Druzdzel und Simon (1993).

maschineller Lernverfahren ist zu bemerken, dass somit auch nur diese geringere Anzahl an (bedingten) Wahrscheinlichkeiten erlernt werden muss. Eine geringere Menge an freien Parametern eines Lernprozesses kann auf der Basis der gleichen Datenmenge im Normalfall robuster erlernt werden, als eine größere, was typischerweise zu einer Erhöhung der Qualität des Lernergebnisses führt.

Anhand des folgenden Beispiels, das bereits in Kapitel 1 kurz angesprochen wurde, werden die eingeführten Begriffe und die Definition eines Bayes'schen Netzes sowie—im anschließenden Abschnitt—der Prozess des Schlussfolgerns mit Bayes'schen Netzen verdeutlicht.

2.1.3 Beispiel: Hypothetisches Bayes'sches Netz eines adaptiven Lehr-/Lernsystems

Ein einfaches adaptives Lehr-/Lernsystem benutzt ein Bayes'sches Netz, um die Anpassung der Lehrstrategie an die Fähigkeiten des Lernenden durchführen zu können. Folgende Variablen spielen dabei eine Rolle: Das VORWISSEN (V) des Lernenden, das durch vorangestellte Tests ermittelt werden kann, die ANZAHL DER (durchgeführten) ÜBUNGEN (Ü) innerhalb des Kurses, die SCHWIERIGKEIT DER KLAUSUR (S), die erzielte KLAUSURNOTE (N) und das WISSENSNIVEAU (W) des Lernenden. In diesem einfachen Beispiel sollen alle Variablen binär sein. Abbildung 2.1 zeigt eine mögliche Struktur des zu verwendeten Bayes'schen Netzes. Es modelliert die direkten kausalen Einflüsse der beiden Variablen VORWISSEN und ANZAHL DER ÜBUNGEN auf WISSEN. Diese Variable wiederum hat einen direkten kausalen Einfluss auf die KLAUSURNOTE. Daneben wird die KLAUSURNOTE von der SCHWIERIGKEIT DER KLAUSUR beeinflusst. Wie häufig ist es auch in dieser Domäne recht einfach, die kausale Struktur zu spezifizieren, indem man die kausale Interpretation der Kanten zugrunde legt. Etwas schwieriger ist es, den quantitativen Teil des Bayes'schen Netzes, die CPTs festzulegen. Tabelle 2.1 zeigt beispielhaft zwei CPTs anhand der beiden Variablen VORWISSEN und WISSEN. Ihnen liegen die qualitativen Annahmen zugrunde, dass sowohl ein erhöhtes Vorwissen als auch eine größere Anzahl an durchgeführten Übungen üblicherweise zu einem höheren Wissensniveau des Lernenden führen. Weiterhin führen ein höheres Wissensniveau und eine geringere Schwierigkeit der Klausur im Normalfall zu einer besseren Klausurnote (nicht als CPT dargestellt). Analog gelten die umgekehrten probabilistischen Zusammenhänge (z.B. bewirkt eine geringere Anzahl an durchgeführten Übungen normalerweise ein geringeres Wissensniveau).

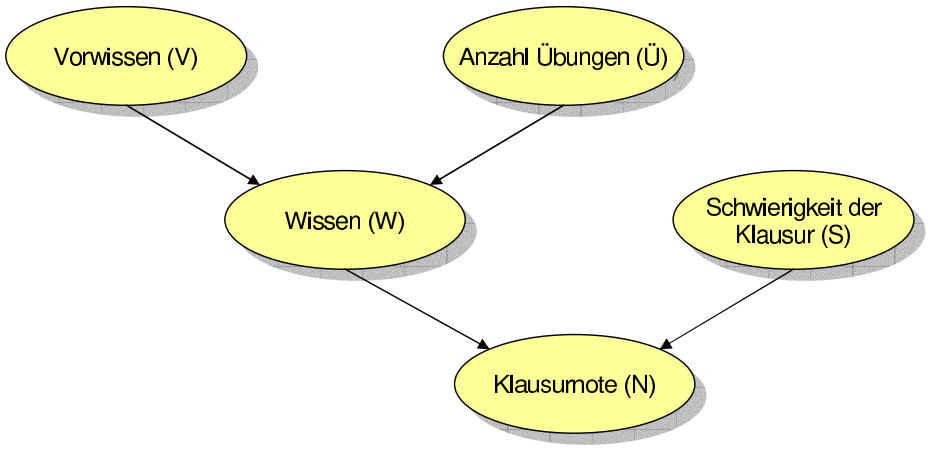


***Abbildung 2.1**: Beispiel eines Bayes'schen Netzes*

An diesem Beispiel lässt sich auch die Kodierung der bedingten Unabhängigkeitsaussagen nach Satz 2.1 erläutern. Beispielsweise ist hier die Variable KLAUSURNOTE bedingt unabhängig von ANZAHL DER ÜBUNGEN bei gegebenem Wert für WISSEN. Damit wird die Aussage repräsentiert, dass es z.B. bei bekannt hohem Wissensniveau egal ist, wie viele Übungen der Lernende bearbeitet hat, um zu einer Einschätzung der Wahrscheinlichkeit einer guten Klausurnote zu kommen. Man geht in der vorliegenden Modellierung davon aus, dass ein hohes Wissensniveau zu einer hohen Wahrscheinlichkeit einer guten Klausurnote führt, unabhängig von der Anzahl der bearbeiteten Übungen.

VORWISSEN:

hoch	0.30
gering	0.70

WISSEN:

ANZAHL DER ÜBUNGEN	hoch		niedrig	
VORWISSEN	hoch	gering	hoch	gering
hoch	0.90	0.80	0.60	0.10
gering	0.10	0.20	0.40	0.90

***Tabelle 2.1**: CPTs der beiden Variablen* VORWISSEN *und* WISSENSNIVEAU *des Bayes'schen Netzes aus Abbildung 2.1*

Obwohl man oft bekanntes qualitatives Wissen dieser Art ausnutzen kann, um die prinzipiellen probabilistischen Zusammenhänge festzulegen, ist es immer noch eine schwierige und mühsame Aufgabe, sich für die definitiven Werte der bedingten Wahrscheinlichkeiten der CPTs zu entscheiden (z.B. 0.85 oder 0.80?). Dies gilt insbesonderen in komplexen Domänen mit vielen Variablen. Aus diesem Grund wurden spezielle Verfahren entwickelt, um die Spezifikation der Werte durch Experten zu erleichtern (siehe z.B. von Winterfeldt & Edwards, 1986; Morgan & Henrion, 1990; van der Gaag et al., 1999). Dennoch bleibt die manuelle Konstruktion der CPTs eine aufwendige und fehleranfällige Aufgabe (Kahneman, Slovic & Tversky, 1982; Druzdzel & van der Gaag, 2000).

2.1.4 Beispiel: Naiver Bayes'scher Klassifizierer

Der für die Bearbeitung von Klassifikationsaufgaben eingesetzte *naive Bayes'sche Klassifizierer* (Duda & Hart, 1973) stellt eine strukturell sehr einfache Variante eines Bayes'schen Netzes dar. Er besitzt eine ausgezeichnete elternlose Variable, deren Zustände die alternativen Klassen repräsentieren. Diese Variable besitzt als Kinder *Merkmale* (engl. *features*), die charakteristisch für die Klassenzugehörigkeit der zu klassifizierenden Objekte sind. Abbildung 2.2 zeigt einen prototypischen naiven Bayes'schen Klassifizierer. Man sieht, dass ihm die Annahme zugrunde liegt, dass die Merkmale bei bekannter Klassenzugehörigkeit gegenseitig bedingt unabhängig sind (in der Struktur durch das Fehlen von Kanten zwischen den Merkmalsvariablen modelliert). Vorteil des naiven Bayes'schen Klassifizierers ist es, dass die Berechnung der Wahrscheinlichkeitsverteilung zur Klassenvariable anhand der beobachteten Merkmale sehr einfach—ohne aufwendige Inferenzverfahren für Bayes'sche Netze—möglich ist.

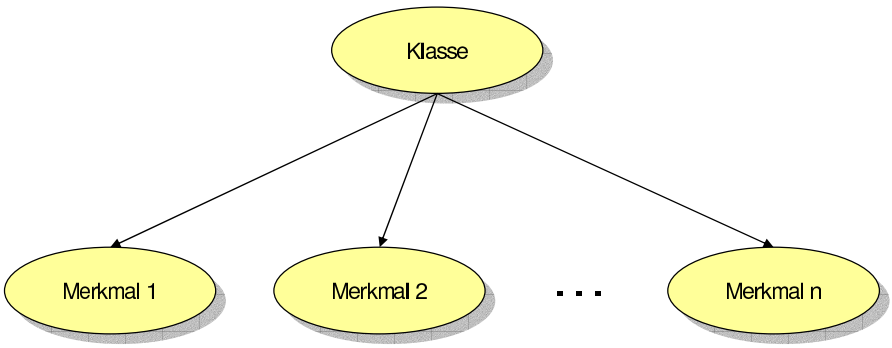


***Abbildung 2.2**: Naiver Bayes'scher Klassifizierer (Prototypische Darstellung)*

Nachdem an zwei Beispielen illustriert wurde, wie eine mit Unsicherheit behaftete Domäne als Bayes' sches Netz repräsentiert wird, stellt sich nun die Frage, wie ein solches Modell genutzt werden kann, um probabilistische Schlussfolgerungen über bestimmte Sachverhalte von Interesse zu ziehen.

2.1.5 Inferenzverfahren

Aufgabe von *Inferenzverfahren* für Bayes'sche Netze ist es—ähnlich wie beim menschlichen Schlussfolgerungsprozess—anhand partieller Beobachtungen in der betrachteten Domäne, den so genannten *Evidenzen,* Aussagen über andere Teile der Domäne zu machen. Im Fall Bayes' scher Netze bedeutet dies konkret, dass man Evidenzen zu Variablen des Modells hat, d.h., dass diese Variablen einen bestimmten Zustand annehmen, und man auf dieser Grundlage Wahrscheinlichkeiten für die Zustände anderer, nicht beobachteter Variablen konditioniert auf die gegebenen Evidenzen ermitteln möchte. Man spricht in diesem Zusammenhang von der *Interpretation der Evidenzen*. Das System aus Beispiel 2.1.3 könnte die Wahrscheinlichkeit eines hohen Wissensniveaus eines Klausurteilnehmers einschätzen wollen, der eine gute Note erzielt hat: Die Evidenz in Form der erzielten guten Note wird dann durch Interpretation innerhalb des Inferenzprozesses mit dem dazugehörigen Netz zur Bestimmung einer Wahrscheinlichkeitsverteilung über den Zuständen des Knotens zur Modellierung des Wissensniveaus eingebracht.

Die Lösung basiert auf der geschickten Kodierung der gemeinsamen Wahrscheinlichkeitsverteilung einer Menge von Variablen in einem Bayes'schen Netz. Im Prinzip kann eine Wahrscheinlichkeit für jede beliebige Zustandskombination einer Teilmenge dieser Variablen durch „Heraussummieren", dem so genannten *Marginalisieren,* aus der expliziten Repräsentation berechnet werden. Dazu werden alle Wahrscheinlichkeiten derjenigen Zustandskombinationen, in denen die Variablen von Interesse die entsprechenden Zustände annehmen, addiert. Will man in Beispiel 2.1.3 $P(N = \mathit{gut}, W = \mathit{hoch})$ ermitteln, so muss folgende Summe berechnet werden:

$$P(N = \mathit{gut}, W = \mathit{hoch}) = \sum_{s,\ddot{u},v} P(S = s, \ddot{U} = \ddot{u}, V = v, N = \mathit{gut}, W = \mathit{hoch}). \qquad (2.2)$$

Analog lässt sich z.B. auch $P(N = \mathit{gut})$ bestimmen.

Da bedingte Wahrscheinlichkeiten als Quotienten aus den Wahrscheinlichkeiten zweier Zustandskombinationen definiert sind, können beliebige bedingte Wahrscheinlichkeiten in dieser

Weise auf Basis der gemeinsamen Wahrscheinlichkeitsverteilung ermittelt werden. Im betrachteten Beispiel bedeutet dies, dass man die bedingten Wahrscheinlichkeit eines hohen Wissensniveaus eines Lernenden, der eine gute Klausur geschrieben hat, unter Verwendung von Gleichung 2.2 folgendermaßen bestimmen kann:

$$P(W = \textit{hoch} \mid N = \textit{gut}) = \frac{P(N = \textit{gut}, W = \textit{hoch})}{P(N = \textit{gut})}. \tag{2.3}$$

Diese prinzipielle Vorgehensweise wird aber im Fall einer größeren Anzahl von Variablen unpraktikabel, da die Anzahl der Summanden beim Marginalisieren in solchen Situationen exponentiell zunimmt. Dieses Problem kann durch Ausnutzen der in der Struktur des Bayes'schen Netzes kodierten bedingten Unabhängigkeiten reduziert werden. Anstelle gemeinsamer Wahrscheinlichkeiten werden als Summanden die Produkte der (bedingten) Wahrscheinlichkeiten der CPTs genutzt. Im Beispiel sieht dies in einem ersten Schritt wie folgt aus:

$$P(W = \textit{hoch} \mid N = \textit{gut}) = \frac{P(N = \textit{gut} \mid W = \textit{hoch})P(W = \textit{hoch})}{P(N = \textit{gut})}. \tag{2.4}$$

In weiteren Schritten müssen noch die verbleibenden—noch nicht elementaren—bedingten Wahrscheinlichkeiten durch (Summen von) Produkten (bedingter) Wahrscheinlichkeiten aus den CPTs des Bayes'schen Netzes ersetzt werden, z.B.:

$$\begin{aligned} P(N = \textit{gut} \mid W = \textit{hoch}) = {} & P(N = \textit{gut} \mid W = \textit{hoch}, S = \textit{hoch})P(S = \textit{hoch}) \\ & + P(N = \textit{gut} \mid W = \textit{hoch}, S = \textit{niedrig})P(S = \textit{niedrig}). \end{aligned} \tag{2.5}$$

Ein solches Vorgehen ist das Grundprinzip einer Vielzahl von entwickelten Inferenzverfahren für Bayes'sche Netze. Es existieren sowohl exakte Verfahren (Pearl, 1988; Jensen, 1996) als auch approximative Methoden (siehe Jordan, 1998), die für sehr komplexe Netze[5] geeignet sind.

2.1.6 Alternative Methoden zur Unsicherheitsbehandlung

Die beiden neben Bayes'schen Netze am häufigsten verwendeten Methoden zur Behandlung von Unsicherheit in benutzeradaptiven Systemen sind die *Dempster-Shafer-Theorie* (siehe z.B. Bauer, 1996) und *Fuzzy Logik* (siehe z.B. Chin, 1989).[6] Ihre Grundideen und einige Aspekte ihrer Anwendung im Benutzermodellierungskontext werden im Folgenden kurz vorgestellt.

2.1.6.1 Dempster-Shafer-Theorie

Der Einsatz der Dempster-Shafer-Theorie bietet sich in Szenarien an, in denen man im Schlussfolgerungsprozess Teilevidenzen (aus verschiedenen Quellen) verarbeiten muss. Eine typisches Beispiel einer solchen Teilevidenz ist etwa die Aussage „Ich denke, ich kenne die Person, die Sie im Sinn haben; wenn sie tatsächlich diejenige ist, die ich meine, dann ist sie soweit ich mich erinnere kein unerfahrener Benutzer." Diese Information kann zwar von gewissem Nutzen sein,

[5] 'Komplex' umfasst in diesem Zusammenhang nicht nur die Anzahlen und die Zustände der Variablen der Netze, sondern auch die strukturelle Eigenschaft, ob und gegebenenfalls wie viele ungerichtete Schleifen in den Strukturen auftreten (siehe Pearl, 1988).

[6] Jameson (1996) führt einen ausführlichen Vergleich Bayes'scher Netze, der Dempster-Shafer-Theorie und Fuzzy Logik sowie entsprechender benutzeradaptiver Systeme durch.

es ist aber nicht offensichtlich, wie sie beispielsweise im Zusammenhang mit einem Bayes'schen Netz als Evidenz genutzt werden kann. Kommen weitere Aussagen dieser Art (möglicherweise aus anderen Quellen) hinzu, wird diese Problemstellung zusätzlich erschwert.

Die Dempster-Shafer-Theorie arbeitet auf Mengen von Hypothesen, denen jeweils drei Arten von Werten zugeordnet werden:

1. *Basiswahrscheinlichkeit:* Sie gibt an, wie wahrscheinlich die zugehörige Hypothesenmenge ist. Es sind damit keine Aussagen über die Wahrscheinlichkeiten von Unter- und Obermengen verbunden.

2. *Belief:* Der Belief repräsentiert den Gesamtvertrauensgrad einer Hypothesenmenge, er berechnet sich als Summe aller Basiswahrscheinlichkeiten aller Teilhypothesenmengen. Er stellt somit einen Index dar, der angibt, wie wahrscheinlich es ist, dass sich die gesuchte Hypothese in der betrachteten Hypothesenmenge (als Teilmenge) befindet.

3. *Plausibilität:* Sie ergibt sich als die Summe aller Basiswahrscheinlichkeitswerte der Hypothesenmengen, die mindestens eine Hypothese mit der betrachteten gemeinsam haben. Die Plausibilität ist eine Größe, die das Potenzial der Wahrscheinlichkeit angibt, das der betrachteten Hypothesenmenge noch zugewiesen werden kann, wenn zusätzliche Evidenzen beobachtet werden.

Evidenzen werden mit der Dempster'schen Regel (siehe z.B. Bauer, 1996, S. 16) kombiniert und in den Schlussfolgerungsprozess eingebracht.

Einer der Vorteile gegenüber Bayes'schen Netzen, die Vertreter der Dempster-Shafer-Theorie anführen, ist, dass keine aufwendige Spezifikation von initialen A-priori-Wahrscheinlichkeitswerten anfallen. Eine Eigenschaft, die als Nachteil angesehen werden kann, besteht in der oft komplexen Formulierung der Entscheidungsprozesse. Anhand der drei beschrieben Werte zu jeder Hypothese muss ein System beispielsweise anhand einer Regelbasis unter Verwendung von Schwellwerten ermitteln, welche der möglichen Entscheidungen in der vorliegenden Situation potenziell am besten geeignet ist. Andererseits besitzen Dempster-Shafer-Systeme aus diesem Grund die Flexibilität, in verschiedenen Kontexten unterschiedliche Entscheidungsmechanismen einsetzen zu können.

2.1.6.2 Fuzzy Logik

Die grundlegende Idee der Fuzzy Logik ist das Schlussfolgern unter Verwendung vager Ausdrücke.[7] Die Zugehörigkeit realer Objekte zu vagen Konzepten wird in gradueller Weise vorgenommen—im Gegensatz zum Bool'schen Ansatz, wo ein Objekt entweder „ganz“ zu einem Konzept gehört oder überhaupt nicht. Die Fuzzy Logik versucht das menschliche Schlussfolgern abzubilden und resultiert deshalb in interpretierbaren und nachvollziehbaren Entscheidungsprozessen.

Im Zusammenhang mit benutzeradaptiven Systemen stellt das Konzept der vagen Ausdrücke oft einen Vorteil dar. Beispielsweise bestehen Selbsteinschätzungen—wie sie häufig vor der Benutzung eines Systems von einem neuen Benutzer gefordert werden—oft aus solchen Ausdrücken. Ein Beispiel hierfür ist die Aussage „Ich verstehe nicht *sehr viel* von Tabellenkalkulation.“. Durch

[7] Besonders prägnant ist der Titel einer Veröffentlichung des Begründers der Fuzzy Logik L.A. Zadeh „Fuzzy Logic = Computing With Words“ (Zadeh, 1996).

die Verwendung solcher Begriffe auch auf Systemseite wird häufig die Interaktion mit dem Benutzer erleichtert.

Ein Nachteil, der z.B. von Jameson (1996) angeführt wird, ist, dass es zwar einigermaßen einfach möglich ist, solche benutzeradaptiven Systeme auf der Basis von Fuzzy Logik einem „Fine-Tuning" hinsichtlich bestimmter Aspekte der Modellierung zu unterziehen, dass man damit oftmals aber aufgrund der fehlenden mathematischen Fundierung, wie sie bei Bayes'schen Netzen und der Dempster-Shafer-Theorie vorliegt, an anderer Stelle des Modells nicht vorhersehbare und möglicherweise nicht erwünschte Veränderungen bewirkt.

2.1.7 Verbale Erklärungen Bayes'scher Netze

Bayes'sche Netze stellen eine explizite graphische Modellierung einer Domäne dar. Durch die kausale Interpretation der Kanten ist ein Bayes'sches Netz keine „Black Box", die mit den Beobachtungen „gefüttert" wird und in einer nicht ohne Weiteres nachvollziehbaren Weise ein Ergebnis produziert. Meist kann—ähnlich, wenn auch nicht in einem solchen Ausmaß wie bei der Fuzzy Logik—selbst eine mit dem Formalismus nicht im Detail vertraute Person anhand der Struktur eines Netzes mit einer moderaten Anzahl an Variablen große Teile der Modellierung verstehen. Allerdings existiert auch bei Bayes'schen Netze eine Grenze, ab der die Nachvollziehbarkeit nicht mehr gegeben ist. Netzen mit Tausenden von Variablen, wie sie durchaus eingesetzt werden, sind meist zu komplex, um—zumindest in der Gesamtheit des Schlussfolgerungsprozesses—von Personen verstanden zu werden, die nicht an der Entwicklung beteiligt waren.

Die Möglichkeit für den Benutzer, den prinzipiellen Schlussfolgerungsvorgang verfolgen und die Ergebnisse des Systems weitestgehend verstehen zu können, dient einer erhöhten benutzerseitigen Akzeptanz des Gesamtsystems (Teach & Shortliffe, 1984; Cook & Kay, 1994; Herlocker et al., 2000). Gerade in benutzeradaptiven Systemen ist die Transparenz des Inferenzprozesses eine wichtige Eigenschaft, die großen Anteil am Erfolg des Systems haben kann (vgl. Abschnitt 3.1.3.5).

Druzdzel (1996) beschreibt einen Ansatz zur Erzeugung verbalisierter Erklärungen Bayes'scher Netze. Darin werden zwei Teilaufgaben identifiziert: (a) die Erklärung des Modells und (b) die Erklärung des Schlussfolgerungsprozesses. Ziel von (a) ist es, dem Benutzer die Annahmen, die der Modellierung zugrunde liegen, zu erläutern. Im Wesentlichen wird hier die Struktur des Bayes'schen Netzes betrachtet. Es existieren Verfahren, die den für die Erklärung gewisser Teilaspekte des Modells relevanten Bereich der Struktur—u.a. unter Verwendung des d-Separationskriteriums—ermitteln können. Die zweite Teilaufgabe (b) beschäftigt sich damit, einem Benutzer die Auswirkungen zu verdeutlichen, die durch die Interpretation beobachteter Evidenzen im Rahmen der Inferenzverfahren entstehen. In dieser Weise können die Entscheidungen des Systems begründet werden.

Die Verfahren zur Behandlung beider Erklärungsaspekte basieren auf den von Wellman (1990) vorgestellten *qualitativen probabilistischen Netzwerken*. Sie bauen auf der Feststellung auf, dass es in einigen Domänen genügt, lediglich die qualitativen Zusammenhänge zwischen den Zuständen der betrachteten Variablen zu kennen, um sinnvolle Inferenzen zu ziehen. Dazu werden die Zusammenhänge zwischen den betrachteten Variablen statt mit (quantitativen) CPTs mit qualitativen Informationen modelliert, beispielsweise bezüglich der Art einer monotonen Beziehung (positiv (+) / negativ (-)) zweier Variablen. In Abschnitt 5.2.2.1 wird eine detaillierte Beschreibung der verschiedenen Arten der qualitativen Information gegeben. Ein einfaches Beispiel hierfür ist die positive monotone Beziehung zwischen den Variablen WISSEN und KLAUSURNOTE in Beispiel

2.1.3: Ein höheres Wissensniveau eines Studenten führt üblicherweise zu einer besseren Note in der Klausur.

Auf der Basis solcher Informationen ist es möglich, verbale Erklärungen zum Schlussfolgerungsprozess und den dem Modell zugrunde liegenden Annahmen zu generieren. Zusätzlich werden die numerischen Aspekte des zu erklärenden Bayes'schen Netzes wie die bedingten Wahrscheinlichkeiten der CPTs oder die im Rahmen der Inferenzverfahren ermittelten Wahrscheinlichkeiten wie schon von Wahlster (1981) beschrieben auf Formulierungen der Art „sehr unwahrscheinlich", „üblicherweise" usw. abgebildet. Damit können die quantitativen Aspekte in die generierten Erklärungen einfließen. Eine Erklärung, die die in Tabelle 2.1 aufgeführten CPTs nutzt, könnte z.B. folgendermaßen lauten: „Es ist sehr wahrscheinlich, dass diese Studentin ein hohes Wissensniveau besitzt, da sie eine hohe Anzahl der Übungen bearbeitet hat und bereits ein großes Vorwissen zu dieser Thematik besaß."

In der Praxis sollten solche verbalen Erklärungen in Kombination mit anderen Möglichkeiten wie z.B. graphischen Darstellungen kombiniert werden, um ein erleichtertes Verständnis zu erzielen.

2.2 Beispielhafte Modellierungen mit Bayes'schen Netzen: Psychologisch motivierte Benutzerstudien des READY-Projekts

Ein Schwerpunkt des READY-Projekts ist die empirische Fundierung der angewendeten Techniken zur Benutzermodellierung und entscheidungstheoretischen Planung durch psychologische Experimente (vgl. Abschnitt 1.1.2). In diesem Abschnitt werden zwei dieser Experimente vorgestellt und mit Hilfe von Bayes'schen Netzen modelliert, die im weiteren Verlauf der Arbeit verwendet werden, um die entwickelten maschinellen Lernverfahren zu diskutieren und empirisch zu evaluieren. Dazu eignen sich Experimentaldaten dieser Art besonders gut, da die Daten in einer kontrollierten Situation erhoben werden und die Ergebnisse weitestgehend frei von unbeeinflussbaren bzw. unvorhersehbaren Effekten sind. Insbesondere können durch die Manipulation der unabhängigen Variablen der Experimente verschiedene Experimentalsituationen betrachten werden, die im Rahmen der Interpretation der Ergebnisse der maschinellen Lernverfahren wichtige Vergleichsmöglichkeiten bieten. Eine Einordnung, inwieweit diese im Rahmen einer Experimentalumgebung gesammelten Daten in den Konstruktionsprozess eines Benutzermodells im Zusammenspiel mit Daten aus anderen Quellen einfließen, erfolgt in Abschnitt 4.1.

Nach der Beschreibung des Experimentalaufbaus und -ablaufs sowie einer kurzen Diskussion der wichtigsten Ergebnisse werden für jedes Experiment alternative Modelle der Domäne in Form Bayes'scher Netze vorgestellt. Für detaillierte Informationen zu den beiden Experimenten, die für den weiteren Verlauf der Arbeit von untergeordneter Bedeutung sind, wird auf die jeweils angegebenen Originalarbeiten verwiesen. Insbesondere wird an dieser Stelle auf eine ausführliche Präsentation und Diskussion der Ergebnisse der traditionellen statistischen Analyse der Experimente verzichtet.

2.2.1 Anweisungsexperiment: Bearbeitung von Anweisungsfolgen

Das erste der beiden Experimente—im Folgenden *Anweisungsexperiment* genannt—beschäftigt sich mit Sequenzen von Anweisungen, die Benutzern präsentiert werden, und der Analyse, inwieweit diese in der Lage sind, diese Instruktionen in verschiedenen Situationen kognitiver Belastung korrekt auszuführen.

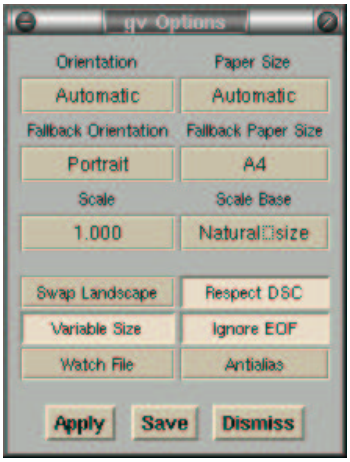


***Abbildung 2.3**: Beispiel eines typischen Optionsfenster*

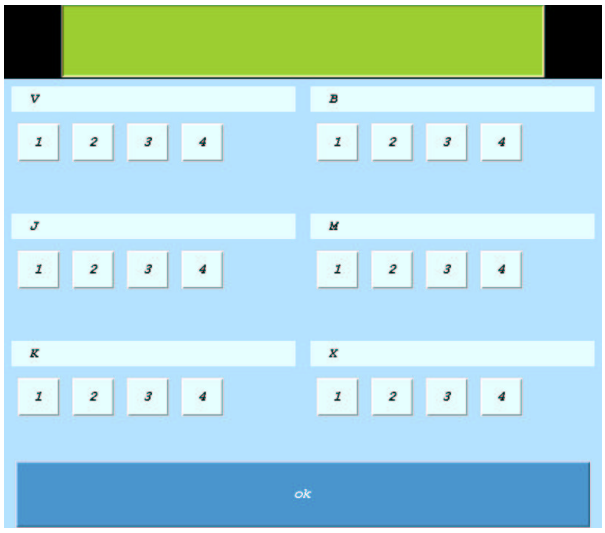
Abbildung 2.3 zeigt den Bildschirmabzug eines typisches Optionsfensters eines Programms.[8] Ein Hilfesystem müsste in einer solche Situationen entsprechend den gewünschten Einstellungen eine Folge von (Einzel-)Anweisungen geben, wie z.B. „Setze ORIENTATION auf LANDSCAPE, setze PAPER SIZE auf DIN A4 …". Es stellt sich die Frage, in welcher Form die Instruktionen gegeben werden sollen: Sollen sie (a) *gebündelt* werden, d.h., alle Instruktionen hintereinander zusammen in einem Paket, bevor der Benutzer mit der Ausführung der ersten Anweisung beginnt, oder (b) *schrittweise*, d.h., das Hilfesystem gibt eine Instruktion und wartet mit der Präsentation der nächsten bis der Benutzer die vorherige Anweisung ausgeführt hat? Eine schrittweise Präsentation hat den Nachteil, dass sie bedingt durch den zusätzlichen Aufwand der jeweiligen benutzerseitigen Rückmeldung bei abgeschlossener Ausführung normalerweise längere Gesamtausführungszeiten nach sich zieht, wohingegen eine gebündelte Anweisungsfolge tendenziell zu einer höheren Fehlerrate bei der Bearbeitung der Aufgaben führt. Dies ist insbesondere dann von Bedeutung, wenn der Benutzer durch eine weitere Aufgabe zusätzlich kognitiv belastet ist, wie z.B. durch ein parallel geführtes Telefongespräch.

2.2.1.1 Aufbau

Abbildung 2.4 zeigt die konkrete Experimentalumgebung, mit der die Versuchspersonen umgehen mussten. Sie stellt eine abstrakte Repräsentation der im vorigen Abschnitt beschrieben Situation dar, um Einflüsse unterschiedlichen Vorwissens der Versuchspersonen hinsichtlich des Umgangs mit einem speziellen Programm auf die Ergebnisse des Experiments weitestgehend zu vermeiden. Sie besteht aus sechs Gruppen zu je vier Buttons, die durch Anklicken mit der Maus aktiviert werden können. Am unteren Rand befindet sich ein OK-Button, durch dessen Betätigung die Versuchspersonen die Beendigung der Bearbeitung einer Instruktion signalisieren können. Im oberen Bereich des Bildschirms ist eine Simulation einer aufleuchtenden Lampe in Form eines farbigen Balkens platziert, der in den Farben Rot und Grün blinkt.

Die Hauptaufgabe der 24 beteiligten Versuchspersonen bestand darin, Anweisungsfolgen der Art wie in Abbildung 2.4 angedeutet auszuführen. Sie bestanden aus zwei, drei oder vier Einzelanweisungen, die in gesprochener Form durch Abspielen aufgezeichneter Audio-Dateien präsen-

[8] gv Linux-Version 3.5.8



Schrittweise:		*Gebündelt:*	
$\mathcal{S}$:	Setze *X* auf 3.	$\mathcal{S}$:	Setze *X* auf 3,
$\mathcal{B}$:	... [OK]		setze *M* auf 1,
$\mathcal{S}$:	Setze *M* auf 1.		setze *V* auf 4
$\mathcal{B}$:	... [OK]		
$\mathcal{S}$:	Setze *V* auf 4.		

***Abbildung 2.4**: Experimentalumgebung des Anweisungsexperiments*
(S: System / B: Benutzer)

tiert wurden. Jede dieser Anweisungsfolgen wurde vom Experimentalsystem entweder schrittweise oder gebündelt dargeboten. Im schrittweisen Modus mussten die Versuchspersonen nach der Ausführung jeder Einzelanweisung die Beendigung durch Anklicken des OK-Buttons signalisieren. Erst danach wurde gegebenenfalls die nächste Einzelanweisung gegeben. Im gebündelten Modus hingegen musste erst nach Ende der Ausführung der kompletten Anweisungsfolge der OK-Button betätigt werden, um das nächste Anweisungsbündel zu erhalten.

Bei der Hälfte der Versuchsaufgaben wurden die Versuchspersonen zusätzlich einer ablenkenden Nebenaufgabe ausgesetzt. Dazu blinkte die im oberen Bereich der Experimentalumgebung angeordnete Lampe in zufälliger Reihenfolge und in mehr oder weniger regelmäßigen Abständen rot und grün. Die Versuchspersonen wurden angewiesen, jedes Mal, wenn die Lampe zweimal hintereinander in der gleichen Farbe aufleuchtete, dies durch Drücken der Leertaste anzuzeigen.

2.2.1.2 Variablen

Folgende Variablen, die in dieser Arbeit relevant sind, wurden im Experiment untersucht:[9]

- **Unabhängige Variablen:**
  - PRÄSENTATIONSMODUS: Die Anweisungen wurden entweder *schrittweise* oder *gebündelt* gegeben.
  - ANZAHL DER ANWEISUNGEN: Die Anweisungsfolgen bestanden aus *zwei, drei* oder *vier* Einzelanweisungen.
  - ABLENKUNG?: Die ablenkende Nebenaufgabe war entweder zu bearbeiten oder nicht.

[9] Von March (1999) werden sowohl eine detailliertere Beschreibung der Variablen gegeben als auch weitere Variablen aus psychologischer Sicht analysiert, die aber in dieser Arbeit von keiner Bedeutung sind.

Damit wurden durch orthogonale Kombination der möglichen Zustände der Variablen zwölf $(2\times3\times2)$ Experimentalbedingungen geschaffen und betrachtet. In jeder dieser Bedingungen mussten die 24 Versuchspersonen sechs Anweisungsfolgen bearbeiten. Somit hat man einen Datensatz von insgesamt 1728 $(24 \times 6 \times 12)$ von allen Versuchspersonen durchgeführten Anweisungsfolgen zur Analyse zur Verfügung.

- **Abhängige Variablen:**
  - FEHLER?: Diese binäre Variable nimmt den Zustand *Ja* an, wenn die Versuchsperson einen Fehler bei der Ausführung einer der Einzelanweisungen der kompletten Anweisungsfolge gemacht hat.
  - AUSFÜHRUNGSZEIT: Diese Variable repräsentiert die Zeit, die die Versuchsperson benötigte, um die komplette Anweisungsfolge zu bearbeiten. Dabei werden im Fall der schrittweisen Präsentation die Zeiten, die das Experimentalsystem benötigt, um die Audio-Dateien mit den Anweisungen abzuspielen, nicht berücksichtigt.
  - FEHLER IN DER NEBENAUFGABE?: Diese binäre Variable nimmt den Zustand *Ja* an, wenn die Versuchsperson einen Fehler in der ablenkenden Nebenaufgabe gemacht hat, d.h., fälschlicherweise die Leertaste gedrückt hat oder bei zweimaligen Aufblinken der gleichen Farbe den entsprechenden Tastendruck nicht vorgenommen hat. Diese Variable spielt im Rahmen der Analysen lediglich eine untergeordnete Rolle. Der Zweck der Ablenkung bestand hauptsächlich in der Erzeugung einer zusätzlichen kognitiven Belastung für die Versuchspersonen.

2.2.1.3 Ergebnisse

Eine traditionelle statistische Varianzanalyse zeigte die folgenden signifikanten Haupteffekte auf (vgl. March, 1999):

- Eine längere Anweisungsfolge führt zu längeren Ausführungszeiten und mehr Fehlern.
- Das Vorhandensein der ablenkenden Nebenaufgabe erhöht ebenfalls die Ausführungszeiten und Häufigkeiten der Fehler.
- Der schrittweise Präsentationsmodus zieht höhere Ausführungszeiten nach sich (im Wesentlichen bedingt durch den zusätzlichen zeitlichen Aufwand durch die Notwendigkeit der Bestätigung der Beendigung der Einzelanweisung), reduziert aber andererseits die Häufigkeit der Fehler. Eine plausible Erklärung für letzteren Effekt ist die Überlegung, dass die Versuchsperson weniger Information im Arbeitsgedächtnis speichern muss und somit nicht so sehr Gefahr läuft, verbleibende Instruktionen zu vergessen.

Interessanter als diese Haupteffekte ist die signifikante Interaktion zwischen den beiden Variablen PRÄSENTATIONSMODUS und ABLENKUNG?: Der Anstieg der Fehlerhäufigkeit bei gebündelter Präsentation ist deutlich höher, wenn die ablenkende Nebenaufgabe zu bearbeiten ist, d.h., ohne Ablenkung sind die Versuchspersonen weitestgehend in der Lage mit dem anspruchsvolleren gebündelten Präsentationsmodus umzugehen.

Zusammenfassend kann man hinsichtlich der praktischen Relevanz sagen, dass ein solches System tendenziell eine schrittweise Präsentation wählen sollte, wenn eine ablenkende Nebenaufgabe zu bearbeiten ist (um Fehler zu vermeiden), andernfalls, um Zeit zu sparen, die gebündelte

Variante wählen sollte. Die Auswahl hängt in der spezifischen Anwendungssituation von zusätzlichen Faktoren ab, wie beispielsweise dem relativen Gewicht zwischen der Bedeutung einer Vermeidung von Fehlern und einer erhöhten Geschwindigkeit der Ausführung der Anweisungen.

Eine solche statistische Varianzanalyse liefert zwar Informationen über Zusammenhänge zwischen Variablen(gruppen), man hat damit aber noch kein Modell, mit dem man in der Lage wäre, aufgrund von Evidenzen, Schlussfolgerungen über andere Sachverhalte in der Domäne zu ziehen. Dieses Problem kann durch Modelle in Form von Bayes'schen Netzen gelöst werden.

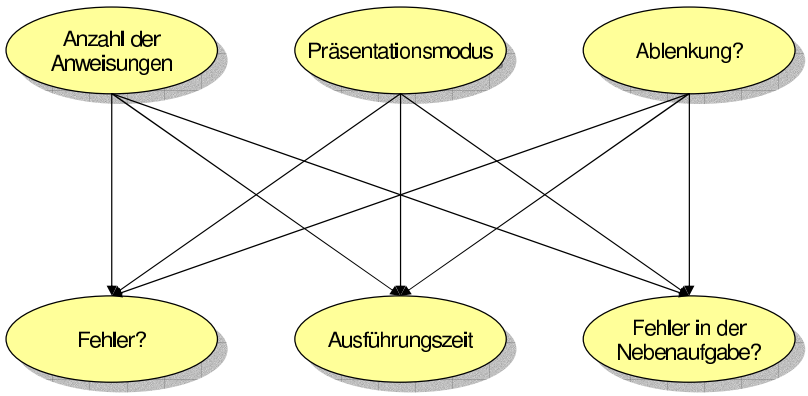
2.2.1.4 Modellierung mit Bayes'schen Netzen

Abbildung 2.5 stellt zwei alternative Modelle der Experimentalsituation, in der sich die Versuchspersonen des Anweisungsexperimentes befinden, dar. Abbildung 2.5 (a) repräsentiert ein naheliegendes, einfaches Bayes'sches Netz, das als Grundlage eines Vergleichs mit aufwendigeren Modellierungsansätzen dienen kann. Abbildung 2.5 (b) zeigt ein Beispiel einer solchen komplexeren Variante.

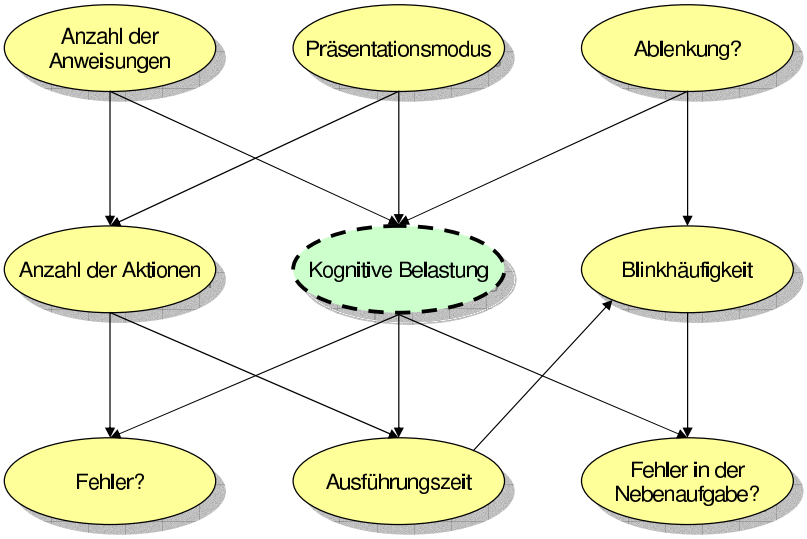
Die Variablen des Bayes'schen Netzes stimmen mit den abhängigen und unabhängigen Variablen des Experiments überein. Durch die Einteilung der Variablen in abhängige und unabhängige und die Annahme, dass eine kausale Beziehung zwischen jeder unabhängigen und jeder abhängigen Variablen besteht, ergibt sich in kanonischer Weise die erste abgebildete Struktur (Abbildung 2.5 (a)). Ein Nachteil dieses Modells ist, dass es zwar die numerischen Ergebnisse der Experimentalsituation repräsentiert, jedoch keinerlei Erklärungsmöglichkeiten hinsichtlich des Warums anbietet.

Dem alternativen Modell aus Abbildung 2.5 (b) liegt das Ziel zugrunde, die Interpretierbarkeit des Modells zu erhöhen. Dazu wurden drei zusätzliche Variablen in das Bayes'sche Netz aufgenommen, ANZAHL DER AKTIONEN, BLINKHÄUFIGKEIT und KOGNITIVE BELASTUNG. Die erste neue Variable ist eine explizit in der Experimentalumgebung messbare Größe. Bei KOGNITIVE BELASTUNG handelt es im Gegensatz dazu um eine nicht explizit messbare, *erklärende* Variable, die hauptsächlich die Interpretierbarkeit des Netzes erhöht. Ihre Aufnahme in das Modell basiert auf der Hypothese, dass die drei unabhängigen Variablen ANZAHL DER ANWEISUNGEN, PRÄSENTATIONSMODUS und ABLENKUNG? einen direkten kausalen Einfluss auf die kognitive Belastung ausüben, der die Versuchspersonen ausgesetzt sind während sie die Instruktionen ausführen. Diese kognitive Belastung hat wiederum direkte kausale Einflüsse auf die Wahrscheinlichkeit der Fehler bzw. der Ausführungsgeschwindigkeiten. Allerdings können die Beziehungen zwischen den unabhängigen und den abhängigen Variablen des Experiments nicht adäquat ausschließlich durch die Variable KOGNITIVE BELASTUNG erfasst werden. Daher wird eine weitere neue Variable ANZAHL DER AKTIONEN eingeführt: Sie repräsentiert die Anzahl der Mausklicks und Tastenbetätigungen, die eine Versuchsperson benötigt, um die Aufgaben in den verschiedenen experimentellen Situationen korrekt zu bearbeiten. Da sie den Arbeitsaufwand modelliert, hat sie einen direkten kausalen Einfluss auf die von der Versuchsperson benötigten Ausführungszeiten. Ebenso bietet jedes Klicken und jeder Tastendruck im Experimentaldesign eine Möglichkeit, einen Fehler bei der korrekten Instruktionsausführung zu begehen, weshalb eine Kante zu FEHLER? eingefügt wurde. Anhand dieses Bayes'schen Netzes sind im Gegensatz zur ersten, einfacheren Variante Erklärungen der Art „Eine durch eine ablenkende Nebenaufgabe erhöhte kognitive Belastung der Versuchsperson führt mit hoher Wahrscheinlichkeit zu einem Fehler." möglich. Es können Aussagen über die postulierten kognitiven Prozesse[10] gemacht werden.

[10] Aus psychologischer Sicht ist dieses Modell sicherlich stark vereinfacht. Hinsichtlich des Ziels, der Verwendung des Bayes'schen Netzes in einem benutzeradaptiven System, stellt diese Modellierung jedoch einen sinnvollen Kompromiss dar.



(a) einfach



(b) komplex

***Abbildung 2.5**: Beispiele Bayes'scher Netze zur Modellierung des Anweisungsexperiments (Durch unterbrochene Linien bzw. mit grüner Farbe markierte Knoten repräsentieren erklärende, verborgene Variablen).*

2.2.2 Flughafenexperiment: Symptome sprachlicher Äußerungen

Im Rahmen des zweiten Experiments, das im Weiteren als *Flughafenexperiment* bezeichnet wird, wurde ein weiterer wichtiger Aspekt des READY-Projekts untersucht: die Erkennung von situativ bedingten kognitiven Ressourcenbeschränkungen auf der Benutzerseite. Im Speziellen wurden die Zusammenhänge zwischen eben diesen situativ bedingten kognitiven Ressourcenbeschränkungen und Symptomen der gesprochenen Sprache der Benutzer betrachtet, wie z.B. Pausen, Satzabbrüchen und Artikulationsgeschwindigkeiten. Intuitiverweise würde man z.B. erwarten, dass ein kausaler Zusammenhang zwischen dem vorhandenen Zeitdruck und der Artikulationsgeschwindigkeit existiert. Um solche Zusammenhänge zu untersuchen, wurde in einer experimentellen Umgebung auf einem PC eine Situation simuliert, in der die Versuchspersonen durch ein belebtes Flughafenterminal navigieren müssen, während sie mittels gesprochener Sprache Anfragen an ein hypothetisches mobiles Hilfesystem stellen.

2.2.2.1 Aufbau

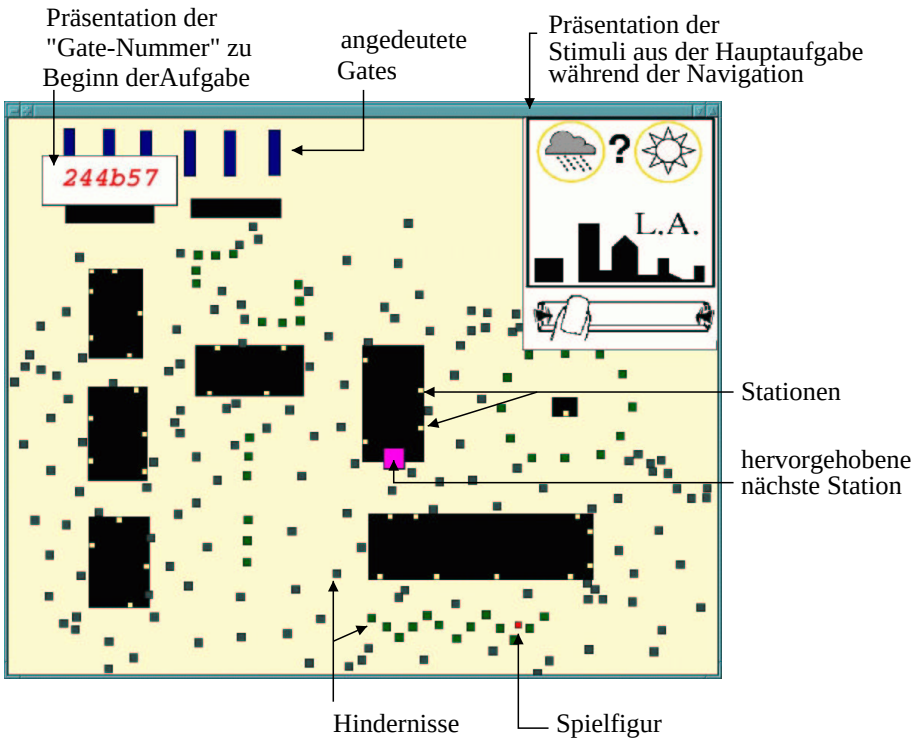


***Abbildung 2.6**: Experimentalumgebung des Flughafenexperiments*

Ein annotierter Bildschirmabzug der Experimentalumgebung ist in Abbildung 2.6 zu sehen.[11] Jede der 32 Versuchsperson musste 80 Aufgaben bearbeiten. Anhand eines in der rechten oberen Ecke der Experimentalumgebung erscheinenden Bildes sollten die Versuchspersonen Fragen formulieren, wie z.B. „Wie ist das Wetter in Los Angeles? Regnet es oder scheint die Sonne?".

[11] Detaillierte Beschreibungen des Experiments und der Ergebnisse finden sich bei Müller (2001) und Müller, Großmann-Hutter, Jameson, Rummer und Wittig (2001).

In der Hälfte der Aufgaben musste zusätzlich eine Navigationsaufgabe durchgeführt werden, in welcher der auf dem Bildschirm als Spielfigur repräsentierte Flughafengast durch das angezeigte Terminal zu bestimmten Zielen mit Hilfe der Cursor-Tasten navigiert werden musste. Dabei sollten Hindernisse vermieden sowie eine aus fünf alphanumerischen Zeichen bestehende Abfluggatebezeichnung memoriert werden.

Der zweite Teilaspekt der experimentellen Konfigurationen bestand aus dem Formulieren der Fragen unter Zeitdruck bzw. unter besonderer Berücksichtigung der Qualität der formulierten Fragen. Dazu wurden die Versuchspersonen durch in Aussicht stellen einer entsprechenden Belohnung angewiesen, entweder (a) eine Frage möglichst schnell zu formulieren, oder (b) eine Frage möglichst klar verständlich ohne Zeitbeschränkungen zu formulieren.

Die Äußerungen der Versuchspersonen wurden aufgezeichnet und manuell auf eine Vielzahl sprachlicher Symptome untersucht (vgl. Berthold, 1998). Weiterhin wurden die Qualitäten der je 32 formulierten Fragen pro Bild sowie die Komplexität jedes Bildes hinsichtlich der Schwierigkeit der Formulierung einer adäquaten Frage von vier unabhängigen Gutachtern bewertet.

2.2.2.2 Variablen

Wie bereits beim Anweisungsexperiment werden auch hier unabhängige und abhängige Variablen unterschieden. Letztere bestehen im Wesentlichen aus den verschiedenen Sprachsymptomen.

- **Unabhängige Variablen:**
  - NAVIGATION?: Diese binäre Variable gibt an, ob die Navigationsaufgabe zu bearbeiten war oder nicht.
  - ZEITDRUCK?: Sollte die Frage möglichst schnell oder möglichst qualitativ hochwertig (ohne Zeitbeschränkung) formuliert werden?
  - SCHWIERIGKEIT DER FRAGEFORMULIERUNG: Diese dritte unabhängige Variable wurde im Experiment nicht explizit manipuliert. Sie dient der Repräsentation der (mit Hilfe der Gutachter ermittelten) Schwierigkeit der Bilder hinsichtlich der Frageformulierung.

  Durch die orthogonale Kombination der Zustände der beiden ersten unabhängigen Variablen ergeben sich vier (2×2) experimentelle Bedingungen. Mit 80 Einzelaufgaben für 32 Versuchspersonen ergeben sich 2560 Einzelfälle als Basis der Analysen.

- **Abhängige Variablen:**
  - ARTIKULATIONSGESCHWINDIGKEIT: Anzahl der Silben pro Sekunde Sprechzeit, ohne die Zeiten der gemessenen stillen Pausen.
  - QUALITÄTSSYMPTOME: Diese Variable repräsentiert eine logische Disjunktion verschiedener binärer Variablen, von denen jede eine Form der formalen Qualität der Äußerungen widerspiegelt: Selbstkorrekturen, Fehlstarts und Unterbrechungen von Wörtern bzw. Sätzen.
  - INHALTLICHE QUALITÄT: Die von den Gutachtern bewertete inhaltliche Qualität der formulierten Äußerung.
  - SILBENANZAHL: Die Anzahl der Silben einer Äußerung.

  - STILLE PAUSEN: Die Gesamtdauer der stillen Pausen einer Äußerung bezogen auf die Anzahl der Wörter.
  - GEFÜLLTE PAUSEN: Die entsprechende Variable für gefüllte Pausen. Gefüllte Pausen sind sprachliche Artefakte wie „Äh", „Ehm" usw.

  Eine detailliertere Analyse sowie die Betrachtung weiterer Variablen wird von Müller (2001) diskutiert.

2.2.2.3 Ergebnisse

Eine statistische Varianzanalyse ergab folgende Hauptergebnisse (siehe ebenfalls Müller, 2001):

- QUALITÄTSSYMPTOME: Diese Symptome treten etwas häufiger auf, wenn die Navigationsaufgabe zu bearbeiten war.
- SILBENANZAHL: Die Silbenanzahl ist im Fall ohne Zeitdruck höher. Der Effekt verringert sich wenn zu navigieren war.
- STILLE PAUSEN: Diese Variable verhält sich ähnlich wie SILBENANZAHL.
- ARTIKULATIONSGESCHWINDIGKEIT: Der Wert dieser Variablen erhöht sich unter Zeitdruck und verringert sich bei vorhandener Navigationsaufgabe.

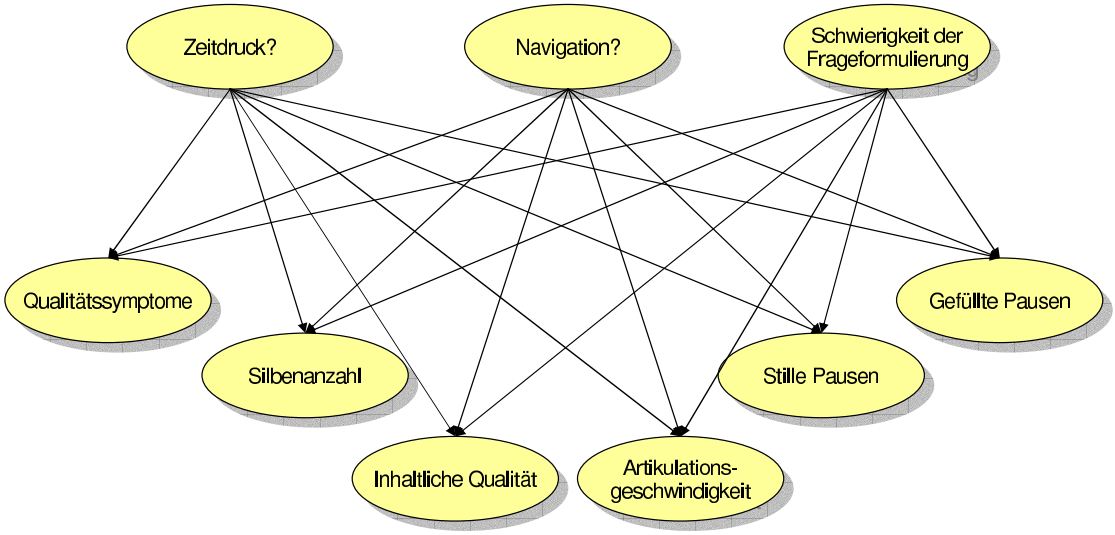
Insgesamt gibt es viele statistisch signifikante Effekte der unabhängigen Variablen auf die Sprachsymptome, die allerdings in den meisten Fällen recht komplex oder subtil sind. Es ist deshalb keine triviale Aufgabe, (a) die Sprachsymptome einer Äußerung einer Versuchperson in einer bestimmten experimentellen Konfiguration vorherzusagen oder (b) die experimentelle Situation anhand der beobachteten Sprachsymptome zu ermitteln.

Die praktische Relevanz dieses Experiments bezüglich eines mobilen sprachbasierten Assistenzsystems liegt darin, dass ein solches System anhand festgestellter Sprachsymptome Inferenzen über den kognitiven Zustand seines Benutzers ziehen kann. Auf der anderen Seite trägt es dazu bei, Vorhersagen über die möglicherweise auftretenden Sprachsymptome zu machen, um beispielsweise zu entscheiden, ob vom System eine Spracheingabe vom Benutzer angefordert werden soll oder eventuell besser eine andere Modalität wie z.B. eine graphische Eingabe zu wählen ist.

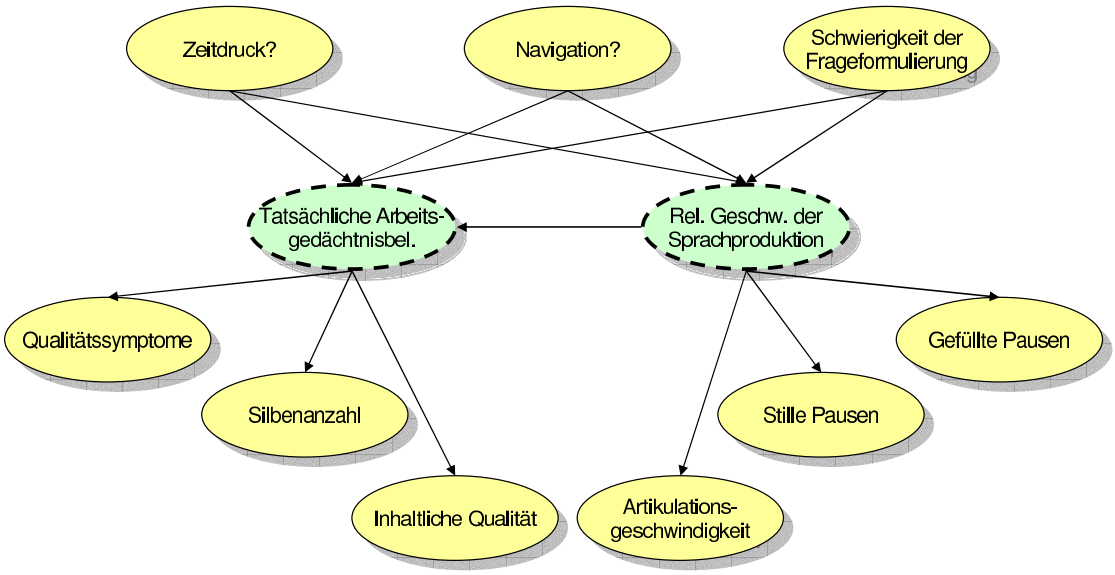
2.2.2.4 Modellierung mit Bayes'schen Netzen

Den beiden in Abbildung 2.7 dargestellten Bayes'schen Netzen liegt die gleiche Motivation zugrunde wie den Strukturen zur Modellierung des Anweisungsexperiments. Abbildung 2.7 (a) zeigt ein einfaches Netz, das die Einteilung der Variablen in unabhängige und abhängige ausnutzt, aber keinerlei tiefergehende Interpretationsmöglichkeiten anbietet.

Die komplexere der beiden Strukturen (Abbildung 2.7 (b)) stellt wiederum eine stärker theoretisch motivierte Variante dar. Auch hier wurden zwei zusätzliche Variablen TATSÄCHLICHE ARBEITSGEDÄCHTNISBELASTUNG und RELATIVE GESCHWINDIGKEIT DER SPRACHPRODUKTION eingeführt. In beiden Fälle handelt es sich um erklärende Variablen, die im Experimentaldesign nicht explizit messbar sind. Folgende theoretische Überlegungen liegen der Integration der beiden neuen Variablen zugrunde (vgl. Berthold, 1998): Das Vorhandensein der Navigationsaufgabe induziert eine erhöhte Arbeitsgedächtnisbelastung, die sich wiederum im verstärkten Auftreten von



(a) einfach



(b) komplex

***Abbildung 2.7**: Beispiele Bayes'scher Netze zur Modellierung des Flughafenexperiments*

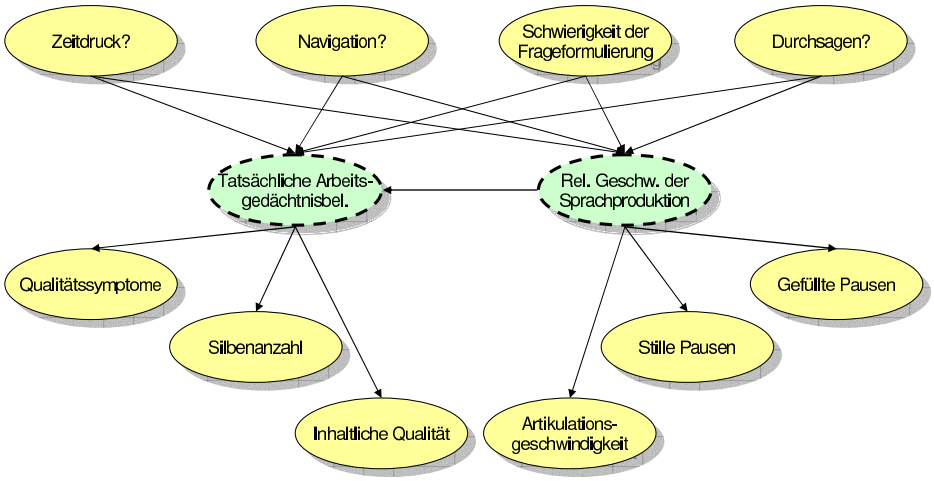


***Abbildung 2.8**: Beispiel eines Bayes'schen Netzes zur Modellierung des erweiterten Flughafenexperiments*

qualitäts-relevanten Symptomen (QUALITÄTSSYMPTOME, INHALTLICHE QUALITÄT und ANZAHL DER SILBEN) äußert. Die Versuchsperson kann diesen Effekt allerdings vermindern, indem sie die relative Geschwindigkeit der Sprachproduktion vermindert, was gleichzeitig zu einer Reduktion der tatsächlichen Arbeitsgedächtnisbelastung führt. Eine Reduktion der relativen Geschwindigkeit der Sprachproduktion spiegelt sich in zeit-relevanten Sprachsymptomen wider, wie ARTIKULATIONSGESCHWINDIGKEIT, STILLE PAUSEN und GEFÜLLTE PAUSEN. Man sieht, dass man auch in dieser Domäne Informationen über die qualitative Art der kausalen Zusammenhänge zwischen Variablen zur Konstruktion des Modells ausnutzen kann.

2.2.2.5 Erweitertes Flughafenexperiment: Zusätzliche Ablenkung durch gehörte Sprache

Das beschriebene Experiment wurde von Kiefer (2002) um die Untersuchung der Auswirkungen einer zusätzlichen Ablenkung durch gehörte Sprache in Form von Lautsprecherdurchsagen erweitert. Es wird im Weiteren als *erweitertes Flughafenexperiment* bezeichnet.

Dazu wurde das Experiment mit 32 weiteren Versuchspersonen repliziert, wobei in diesem Fall während der Bearbeitung der Experimentalaufgaben Lautsprecherdurchsagen abgespielt wurden. Die Lautsprecherdurchsagen wurden im Frankfurter Flughafen aufgezeichnet und bestanden aus Flughinweisen, Suchaufrufen von Personen, Sicherheitshinweisen u.Ä. Unter bestimmten Umständen können solche Durchsagen belastend wirken, da je nach Art der Durchsage die Aufmerksamkeit des Flughafengastes mehr oder weniger erforderlich ist. Weitere Details zu den aufgezeichneten Daten, der Durchführung des Experiments und den Ergebnissen beschreibt Kiefer (2002).

Konzeptuell ergibt sich für die Modellierung mit Bayes'schen Netzen aus der Kombination der beiden Datensätze zu einer Gesamtdatenmenge von 5120 Einzelfällen die Aufnahme einer dritten unabhängigen, binären Variablen DURCHSAGEN? wie in Abbildung 2.8 dargestellt.

Es ist hinsichtlich der erhobenen Daten zu beachten, dass Unterschiede zwischen den beiden Varianten des Flughafenexperiments existieren: (a) Die manuelle Kodierung der Sprachsymptome wurde bei dieser replizierten Variante des Flughafenexperiments von einer zweiten Person vorgenommen. Und (b), die beiden Varianten wurden mit unterschiedlichen Versuchspersonen

durchgeführt. Diese Fakten müssen bei der Interpretation der mit den Daten der beiden Flughafenexperimente erzielten Ergebnisse berücksichtigt werden.

2.3 Erweiterung Bayes'scher Netze zu Einflussdiagrammen

Die in den letzten beiden Abschnitten vorgestellten Modellierungen mit Bayes'schen Netzen sind zwar in der Lage, die probabilistischen Zusammenhänge zwischen den betrachteten Variablen zu repräsentieren, sie alleine bieten allerdings noch keine Möglichkeit, anhand von Beobachtungen adäquate *Entscheidungen* zu treffen, wie z.B. welcher Präsentationsmodus in einer vorliegenden Situation gewählt werden sollte (gebündelt oder schrittweise). Einen solchen entscheidungstheoretischen, eng mit Bayes'schen Netzen verwandten Ansatz stellen *Einflussdiagramme* dar (siehe beispielsweise Neapolitan, 1990; Jensen, 1996, 2001). Sie können als eine Erweiterung des Konzepts Bayes'scher Netze angesehen werden, wie auch die folgende formale Definition zeigt. Sie orientiert sich an der von Jensen (2001):

Definition 2.2 (Einflussdiagramm) *Ein* Einflussdiagramm *besteht aus einem gerichteten azyklischen Graphen über der Vereinigung dreier verschiedener Knotenmengen: je einer Menge von* Zufallsknoten $\boldsymbol{X} = \{X_1, \ldots, X_n\}$, Entscheidungsknoten $\boldsymbol{D} = \{D_1, \ldots, D_m\}$ *und* Bewertungsknoten $\boldsymbol{U} = \{U_1, \ldots, U_l\}$. *Es gelten folgende strukturelle Eigenschaften:*

- *es existiert ein gerichteter Pfad, der alle Entscheidungsknoten $\boldsymbol{D}$ beinhaltet,*
- *die Bewertungsknoten $\boldsymbol{U}$ haben keine Kinder.*

Weiterhin gilt:

- *die Zufalls- und Entscheidungsknoten $\boldsymbol{X}$ bzw. $\boldsymbol{D}$ besitzen jeweils eine Menge sich gegenseitig ausschließender Zustände, die den kompletten Wertebereich der jeweiligen Variablen überdecken,*
- *jedem Zufalls- und Entscheidungsknoten X_i bzw. D_i ist eine CPT $\boldsymbol{\theta}_i$ zugeordnet,*
- *die Bewertungsknoten $\boldsymbol{U}$ haben keine Zustände,*
- *jedem Bewertungsknoten U_j ist eine reell-wertige* Bewertungsfunktion f_{U_j} *über $\boldsymbol{pa}(U_j)$ zugeordnet.*

Die Zufallsknoten eines Einflussdiagramms entsprechen den Knoten bzw. Variablen eines Bayes'schen Netzes. Gemeinsam mit den Kanten (inklusive assoziierter CPTs), die zwischen zwei Zufallsknoten angesiedelt sind, bilden sie das dem Einflussdiagramm zugrunde liegende Bayes'sche Netz. Entscheidungsknoten modellieren diskrete Punkte eines Entscheidungsprozesses, die die Wahl alternativer Optionen in Abhängigkeit von vorherigen Beobachtungen und Entscheidungen ermöglichen. Diese zeitlichen Abhängigkeiten werden durch Kanten (ohne CPTs) im Einflussdiagramm (zusätzlich zu jenen des zugrunde liegenden Bayes'schen Netzes) repräsentiert. Die dritte Knotenmenge, die Bewertungsknoten, dient der Bewertung der alternativen Möglichkeiten. Dazu werden die mittels der Inferenzverfahren im Bayes'schen Netz bestimmten Wahrscheinlichkeiten der Elternzustände eines Bewertungsknotens unter Anwendung der Bewertungsfunktion

f_{U_j} bewertet. Verschiedene (Teil-)Bewertungen der einzelnen Bewertungsknoten eines Einflussdiagramms werden additiv verknüpft, d.h., die Gesamtbewertung ergibt sich als Summe der Einzelbewertungen der l Bewertungsknoten. Anhand des Beispiels des Anweisungsexperiments wird im Folgenden das Konzept und die Anwendung eines Einflussdiagramms verdeutlicht.

Die zentrale Problemstellung im Szenario des Anweisungsexperiments ist adäquate Auswahl einer der beiden Präsentationsmodi in einer bestimmten Situation, die durch die Anzahl der Einzelinstruktionen und das (Nicht-)Vorhandensein einer ablenkenden Nebenaufgabe gekennzeichnet ist. Will man zur Lösung dieses Problems ein Einflussdiagramm konstruieren, so kann man beispielsweise die Struktur des Bayes'schen Netzes aus Abbildung 2.5 (a) als Ausgangspunkt zugrunde legen. Bisher wurde noch nicht erläutert, wie die CPTs dieses Netzes spezifiziert werden. In einem kleinen Vorgriff auf Kapitel 4 kann gesagt werden, dass in einer solchen Situation, in der die Werte aller Variablen des zu lernenden Bayes'schen Netzes in der Datensammlung beobachtet wurden, die bedingten Wahrscheinlichkeiten als so genannte *Maximum-Likelihood-Schätzungen* in Form der relativen Häufigkeiten „ausgezählt" werden können (siehe z.B. Buntine, 1996). Das „Lernverfahren" ist in einem solchen Fall also sehr einfach. In einem weiteren Schritt zur Konstruktion des Einflussdiagramms wird die Variable PRÄSENTATIONSMODUS in eine Entscheidungsvariable umgewandelt. Zur Bewertung der beiden Optionen *gebündelt* und *schrittweise* muss ein Bewertungsknoten samt Bewertungsfunktion eingeführt werden. Die Bewertung einer Situation ist abhängig davon, ob ein Fehler während der Ausführung der Instruktionen gemacht wurde und wie schnell die Anweisungsfolge bearbeitet wurde. Deshalb wird der Bewertungsknoten BEWERTUNG als Kind der beiden Knoten AUSFÜHRUNGSGESCHWINDIGKEIT und FEHLER? in die Struktur eingebaut. Zusätzlich hängt die Bewertung auch vom relativen Gewicht der Vermeidung von Fehlern gegenüber einer schnelleren Ausführungsgeschwindigkeit ab. Deshalb wird ein weiterer (Zufalls-)Knoten RELATIVES GEWICHT als Elternteil des Bewertungsknotens integriert. Mit ihm kann z.B. modelliert werden, dass zur Einsparung von 1 Sekunde (= 1000 msec) Ausführungszeit ein Fehler bei der Ausführung in Kauf genommen wird (bei Instantiierung des ersten Zustandes des Knotens). Abbildung 2.9 zeigt die Struktur des resultierenden Einflussdiagramms. Da in diesem Beispiel lediglich eine einzige Entscheidung betrachtet wird, kann auf die notwendigen Kanten zur Modellierung temporaler Beziehungen zwischen mehreren aufeinander folgenden Entscheidungen bzw. der sie repräsentierenden Entscheidungsknoten verzichtet werden.

Dieses komplett spezifizierte Einflussdiagramm kann genutzt werden, um alle möglichen Situationen und potenziellen Entscheidungen hinsichtlich des besten Präsentationsmodus zu evaluieren. Die dazu benötigten Evaluationsalgorithmen werden beispielsweise von Shachter (1986) sowie Jensen, Jensen und Dittmer (1994) beschrieben. Will man z.B. eine aus drei Einzelinstruktionen bestehende Anweisungsfolge geben während der Benutzer zusätzlich eine ablenkenden Nebenaufgabe bearbeitet, dann werden im Rahmen des Evaluationsprozesses mit Hilfe des zugrunde liegenden Bayes'schen Netzes Vorhersagen bezüglich der Fehlerwahrscheinlichkeit und der Ausführungsgeschwindigkeit ermittelt, die wiederum unter Verwendung der Bewertungsfunktion und unter Berücksichtigung des relativen Gewichtes zur Bewertung der beiden Alternativen einer schrittweisen bzw. gebündelten Präsentation genutzt werden. In der in Abbildung 2.9 dargestellten Modellierung des angesprochenen Beispiels wird die gebündelte Darbietung der Anweisungen der schrittweisen Variante vorgezogen (in der Abbildung durch die höhere Bewertung -5293.94 gegenüber -7647.01 gekennzeichnet).

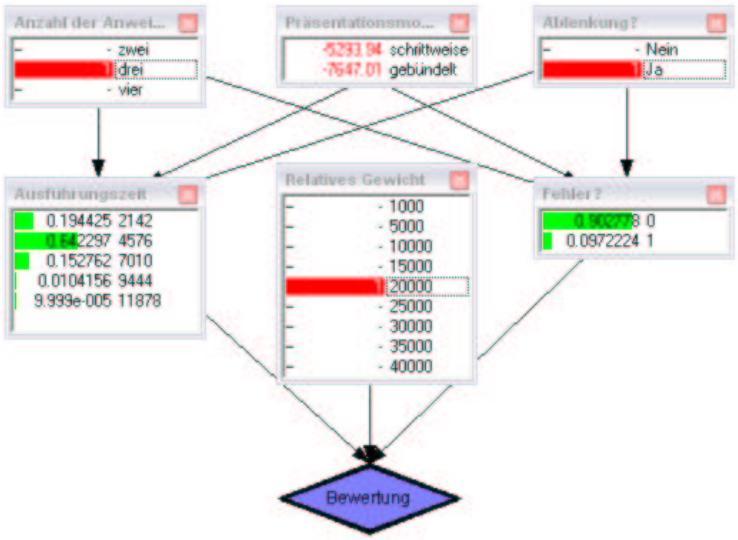


***Abbildung 2.9**: Beispiel eines Einflussdiagramms zur Modellierung des Anweisungsexperiments (Bildschirmabzug unter Verwendung des* HUGIN*-Systems. Rote (dunkle) Balken repräsentieren Evidenzen, grüne (helle) Balken stellen berechnete Wahrscheinlichkeitswerte dar.)*

Es existieren Verfahren, die eine vollständige *Policy* berechnen, d.h., eine Liste bestehend aus optimalen Entscheidungen für jede Zustandskombination der beobachtbaren Variablen. Tabelle 2.2 zeigt eine solche Policy für die Variable PRÄSENTATIONSMODUS, für den Fall, dass jeweils die Zustände der beiden Variablen ANZAHL DER ANWEISUNGEN und ABLENKUNG? bekannt sind. In diesem Beispielszenario wird jeweils eine binäre Entscheidung getroffen, d.h., entweder alle Instruktionen in einem Block zu geben oder alle einzeln. Man kann sich aber auch vorstellen, beispielsweise zuerst einen Block von zwei Anweisungen gebündelt zu präsentieren, gefolgt von zwei Einzelanweisungen. Ein Verfahren, mit dem solche differenzierten Anweisungsfolgen ermittelt werden können, basiert auf *Markov-Entscheidungsprozessmodellen* (engl. *Markov decision processes, MDPs)* und wird von Bohnenberger und Jameson (2001) sowie Jameson et al. (2001) vorgestellt.

Anweisungen	**ohne Ablenkung**		**mit Ablenkung**	
	rel. Gewicht	**Präsentationsmodus**	**rel. Gewicht**	**Präsentationsmodus**
2	≥ 1	gebündelt	1 - 10	gebündelt
			> 10	schrittweise
3	1 - 30	gebündelt	1 - 5	gebündelt
	> 30	schrittweise	> 5	schrittweise
4	1 - 5	gebündelt	1	gebündelt
	> 5	schrittweise	> 1	schrittweise

***Tabelle 2.2**: Mit einem erlernten Einflussdiagramm ermittelte Policy für das Anweisungsexperiment*

2.4 Dynamische Bayes'sche Netze

Ein *dynamisches Bayes'sches Netz* (Dagum et al., 1992) ist formal betrachtet ein Spezialfall eines „normalen" Bayes'schen Netzes im Sinn von Definition 2.1. Mit dynamischen Bayes'schen Netzen ist es möglich, zeitlich veränderliche Aspekte der zu modellierenden Domäne explizit zu repräsentieren. Diese Eigenschaft ist von großer Bedeutung für die Verwendung dynamischer Bayes'scher Netze in benutzeradaptiven Systemen, um während der Interaktion veränderliche Eigenschaften, Ziele usw. des Benutzers berücksichtigen zu können (siehe Schäfer & Weyrath, 1997; Schäfer, 1998). Dynamische Bayes'sche Netze werden in vielen verschiedenen Anwendungsszenarien eingesetzt, u.a. zur automatischen Steuerung von Fahrzeugen (Nicholson & Brady, 1994; Forbes, Huang, Kanazawa & Russell, 1995), zur Analyse des Gehverhaltens älterer Personen (Nicholson, 1996) und genetischer Daten (Murphy & Mian, 1999).

2.4.1 Aufbau

Grundlage eines dynamischen Bayes'schen Netzes sind *Zeitscheiben*, die diskrete Zeitpunkte der Modellierung repräsentieren. Eine Zeitscheibe besteht aus einem „normalen" Bayes'schen Netz, das durch zusätzliche Informationen hinsichtlich des Übergangs von einer Zeitscheibe zur nächsten ergänzt wird. Dabei modelliert eine der Zeitscheiben eines dynamischen Bayes'schen Netzes den aktuellen Zeitpunkt, die anderen repräsentieren zurückliegende bzw.—falls erforderlich—zukünftige Zeitpunkte (vgl. Abbildung 2.10).

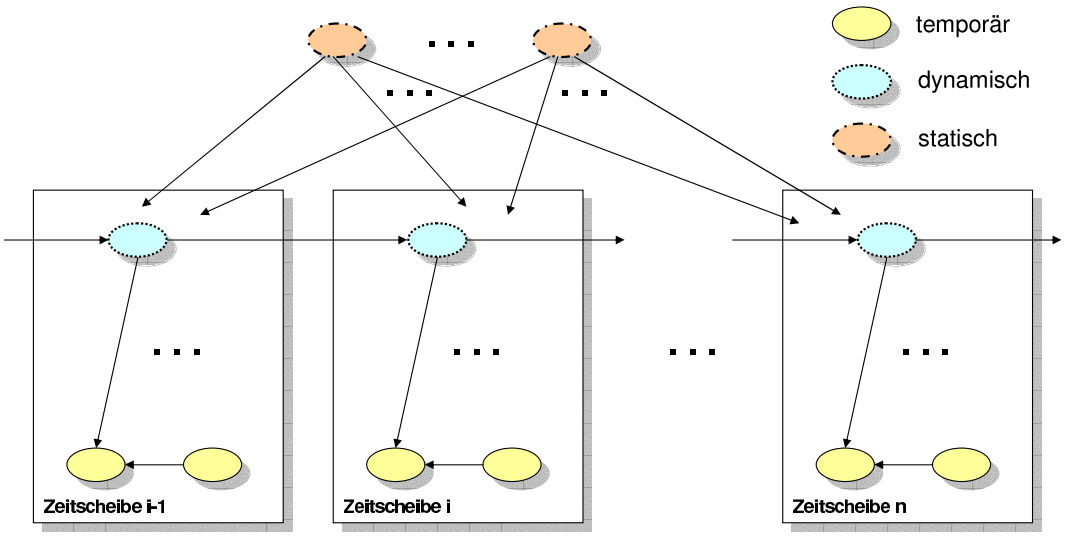


***Abbildung 2.10**: Dynamisches Bayes'sches Netz (Prototypische Darstellung)*

Ein dynamisches Bayes'sches Netz wird sequentiell aufgebaut, d.h., nacheinander werden neue Zeitscheiben an das aktuelle dynamische Bayes'sche Netz angehängt. Um zu vermeiden, dass die immer länger werdende Kette von Zeitscheiben zu Komplexitätsproblemen führt, können *Roll-up*-Verfahren zum „Aufrollen" der wachsenden Netze angewendet werden, die ältere Zeitscheiben des Netzes „abschneiden" (siehe z.B. Kjærulff, 1995; Brandherm, 2000). Dabei muss beachtet werden, dass durch das Entfernen von Zeitscheiben ein Informationsverlust entstehen kann.

Normalerweise besitzen alle Zeitscheiben eines dynamischen Bayes'schen Netzes die gleiche Struktur und die gleichen CPTs. Allerdings existieren Ansätze—gerade auch hinsichtlich benutzeradaptiver Systeme—, die sowohl zeitlich veränderliche Strukturen sowie bedingte Wahrscheinlichkeiten verwenden (siehe z.B. Schäfer, 1998).

In einem dynamischen Bayes'schen Netz werden die Knoten in drei Typen eingeteilt (vgl. Abbildung 2.10):

1. *Temporäre Knoten:* Damit werden Knoten bezeichnet, die jeweils lediglich innerhalb einer einzigen Zeitscheibe eine Rolle spielen. Ihr Zustand wird entweder als Evidenz beobachtet oder aufgrund der Einflüsse der anderen Variablen der Zeitscheibe im Rahmen der Inferenz berechnet. Sie haben nur indirekt über andere Knoten Auswirkungen auf Knoten in anderen Zeitscheiben. Beispielsweise würde man in einem dynamischen Bayes'schen Netz für das Flughafenexperiment die Knoten zur Repräsentation der Sprachsymptome als temporäre Knoten modellieren. Eine Zeitscheibe des Netzes würde in dieser Weise einer Äußerung der Versuchsperson entsprechen.

2. *Dynamische Knoten:* Dynamische Knoten repräsentieren Variablen, deren Zustand sich zeitlich verändern kann. Typischerweise befindet sich in jeder der Zeitscheiben eine Instanz des dynamischen Knotens (gleichen Namens), der den jeweiligen der Zeitscheibe entsprechenden Zustand des Knotens modelliert. Beispielsweise kann sich die kognitive Belastung des Benutzers erhöhen, wenn er eine neue zusätzliche Aufgabe bearbeiten muss. Andererseits sollte sie sich verringern, wenn eine der Aufgaben wegfällt. Im Flughafenexperiment könnte die Versuchperson z.B. die kognitive Belastung dadurch reduzieren, indem sie stehen bleibt, d.h., die Bearbeitung der Navigationsaufgabe einstellt bzw. unterbricht. Dieser Knotentypus dient explizit zur Modellierung dynamischer Aspekte der Domäne.

3. *Statische Knoten:*[12] Einen Sonderfall stellen Variablen dar, deren Wert sich im Verlauf der Existenz des dynamischen Bayes'schen Netzes nicht verändert. Ein Beispiel dafür sind die Vorlieben eines Benutzers, die sich—wenn überhaupt—nur in sehr großen Zeiträumen verändern. In einem solchen Fall kann man die zugehörige Variable für den relevanten Zeitraum der Modellierung als statisch annehmen. Der entsprechende Knoten existiert nur ein einziges Mal im dynamischen Bayes'schen Netz und ist in der Struktur des dynamischen Bayes'schen Netzes außerhalb der Zeitscheiben angesiedelt. Eine solche statische Eigenschaft einer Versuchsperson des Flughafenexperiments ist z.B. die Eigenschaft, üblicherweise schnell zu sprechen. Ein entsprechender statischer Knoten des Modells kann dazu dienen, die aktuell beobachtete Artikulationsgeschwindigkeit im Schlussfolgerungsprozess besser bewerten zu können.

Zeitlich bedingte Veränderungen werden durch *Übergangs-CPTs* modelliert, die mit den Kanten assoziiert sind, welche zwischen zwei dynamischen Knoten aufeinander folgender Zeitscheiben angesiedelt sind. So kann man beispielsweise durch entsprechende Festlegung der Übergangs-CPTs zwischen mehreren Instanzen des dynamischen ZEITDRUCK?-Knotens eine Situation repräsentieren, in der der Zeitdruck des Benutzers im Verlauf der Interaktion mit dem System im-

[12] In der ursprünglichen Definition dynamischer Bayes'scher Netze existieren keine explizit statischen Knoten. Sie können dort dennoch modelliert werden, indem sie als dynamische Knoten mit Übergangs-CPTs deklariert werden, die keine Veränderung der Werte bewirken. Gerade im Benutzermodellierungskontext, wo statische Eigenschaften der Benutzer modelliert werden müssen, erscheint die Verwendung explizit statischer Knoten sinnvoll.

mer mehr zunimmt. Dazu werden die Übergangs-CPTs derart spezifiziert, dass sich die Wahrscheinlichkeit eines stärkeren Zeitdrucks in der neuen Zeitscheibe im Vergleich zur derjenigen der alten erhöht. Gleichzeitig können aber in der Domäne Evidenzen zu temporären Knoten in widersprüchlicher Art und Weise dafür sprechen, dass sich der Zeitdruck verringert hat. Dies kann durch Anwenden der Inferenzalgorithmen im Netz berücksichtigt werden. Sind diese Evidenzen der temporären Knoten „stark" genug, dann wird sich die Wahrscheinlichkeit eines geringeren Zeitdrucks erhöhen, obwohl die Übergangs-CPTs des Netzes in entgegengesetzter Weise auf einen erhöhten Zeitdruck hinwirken. Ähnlich wie bei den Zeitscheiben werden im Standardansatz dynamischer Bayes'scher Netze die gleichen Übergangs-CPTs im kompletten Netz verwendet, wobei es auch hier Ansätze gibt, die dies flexibler gestalten (siehe z.B. Schäfer, 1998).

Das folgende Beispiel eines dynamischen Bayes'schen Netzes im Rahmen des Flughafenexperiments soll den Aufbau sowie die Verwendung dieses Werkzeugs zur Modellierung temporaler probabilistischer Eigenschaften einer Domäne verdeutlichen.

2.4.2 Beispiel: Erkennung kognitiver Ressourcenbeschränkungen anhand Symptomen gesprochener Sprache

Eines der Ziele, das mit dem Flughafenexperiment verfolgt wurde, ist es, zu untersuchen, inwieweit man aufgrund von Merkmalen der gesprochenen Sprache eines Benutzers in der Lage ist, Rückschlüsse über eventuell vorhandene Beschränkungen der kognitiven Ressourcen zu ziehen. Zu diesem Zweck wurde die in diesem Abschnitt beschriebene Fallstudie unter Verwendung eines dynamischen Bayes'schen Netzes durchgeführt.[13]

Abbildung 2.11 zeigt eine Zeitscheibe des verwendeten dynamischen Bayes'schen Netzes, die jeweils aufeinander folgende Äußerungen einer Versuchsperson repräsentiert. Als Grundlage der Struktur dient eine leicht variierte Version derjenigen aus Abbildung 2.7. Eine Zeitscheibe besteht in diesem Fall nur aus temporären Variablen: einerseits den zum entsprechenden Zeitpunkt beobachteten Sprachsymptomen und andererseits aus der Schwierigkeit der Frageformulierung. Die Zustände dieser beiden temporären Variablen sind immer nur zum durch die Zeitscheibe modellierten Zeitpunkt relevant, d.h., während bzw. nach der Formulierung einer einzigen Äußerung. Die beiden Variablen, die die experimentelle Bedingung repräsentieren, werden in diesem Beispiel als statisch angenommen, d.h., man geht davon aus, dass sich die experimentelle Bedingung, die durch die Verwendung des dynamischen Bayes'schen Netzes eingeschätzt werden soll, nicht ändert. Deshalb sind diese beiden unabhängigen Variablen außerhalb der Zeitscheiben angesiedelt und besitzen ausgehende Kanten in alle Zeitscheiben hinein; genauer gesagt, zu jeder temporären Variable, die ein Sprachsymptom repräsentiert. Zusätzlich werden hier *individuelle Parametervariablen* in die Struktur integriert, die charakteristische Eigenschaften des individuellen Benutzers darstellen, wie z.B. seine durchschnittliche Artikulationsgeschwindigkeit. Es ist allgemein bekannt, dass einige Personen generell sehr schnell reden, wohingegen andere wiederum recht langsam artikulieren. Die Werte dieser Variablen können prinzipiell sehr einfach als Durchschnitt der entsprechenden Werte einer Versuchsperson über das komplette Experiment mit allen Bedingungen ermittelt werden. Diese individuellen Parametervariablen haben einen kausalen Einfluss auf die zugehörigen temporären Symptomvariablen und werden im dynamischen Bayes'schen Netz als statische Variablen klassifiziert. Mit ihrer Hilfe soll eine verbesserte Anpassung an den Benutzer ermöglicht werden. Stehen dem System solche Informationen über individuelle Eigenschaften zur Verfügung,

[13] In Kapitel 7 werden ähnliche Studien desselben Szenarios vorgestellt, die sich u.a. hinsichtlich der verwendeten Netz-Strukturen unterscheiden.

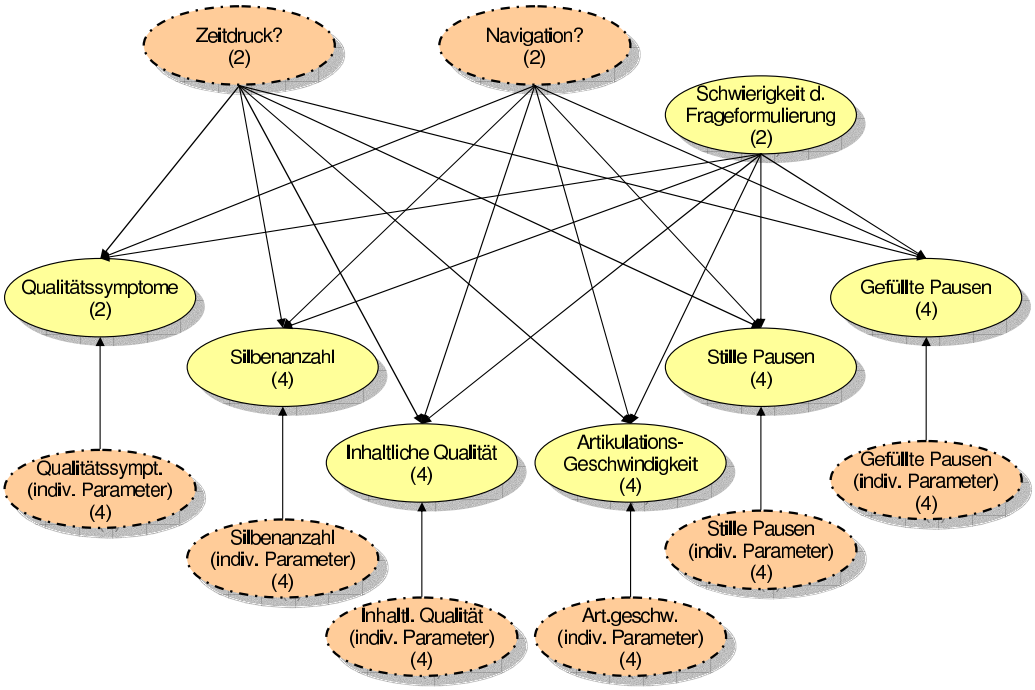


***Abbildung 2.11**: Beispiel einer Zeitscheibe eines dynamischen Bayes'schen Netzes zur Modellierung des Flughafenexperiments*
(Die Kennzeichnung der Knotentypen entspricht derjenigen aus Abbildung 2.10. Die Zahlen in den Klammern geben die Anzahlen der Zuständen der jeweiligen Variablen an. Aufgrund der Modellierungsannahmen (siehe Text) existieren in dieser Zeitscheibe keine dynamischen Knoten)

dann können vorliegende Evidenzen deutlich differenzierter interpretiert werden, was letztendlich zu einer erhöhten Genauigkeit der Modellierung führen sollte. Entsprechende Ergebnisse werden in Abschnitt 7.1.3 vorgestellt.

Die benötigten (bedingten) Wahrscheinlichkeiten der CPTs können prinzipiell wie in Abschnitt 2.3 als Maximum-Likelihood-Schätzungen in Form der (relativen) Häufigkeiten ermittelt werden. Dies gilt auch für die zwar nicht explizit in der Experimentalumgebung beobachteten, aber auf Basis der gesammelten Daten leicht zu berechnenden, individuellen Parameter.

Da überprüft werden sollte, ob und gegebenenfalls wie gut die Erkennung eventueller Ressourcenbeschränkungen einer Versuchsperson in einer der experimentellen Situationen möglich ist, wurde eine 32-fache *Leave-one-out-Kreuzvalidierung* durchgeführt (vgl. auch Abschnitt 3.1.3.8). Das bedeutet, die Daten von 31 Versuchspersonen werden genutzt, um ein allgemeines Benutzermodell in Form einer allgemein gültigen Zeitscheibe (inklusive der statischen Variablen) wie beschrieben zu erlernen. Dieses allgemeine Benutzermodell wird anschließend verwendet, um Vorhersagen über die experimentelle Bedingung zu machen, der die verbleibende 32. Versuchsperson ausgesetzt war, als sie die zu interpretierenden Äußerungen (inklusive der aufgetretenen Sprachsymptome) produzierte. Alle 20 Äußerungen einer Versuchsperson in einer der experimentellen Bedingungen werden sequentiell wie sie im Experiment auftraten im dynamischen Bayes'schen Netz durch Anfügen neuer Zeitscheiben verarbeitet. Dazu werden jeweils die aufgetretenen Sprachsymptome durch entsprechende Instanziierung der entsprechenden Symptomvaria-

blen berücksichtigt. Weiterhin werden die allgemeinen Tendenzen einer Versuchsperson durch Instanziierung der individuellen Parametervariablen in den Inferenzprozess eingebracht. Mit dieser Vorgehensweise sollte das dynamische Bayes'sche Netz in der Lage sein, sukzessive die Einschätzung der kognitiven Ressourcenbeschränkungen bzw. experimentellen Bedingungen zu verbessern. Dieses Verfahren wurde für alle 32 möglichen Kombinationen von Lern- und Testdaten durchgeführt. Die im Folgenden präsentierten Ergebnisse stellen die zugehörigen Durchschnittswerte separat für jede der vier experimentellen Bedingungen dar. Tabelle 2.3 fasst das Analyseverfahren bezüglich einer Versuchsperson kompakt in übersichtlicher Form zusammen.

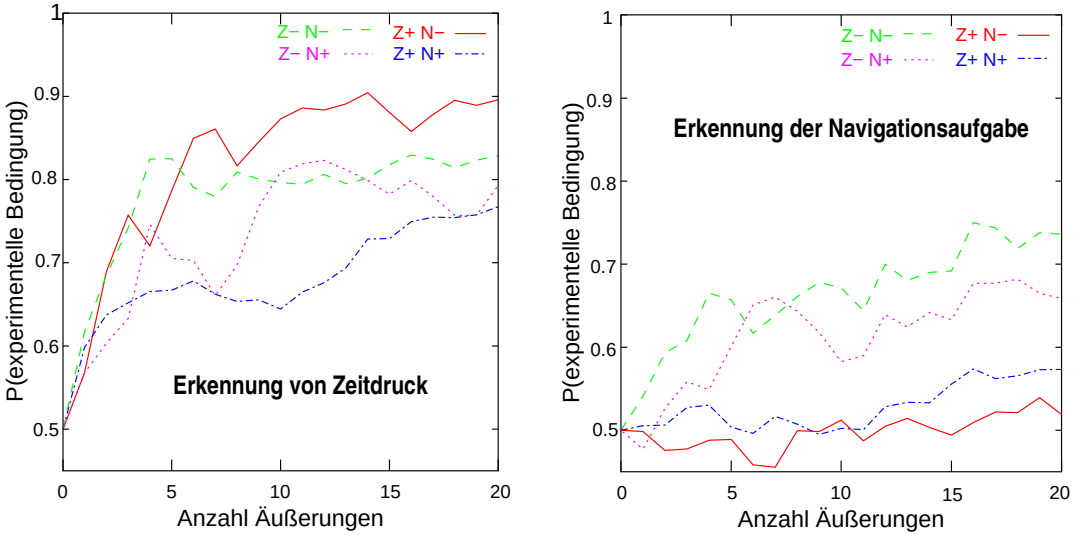


***Abbildung 2.12**: Erkennungsleistung des dynamischen Bayes'schen Netzes zur Erkennung der experimentellen Bedingungen im Flughafenexperiments*
(Z+ / Z-: Zeitdruck vorhanden / nicht vorhanden, N+ / N-: Navigationsaufgabe vorhanden / nicht vorhanden. Die Erkennungsleistung wird als die von dynamischen Bayes'schen Netz ermittelte Wahrscheinlichkeit der tatsächlich vorliegenden Teilbedingung P(experimentelle Bedingung) gemessen.)

Abbildung 2.12 zeigt die Ergebnisse separat für jede der vier experimentellen Bedingungen aufgeschlüsselt nach der Erkennungsleistung hinsichtlich der beiden explizit manipulierten unabhängigen Variablen ZEITDRUCK? und NAVIGATION?.

Die Erkennungsleistung des dynamischen Bayes'schen Netzes bezüglich der Variablen ZEITDRUCK? (linker Graph) ist relativ hoch. Die korrekte experimentelle Teilbedingung (kein Zeitdruck oder Zeitdruck vorhanden) wird schon nach wenigen (ca. 4) Äußerungen mit einer Wahrscheinlichkeit zwischen 0.65 und 0.85 eingeschätzt. Nach Berücksichtigung aller 20 Äußerungen werden Werte etwa zwischen 0.75 und 0.90 erzielt. Insgesamt wird ZEITDRUCK? besser erkannt, wenn die Navigationsaufgabe nicht bearbeitet werden musste. Dieser Effekt kann damit erklärt werden, dass in dieser Situation die Versuchspersonen besser auf die geforderte Bedingung—schnelle bzw. ausführliche Frageformulierung—reagieren können, was sich in den entsprechenden Sprachsymptomen widerspiegelt.

Insgesamt wird die andere experimentelle Teilbedingung—der Zustand der Variablen NAVIGATION? (rechter Graph)—schlechter erkannt. Insbesondere im Fall vorliegenden Zeitdrucks ist das dynamische Bayes'sche Netz kaum in der Lage, Wahrscheinlichkeitswerte über dem Zufalls-

1. Relevante Variablen und ihre Werte
   - Eine Versuchsperson V
   - Werte z und n der binären Variablen Z (ZEITDRUCK?) und N (NAVIGATION?)
2. Aufgabe
   - Inferiere die Werte von Z und N auf der Basis von Sprachsymptomen in den Äußerungen von V
3. Vorbereitung der Testdaten
   - Wähle 20 Beobachtungen für V mit $Z = z$ und $N = n$, in der Reihenfolge, in der sie im Experiment auftraten
4. Evaluation der Erkennungsleistung
   (a) Initialisierung des Modells
      i. Erstelle die erste Zeitscheibe des dynamischen Bayes'schen Netzes für V
      ii. Instantiiere alle individuellen Parametervariablen mit den richtigen Werten für V. Lasse die Variablen Z und N uninstanziiert.
   (b) Für jede Evidenz E aus der Menge der Evidenzen für V
      i. In der neuesten Zeitscheibe des Netzes, leite Annahmen über Z und N ab
         - Instanziiere alle temporären Variablen für die Zeitscheibe mit ihren Werten in E
         - Wende die Inferenzverfahren an, um die Wahrscheinlichkeiten für Z und N zu erhalten
         - Speichere die Wahrscheinlichkeiten, die zu diesem Zeitpunkt mit den Zuständen von Z und N verbunden sind
      ii. Füge eine neue Zeitscheibe zum dynamischen Bayes'schen Netz hinzu (für die nächste Äußerung)

***Tabelle 2.3**: Prozedur zur Evaluation der Erkennungsleistung der erlernten dynamischen Bayes' schen Netze hinsichtlich Beschränkungen kognitiver Ressourcen mit den Daten des Flughafenexperiments*

niveau von 0.50 zu erzielen. Liegt kein Zeitdruck vor, so werden immerhin noch Werte zwischen 0.60 und 0.75 erreicht. Diese Beobachtung kann erklärt werden, indem man berücksichtigt, dass die Versuchspersonen unter Zeitdruck ihre Äußerungen auf ein Minimum reduzieren, womit auch eine Verringerung der absoluten Anzahl sprachlicher Symptome einhergeht. Aufgrund einer geringeren Anzahl an Symptomen wird es für das dynamische Bayes'sche Netz zunehmend schwieriger das (Nicht-)Vorhandensein der Navigationsaufgabe zu erkennen.[14]

[14] An dieser Stelle ist zu bemerken, dass es im Rahmen des Experimentaldesigns möglich gewesen wäre, die Erkennung der Navigationsaufgabe zu erleichtern, indem man ihre Komplexität erhöht hätte—möglicherweise bis zu einem Grad, an dem der Sprachproduktionsprozess der Versuchspersonen komplett zusammenbricht. Ziel des Experiments war es in diesem Fall allerdings, eine Nebenaufgabe zu betrachten, die in etwa der einer realistischen zusätzlichen

2.5 Objekt-orientierte Bayes'sche Netze und probabilistische relationale Modelle

Durch die Verwendung Bayes'scher Netze in immer komplexeren und umfangreicheren Domänen kommt es auch zu einer Erhöhung der Komplexität des Konstruktionsprozesses der verwendeten Netze. Zur Minimierung des Arbeits- und Wartungsaufwands werden ähnlich wie in traditionellen Software-Engineering-Projekten zunehmend auch im Knowledge-Engineering-Prozess der Konstruktion eines Bayes'schen Netzes objekt-orientierte Ansätze eingesetzt. Die Vorteile schließen, wie bei der objekt-orientierten Programmierung, u.a. die Wiederverwendbarkeit von Netzteilen, Default-Werte und Vererbbarkeit ein. Außerdem trägt die explizite Modellierung von Objekten inklusive ihrer Eigenschaften in Form von Klassen mit Attributen (als Teilnetze) und die zwischen ihnen bestehenden Beziehungen (Relationen) zur Interpretierbarkeit der resultierenden Modelle bei.

Ein weiterer wichtiger Aspekt der Anwendung Bayes'scher Netze in realistischen Szenarien besteht in der Möglichkeit im Rahmen eines solchen objekt-orientierten Ansatzes, *situationsspezifische* Netze zur Laufzeit des Systems unter Berücksichtigung der aktuellen Gegebenheiten und Anforderungen zu konstruieren (Laskey & Mahoney, 1997; Mahoney & Laskey, 1998). Es ist nicht notwendig, ein möglicherweise sehr komplexes Modell zu erstellen, das alle potenziellen Sachverhalte berücksichtigt. Anhand einer Bibliothek von *Netzfragmenten* können ad hoc aktuell relevante Fragmente kombiniert werden, um die gewünschten Schlussfolgerungen zu berechnen. Neben der größeren Flexibilität kann mit einem solchen Vorgehen sichergestellt werden, dass zu jedem Zeitpunkt ein möglichst minimales Netz im Inferenzprozess genutzt wird, was im Allgemeinen zu einer Verkürzung der Antwortzeiten führt.

Zusätzlich kann die objekt-orientierte Modellierung im Rahmen der Inferenzverfahren ausgenutzt werden, um beispielsweise wiederholt durchzuführende Berechnungen zu vermeiden (Koller & Pfeffer, 1997; Pfeffer, Koller, Milch & Takusagawa, 1999).

Langseth und Bangsø (2000) und Bangsø, Langseth und Nielsen (2001) stellen maschinelle Lernverfahren vor, die das in Form der objekt-orientierten Modellierung vorliegende Hintergrundwissen ausnutzen, um sowohl (Teil-)Strukturen als auch die bedingten Wahrscheinlichkeiten der Objekte zu lernen. Durch das zusätzliche, den Lernverfahren zur Verfügung stehende, strukturelle Wissen kann eine Performanzsteigerung des Lernvorgangs und der resultierenden Bayes'schen Netze im Vergleich zum unstrukturierten Lernen erzielt werden.

Den mächtigsten Ansatz, der auf einer objekt-orientierten Grundidee basiert, stellen *probabilistische relationale Modelle (PRMs)* (Koller & Pfeffer, 1998; Getoor, Friedman, Koller & Pfeffer, 2001) zur Modellierung relationaler Domänen dar. In vielen Unternehmen werden die Daten in relationalen Datenbanken abgelegt, d.h., die Daten werden in Form von Tabellen und ihren Beziehungen organisiert. PRMs übertragen diese auf der Basis der relationalen Algebra theoretisch fundierte Organisationsstruktur auf Bayes'schen Netze. Eine Klasse repräsentiert eine Tabelle inklusive ihrer Attribute. PRMs bieten die Möglichkeit, Unsicherheit sowohl hinsichtlich der Relationen zwischen den Klassen bzw. Instanzen als auch über die Menge der Instanzen des Modells zu repräsentieren. Diese Eigenschaften stellen einen entscheidenden Vorteil gegenüber „normalen" Bayes'schen Netzen dar, wo üblicherweise ein einziges Netz im Schlussfolgerungsprozess zur Verfügung steht, was bedeutet, dass in ihm alle Informationen zur Anzahl und den Beziehungen zwischen den Objekten der modellierten Domänen in der Struktur und den CPTs kodiert

Belastung entspricht.

werden müssen.

Im Kontext benutzeradaptiver Systeme bietet es sich beispielsweise an, die Benutzermodelle in entsprechender Weise zu verwalten. Ähnlich wie bei einem Stereotypen-Ansatz könnten unterschiedliche Benutzer*klassen* modelliert werden, die z.B. in ihrer Grundstruktur übereinstimmen, d.h. diese von einer übergeordneten Klasse ererben, andere Klassen- bzw Stereotypen-spezifische Eigenschaften können im Rahmen des Vererbungsprozesses die entsprechenden Werte der Elternklassen „überschreiben“. In komplexeren Domänen wie z.B. in einem mobilen Szenario mit einer Vielzahl potenziell nutzbarer Sensoren vereinfacht der Ansatz der probabilistischen relationalen Modelle durch die explizite Repräsentationsmöglichkeit von Relationen zwischen den Objekten den Modellierungsprozess. Es kann die Unsicherheit der Beziehungen zwischen den mit unterschiedlichen Sensoren beobachteten Evidenzen in expliziter Form im Modell dargestellt werden.

Die bereits entwickelten maschinellen Lernverfahren (Getoor et al., 2001) machen die PRMs zu einem vielversprechenden Werkzeug zur Erzeugung probabilistischer Modelle in einer Vielzahl von kommerziellen, potenziell sehr komplexen Domänen.

Im Zusammenhang mit dem Einsatz maschineller Lernverfahren können objekt-orientierte Ansätze zur Aufteilung der gesamten Lernaufgabe in kleinere, in sich abgeschlossene, lokale Teillernprobleme genutzt werden. Aufgrund der Komplexität bestimmter Verfahren (vgl. Kapitel 4) kann das Ausnutzen des in Form der objekt-orientierten Struktur modellierten Wissens über die Domäne entscheidend zur Anwendbarkeit von Lernverfahren beitragen. Auch stehen oft nur zu Teilen des Gesamtmodells empirische Daten zur Verfügung. Es ist dann beispielsweise möglich, einige Klassen des objekt-orientierten Bayes’schen Netzes zu erlernen, wohingegen andere manuell konstruiert werden.

Die in dieser Arbeit entwickelten Verfahren können zur Lösung solcher Teillernprobleme ohne Modifikation eingesetzt werden.

2.6 Stand der Forschung: Benutzeradaptive Systeme auf der Basis Bayes’scher Netze

Es existiert eine Vielzahl an Arbeiten im Bereich benutzeradaptiver Systeme, die (dynamische) Bayes’sche Netze als Grundlage ihrer Inferenzkomponente einsetzen. Jameson (1996) gibt eine frühe Übersicht über solche Systeme und vergleicht den Ansatz der Bayes’schen Netze zur Unsicherheitsbehandlung mit alternativen Techniken. Eine ähnliche, etwas aktuellere Diskussion mit dem Fokus der Behandlung temporaler Aspekte in benutzeradaptiven Systemen bietet Schäfer (1998). Diese beiden genannten Arbeiten repräsentieren den jeweiligen aktuellen Stand der Forschung der Verwendung Bayes’scher Netze in benutzeradaptiven Systemen.

Im Folgenden werden (in chronologischer Reihenfolge) neuere Arbeiten vorgestellt und ein Überblick über die Entwicklungen und Fragestellungen gegeben, die zur Zeit im Mittelpunkt der Forschung stehen. Dabei wird—dort wo es Sinn macht—ein besonderes Augenmerk auf die folgende Kriterien gelegt, die im Zusammenhang mit dieser Arbeit von verstärktem Interesse sind (vgl. Abschnitt 1.3):

- *Welche maschinellen Lernverfahren werden gegebenenfalls eingesetzt?*
- *Sind die verwendeten Bayes’schen Netze interpretierbar?*
- *Werden individuelle Unterschiede zwischen den Benutzern repräsentiert?*
- *Werden temporale Aspekte modelliert? Gegebenenfalls in welcher Form?*

- *Wird a priori vorhandenes Wissen in besonderer Form bei der Erstellung der Netze genutzt?*

2.6.1 Horvitz et al. (1998): LUMIÈRE

Im Rahmen des LUMIÈRE-Projekts (Horvitz et al., 1998) wurde ein Prototyp eines benutzeradaptiven Assistenzsystems zur Tabellenkalkulationssoftware EXCEL des MS OFFICE 97-Paketes der Firma MICROSOFT entwickelt. Es soll den Benutzer bei der Erledigung einer Vielzahl möglicher Aufgaben unterstützen, ohne dass eine explizite Anforderung der Hilfe seitens des Benutzers erfolgen muss. Dazu mussten im Wesentlichen zwei Teilprobleme gelöst werden: (a) die Erkennung der aktuellen Ziele des Benutzers, d.h., was will er mit den Aktionen, die er bisher ausgeführt hat, erreichen, und (b) die Ermittlung geeigneter Zeitpunkte, zu denen es sinnvoll erscheint, Hilfe anzubieten. Das bedeutet, es muss erkannt werden, ob der Benutzer Probleme hat, eine Aufgabe zu lösen, oder ob er suboptimale Vorgehensweisen anwendet, die er bei der Bearbeitung ähnlicher Aufgaben zukünftig vermeiden sollte.

Zur Lösung dieser Problemstellung werden dynamische Bayes'sche Netze in Kombination mit Einflussdiagrammen verwendet. Auf der Basis der zuletzt getätigten Aktionen des Benutzers (z.B. Suchen in der Menüstruktur, Rücknahme der letzten Aktion, u.Ä.) sowie dem aktuellen Systemzustand des Tabellenkalkulationssystems werden die Benutzerziele inferiert und mittels adäquater Situationsbewertungen die Einschätzung getroffen, ob der Benutzer zum betrachteten Zeitpunkt potenziell von einer aktiven Hilfestellung profitieren wird.

Zur Erhebung von Domänenwissen wurden in Zusammenarbeit mit Psychologen Benutzerstudien in Form von *Wizard-of-Oz*-Studien durchgeführt, die grundlegende Erkenntnisse hinsichtlich des Benutzerverhaltens in diesem Tabellenkalkulationssystem lieferten. Die in dieser Weise gewonnenen Einsichten wurden in die Strukturen und CPTs der dynamischen Bayes'schen Netze eingearbeitet. Maschinelle Lernverfahren kommen bei der Adaption der bedingten Wahrscheinlichkeiten zum Einsatz. Neben dieser Form der Individualisierung der Bayes'schen Netze existieren Variablen zur expliziten Modellierung individueller Eigenschaften der Benutzer, die als individuelle Parametervariablen im Sinne der Diskussion aus Abschnitt 2.4.2 angesehen werden können. Im Wesentlichen wird damit die Erfahrung der Benutzer im Umgang mit dem System erfasst. Zusätzlich zur Modellierung des zeitlichen Verlaufs der Interaktion unter Verwendung der dynamischen Bayes'schen Netze werden einige der temporalen Aspekte wie z.B. Pausen, die vom Benutzer bei der Interaktion mit dem System eingelegt werden, explizit durch Variablen im Modell abgebildet. Solche Pausen können beispielsweise darauf hindeuten, dass der Benutzer über sein weiteres Vorgehen nachdenken muss.

Ergebnisse aus der Entwicklung dieses Forschungsprototyps sind—wenn auch in deutlich vereinfachter Form—in das kommerzielle MS OFFICE97-Paket in Form der MS OFFICE 97 ASSISTENTEN eingeflossen (siehe Horvitz et al., 1998).

2.6.2 Albrecht et al. (1998): MUD-Spiele

Albrecht et al. (1998) verwendeten Bayes'sche Netze im Zusammenhang mit einer empirischen Studie in einer Multi-User-Dungeon-Spiele-Domäne (MUD). In einem MUD-Spiels geht es darum, dass mehrere Spieler gleichzeitig in einem gemeinsamen Szenario unterschiedliche Aufgaben zu lösen versuchen. Dabei müssen Teilaufgaben zur Erfüllung des Gesamtziels bearbeitet werden.

Ziel der Arbeit von Albrecht et al. war es, anhand verschiedener Informationen, die globalen Ziele und nächsten Aktionen eines Spielers vorherzusagen. Dabei wurden die Ziele der Spieler als

statisch, d.h., als für den Betrachtungszeitraum gleichbleibend, angenommen. Zur Erkennung der Ziele wurden dynamische Bayes'sche Netze aufgebaut, die als Evidenzen die Aktionen und Positionen des Spielers berücksichtigten. Die bedingten Wahrscheinlichkeiten dieser Netze wurden auf der Basis in der Domäne erhobener empirischer Daten in Form von Maximum-Likelihood-Schätzungen gelernt. Hauptziel dieser Studie war die Untersuchung der Performanz alternativer Strukturen (der Zeitscheiben) bei dieser Planerkennungsaufgabe. Es wurden relative einfache Strukturen verwendet, die lediglich der numerischen Genauigkeit dienten und keinerlei weitergehende Erklärungsaufgaben wahrnahmen. Außerdem erforderte die vorliegende Komplexität der Domäne (viele unterschiedliche Aufgaben, viele mögliche Positionen) einfache Strukturen für die Zeitscheiben, um die Analysen mit einem sinnvollen zeitlichen Aufwand durchführen zu können. Alle Spieler wurden mit den gleichen Modellen behandelt, es wurden keine individuellen Unterschiede im Spielverhalten berücksichtigt.

2.6.3 Billsus und Pazzani (1999): NEWSDUDE

NEWSDUDE (Billsus & Pazzani, 1999) ist ein persönliches Assistenzsystem zur täglichen Zusammenstellung von den Interessen des Benutzers entsprechenden Nachrichtenartikeln aus verschiedenen Quellen im WWW. Ein Schwerpunkt dieses Projekts war es, sowohl zeitliche Veränderungen der Benutzerinteressen als auch die Tatsache zu berücksichtigen, dass dem Benutzer gewisse Informationen manchmal bereits bekannt sind (wie es z.B. durch einen bereits vom Benutzer gelesenen Artikel, der sich mit dem gleichen Thema beschäftigt, der Fall ist).

Zur Erfassung dieser zeitlichen Aspekte in dieser Domäne werden ein *Langzeit*- und ein *Kurzzeit*-Benutzermodell unterschieden. Ersteres modelliert die eher allgemeinen, sich langsamer verändernden Benutzerinteressen, wie z.B. ein allgemeines Interesse an globalen Kategorien wie Sport, Politik, usw. Das Kurzzeit-Benutzermodell repräsentiert hingegen das Interesse an verwandten (Folge-)Artikeln zu speziellen Ereignissen—in einem begrenzten Zeitraum. Die Autoren konnten zeigen, dass dieses hybride Benutzermodell eine Verbesserung gegenüber den Einzelmodellen erzielen konnte. Beide Modelle werden anhand expliziter Rückmeldungen zu den Artikeln (im Wesentlichen 'interessant' / 'nicht interessant') an die Interessen des Benutzers adaptiert.

Bayes'sche Netze kommen im Rahmen des Langzeit-Benutzermodells zum Einsatz. Bei der Bewertung der Nachrichten handelt es sich um eine binäre *Klassifikationsaufgabe*: Ist die Nachricht für den Benutzer von Interesse oder nicht? Das hier—wie auch sonst häufig—benutzte Klassifikationsverfahren ist der naive Bayes'sche Klassifizierer (siehe Abschnitt 2.1.4)

In NEWSDUDE werden als Merkmale Schlüsselwörter der Nachrichtenartikel verwendet. Ist ein solches Schlüsselwort in einem Artikel enthalten, dient dies als Evidenz für die entsprechende Merkmalsvariable des naiven Bayes'schen Klassifizierers. Ob ein Artikel potenziell für den Benutzer von Interesse ist, kann somit durch Interpretation der Evidenzen, d.h., des Auftretens der Schlüsselwörter, im naiven Bayes'schen Klassifizierer anhand der resultierenden Wahrscheinlichkeitsverteilung der Klassenvariable mit den Zuständen *interessant* und *nicht interessant* ermittelt werden.

Die bedingten Wahrscheinlichkeiten des initialen naiven Bayes'schen Klassifizierers werden anhand von Bewertungen von Nachrichtenartikel erlernt, die der Benutzer bei Beginn der Nutzung des Systems abgeben muss. Als Lern- bzw. Adaptionverfahren kommt in diesem System das Standardvorgehen des Bayes'schen Lernansatzes zum Einsatz (siehe Abschnitt 4.3). Damit werden sukzessiv individuelle Benutzermodelle erlernt, die die Interessen des jeweiligen Benutzers widerspiegeln. Die Entwickler dieses Systems nutzten vorhandenes Hintergrundwissen aus, indem sie

bei der Auswahl der verwendeten Merkmalsvariablen des naiven Bayes'schen Klassifizierers eine Liste aussagekräftiger Schlüsselwörter unter Berücksichtigung der einzelnen Kategorien wie Politik, Sport usw. wählten. Die Interpretierbarkeit der Benutzermodelle spielt in NEWSDUDE keine Rolle, was sich in der von den Entwicklern getroffenen Wahl der Methoden zur Repräsentation der Benutzermodelle (nächste Nachbarn, naiver Bayes'scher Klassifizierer) ausdrückt.

2.6.4 Lau und Horvitz (1999): WWW-Suchanfragen

Einen Ansatz zur Modellierung des Benutzerverhaltens im Zusammenhang mit einer WWW-Suchmaschine beschreiben Lau und Horvitz (1999). Sie konstruieren Benutzermodelle, die verwendet werden können, um zeitabhängig die nächste Aktion des Benutzers und seine Ziele vorherzusagen. Diese Information kann beispielsweise genutzt werden, um etwa entsprechende Suchanfragen schon frühzeitig—vor expliziter Anforderung durch den Benutzer—anzustoßen und um eventuell gezielt auf den Benutzer zugeschnittene Werbung auf den Resultatsseiten zu platzieren.

Es werden Bayes'sche Netze eingesetzt, um auf der Basis von Evidenzen wie der expliziten Modellierung der verstrichenen Zeit seit der letzten Anfrage, der Anzahl der Suchbegriffe und der Art der letzten Anfrage (neue Anfrage, Verfeinerung / Verallgemeinerung der Anfrage, Umformulierung, Anforderung zusätzlicher Ergebnisse, Unterbrechung der Anfrage durch eine andere Anfrage) eine Wahrscheinlichkeitsverteilung über die möglichen nächsten Aktionen des Benutzers zu bestimmen. Intuitiv plausibel erscheint es beispielsweise, dass nach einer Pause, die eine bestimmte Dauer (z.B. 20 Minuten) seit der letzten Anfrage überschreitet, die wahrscheinlichste nächste Aktion das Stellen einer neuen Anfrage ist. Zusätzlich wurde von den Autoren das Einbringen einer inhaltlichen Kategorisierung der Anfragen (z.B. Unterhaltung, Sport, Politik usw.) in die Benutzermodelle und die Vorhersage der Kategorien auf Basis der genannten Informationen betrachtet.

Die bedingten Wahrscheinlichkeiten der CPTs der Benutzermodelle in Form Bayes'scher Netze wurden anhand empirischer Daten bestehend aus (semi-manuell) aufbereiteten Log-Dateien der Suchmaschine als Maximum-Likelihood-Schätzungen maschinell gelernt. Dabei wurde der komplette Datensatz ohne eine differenziertere Betrachtungsweise—beispielsweise nach Benutzergruppen—als Grundlage des Lernverfahrens genutzt. Es werden keine individuellen Unterschiede modelliert. Hintergrundwissen floss in die manuelle Spezifikation der interpretierbaren Strukturen der Bayes'schen Netze und in den Datenaufbereitungsprozess insbesondere bei der Diskretisierung in Zeitintervalle und der thematischen Einordnung der Suchanfrage ein.

2.6.5 Conati und VanLehn (1999): Selbsterklärungen

Conati und VanLehn (1999, 2001) stellen ein intelligentes Lehr-/Lernsystem für physikalische Formeln vor, das zur Modellierung des Wissenstands des Lernenden dynamisch aufgebaute Bayes' sche Netze nutzt.

Die in diesem System angewandte Lernstrategie basiert auf *Selbsterklärungen* des Lernenden. Das System beobachtet anhand der Interaktion des Benutzers mit der Lernumgebung (Menüauswahl, Betätigen von Buttons u.Ä.), wie der Lernende sich die Funktionsweise und Anwendung einer Formel unter Zuhilfenahme der Werkzeuge der Systemumgebung erklärt. Auf der Basis von Informationen über das vorhandene Vorwissen und weiteren Evidenzen, wie etwa der Dauer der Bearbeitung einer Aufgabe, werden unter Verwendung der Bayes'schen Netze gegebenenfalls Verbesserungsvorschläge zur Anwendung der Formeln ermittelt und präsentiert.

Dazu werden im verwendeten Bayes'schen Netz mehrere Variablentypen unterschieden, die potenzielle Evidenzen, potenziell anwendbare Regeln zum Umgang mit den Formeln, tatsächliche Anwendungen dieser Regeln und (Teil-)Ziele, die der Lernende verfolgen kann, modellieren. So können Schlussfolgerungen über die vorhandenen Kenntnisse zur korrekten Anwendung der Regeln und Formeln gezogen werden, um gegebenenfalls Verbesserungen zur Strategie des Lernenden vorzuschlagen.

Im Verlauf mehrerer Interaktionsphasen zwischen System und Benutzer wird ein Langzeitbenutzermodell erstellt, das die individuellen Eigenschaften der Lernenden aggregiert. Diese Information wird bei der aufgabenspezifischen Konstruktion der Bayes'schen Netze in Form von individuellen Parametervariablen eingebracht.

2.6.6 Horvitz et al. (1999 – 2002): Situative Benachrichtigungen, COORDINATE

Viele Systeme teilen ihren Benutzern potenziell kritische Sachverhalte durch explizite Benachrichtigungen oder Alarme z.B. durch plötzlich auf dem Bildschirm erscheinende, von einem akustischen Signal begleitete Fenster mit. Dadurch kann der Benutzer von seiner eigentlichen Arbeit mit einer anderen (Teil-)Funktionalität des Gesamtsystems oder gar einem anderen System in unerwünschter Weise abgelenkt werden. Eine Nachricht über eine neu eingegangene (Spam-)E-Mail erscheint einem Benutzer möglicherweise als störende Ablenkung während er an einem komplizierten Text (z.B. seiner Dissertation) arbeitet.

Horvitz, Jacobs und Hovel (1999) gehen diese Problematik mittels eines entscheidungstheoretischen Ansatzes unter Verwendung von Einflussdiagrammen an. Sie benutzen dynamische Bayes' sche Netze, um den zeitlichen Verlauf des situativen Kontexts zu verfolgen. Wie in Abschnitt 2.3 beispielhaft veranschaulicht, wurden diese dynamischen Bayes'schen Netze zu Einflussdiagrammen erweitert, die nicht nur entscheiden, ob eine Nachricht mitgeteilt oder unterdrückt werden soll, sondern auch in einem weitergehenden Ansatz, (a) in welcher Modalität (sprachlich, graphisch, textuell) und (b) zu welchem Zeitpunkt die Nachricht möglichst optimal unter Berücksichtigung der potenziellen kognitiven Kosten dem Adressaten zu präsentieren ist. Dabei spielen Faktoren wie aktuell anstehende Termine (aus dem Kalender des Adressaten zu ermitteln), die Tageszeit, Umgebungsgeräusche (durch Sensoren) und der Status der verwendeten Software (z.B. Textverarbeitungsprogramm) eine Rolle. Im Wesentlichen wird der geschätzte Wert der Information (engl. *value of information*, siehe z.B. Russell & Norvig, 1995) gegen die durch den situativen Kontext induzierten kognitiven Kosten einer Benachrichtigung abgewogen. Maschinelle Lernverfahren für Bayes'sche Netze kommen hier (noch) nicht zum Einsatz.

In der Weiterentwicklung COORDINATE (Horvitz, Koch, Kadie & Jacobs, 2002) des beschriebenen Systems wurden Komponenten integriert, die Voraussagen über die Verfügbarkeit des Benutzers für gemeinsame Aktivitäten wie Videokonferenzen, gemeinsame Dokumentenbearbeitung oder Meetings anhand des situativen Kontexts liefern. Dazu wird eine Vielzahl an Daten gesammelt, die bei Bedarf genutzt werden, eine der aktuellen Situation entsprechende Stichprobe der Daten auszuwählen. Auf Basis dieser Stichprobe werden Bayes'sche Netze zur Laufzeit des Systems—sowohl Struktur als auch die zugehörigen bedingten Wahrscheinlichkeiten—erlernt, mit deren Hilfe anhand des aktuellen Kontexts Wahrscheinlichkeitsverteilungen über die aktuelle Verfügbarkeit bzw. den möglichst optimalen Zeitpunkt der gemeinsamen Aktivität gemacht werden. Die so erlernten Netze stellen Modelle dar, die Zusammenhänge zwischen verschiedenen typischen Verhaltensweisen des Benutzers abbilden können. Beispielsweise kann von der Komponente des Systems, die für die automatische Beantwortung der eingehenden E-Mail verantwortlich

ist, in wichtigen Fällen eine Antwort generiert und verschickt werden, die bei kurzfristiger Abwesenheit des Adressaten den aufgrund der in Form der Daten gesammelten Erfahrungen geschätzten Rückkehrzeitpunkt mitteilt.

In der Arbeit werden nur individuelle Benutzermodelle eingesetzt. Die Interpretierbarkeit der erlernten Bayes'schen Netze wird nicht gefordert, da in dem System (bislang) keine Komponente vorgesehen ist, die die getroffenen Entscheidungen begründet, was aber durchaus zumindest am Beispiel der automatischen E-Mail-Beantwortung Sinn machen könnte.

2.6.7 Paek und Horvitz (1999 – 2001): BAYESIAN RECEPTIONIST, DEEPLISTENER

Mit den beiden Forschungsprototypen BAYESIAN RECEPTIONIST und DEEPLISTENER (Horvitz & Paek, 1999; Paek & Horvitz, 2000; Horvitz & Paek, 2001) untersuchen die beiden Autoren, inwieweit es möglich ist, mit Hilfe dynamischer Bayes'scher Netze die Performanz eines adaptiven Dialogsystems durch Kombination von Evidenzen unterschiedlicher sensorischer Quellen zu verbessern. Insbesondere werden Sensorinformationen von Spracherkennern und Kameras betrachtet.

Der BAYESIAN RECEPTIONIST ist ein Dialogsystem, das die Aufgaben einer Person ausführen soll, die am Empfang eines Forschungszentrums Auskünfte gibt. Typische Anfragen an ein solches System sind etwa Wegauskünfte, Bestellen eines Taxis u.Ä. Ein solches System muss zur adäquaten Kommunikation prinzipiell (a) auf der Signalebene die natürlichsprachlichen Äußerungen des Besuchers entgegennehmen und (b) auf der semantischen Ebene, nach der Interpretation der akustischen Signale durch das Spracherkennungsmodul die Wünsche (Ziele) des Besuchers erkennen. Eine schlechte Qualität des eingehenden akustischen Signals, beispielsweise durch eine undeutliche Aussprache des Besuchers, wirkt sich somit indirekt auf die Erfolgsaussichten der semantischen Interpretation aus. Diesen Effekt versuchen die Autoren durch die Verwendung Bayes'scher Netze zum Einbringen von zusätzlichen Beobachtungen in den Interpretationsprozess (z.B. durch Analyse der Ergebnisse von Bilderkennungsverfahren, die auf sensorische Daten einer installierten Kamera angewendet werden) zu vermindern. Generell werden Einflussdiagramme genutzt, um den Dialog zu steuern. Mit ihrer Hilfe wird beispielsweise entschieden, ob im Falle eines erkannten globalen Ziels, wie z.B. der Bitte um einen Transfer, weitere Details nachgefragt werden sollen (z.B. die Abholzeit), oder ob im Fall einer schlechten Erkennung durch den Spracherkenner die Unsicherheit explizit durch Anfordern einer Wiederholung der Anfrage durch den Benutzer aufgelöst werden soll. Das Beispiel macht deutlich, dass es mit diesem Ansatz möglich ist, die unterschiedlichen Sensoren des Systems zu koordinieren, um Unsicherheiten über gewisse Sachverhalte zu vermindern und somit die Gesamtperformanz des Systems zu verbessern.

DEEPLISTENER fokussiert diesen Ansatz auf Steuerungsaufgaben, wie beispielweise das Bedienen der Präsentationssoftware während eines Vortrags durch das Geben natürlichsprachlicher Kommandos. Ein Schwerpunkt liegt bei diesem zweiten Prototyp im Verfolgen des Aufmerksamkeitsfokusses des Benutzers. Dies liefert Hinweise, ob ein potenziell vom Spracherkenner unsicher erkanntes Kommando an das Präsentationssystem adressiert war oder Teil des Vortrags ist und deshalb vom System ignoriert werden sollte.

Die Autoren deuten in ihren Veröffentlichungen die Anwendung von maschinellen Lernverfahren zur Konstruktion der dynamischen Bayes'schen Netze an, gehen allerdings an keiner Stelle detailliert auf diese Thematik ein, was den Schluss nahelegt, das es sich bei den benutzten Methoden allenfalls um Standardverfahren handeln kann. Zumindest auf der Ebene der Bewertung der Ergebnisse des Spracherkenners sind die Modelle individuell parametrisiert, um zu modellieren, inwieweit der Spracherkenner auf diese Person trainiert wurde.

2.6.8 Zukerman (2001): Argumentieren

Zukerman (2001) stellt ein Argumentationssystem für ein Krimi-Szenario vor. Der Benutzer des Systems kann mit ihm Argumente austauschen, z.B. über die Relevanz von Beweismitteln in einem Mordfall hinsichtlich eines potenziellen Täters.

Das System generiert seine Argumente als Erwiderung auf diejenigen des Benutzers auf der Basis von Diskrepanzen zwischen in zwei Bayes'schen Netzen ermittelten Wahrscheinlichkeitswerten. Je eines dieser Netze modelliert das Wissen und die (subjektiven) Ansichten über den Kriminalfall einerseits des Systems und andererseits des Benutzers. Beide Netze werden dynamisch im Verlauf des Dialogs konstruiert. Mit Hilfe des Benutzermodells und den Evidenzen, die aus den Argumenten des Benutzers extrahiert werden können, ermittelt das System eine Wahrscheinlichkeitsverteilung über die Ansichten des Benutzers. Stimmen diese nicht mit denjenigen im Weltmodell des Systems überein, generiert es eine adäquate Erwiderung zu dem betrachteten Aspekt des Falls. In diesem System kommen keinerlei maschinelle Lernverfahren zur Anwendung.

2.6.9 Bunt et al. (2001): Exploratives Lernen

Im ACE-Projekt (Bunt, Conati, Huggett & Muldner, 2001; Bunt & Conati, 2001) wird an einem intelligenten Lehr-/und Lernsystem gearbeitet, in dem die Lernenden ihr Wissen durch exploratives Lernen verbessern können, d.h., die Lernumgebung des Systems bietet Hilfsmittel an, gibt dem Lernenden aber keine feste Bearbeitungsanweisungen vor. Es bleibt ihm weitestgehend selbst überlassen, wie er unter Verwendung der angebotenen Hilfsmittel sein Wissen erweitert.

Das System nutzt ein manuell konstruiertes Bayes'sches Netz, um die Effektivität des Vorgehens des Benutzers einzuschätzen. Stellt es Defizite fest, so kann es dem Lernenden Unterstützungen und Hinweise anbieten. Im Bayes'schen Netz werden dazu auf verschiedenen Detaillierungsebenen Variablen modelliert, die das explorative Verhalten repräsentieren. Evidenzen erhält das System anhand der Interaktion des Benutzers im Rahmen der Bearbeitung von Aufgaben mit den zur Verfügung gestellten Hilfsmitteln.

2.6.10 Nicholson et al. (2001): Fallstudie

Ein weiteres Beispiel eines auf der Verwendung Bayes'scher Netze basierenden adaptiven Lehr-/Lernsystems stellen Nicholson et al. (2001) vor. Es behandelt die Domäne der Dezimalnotation, insbesondere das Erlernen der Fähigkeit, zu entscheiden, welche zweier gegebener Dezimalzahlen den größeren Wert repräsentiert. Studien der Autoren haben ergeben, dass dies für Schüler der mittleren Klassenstufen ein schwierig zu erlernendes Konzept ist. Es existieren verschiedene Kategorien typischer Fehlannahmen, wie z.B. dass die Zahl, die in ihrer Dezimalschreibweise länger als die andere ist, auch die größere der beiden darstellt. Beispielsweise schließt ein Schüler, der fälschlicherweise diese Fehlannahme verinnerlicht hat, dass $3.4342 > 3.44$.

Das System arbeitet mit unterschiedlichen Spielszenarien, in denen Teilaspekte trainiert werden. Das zugrunde liegende dynamische Bayes'sche Netz dient der möglichst optimalen Steuerung des Spielverlaufs, d.h. der Auswahl der Teillernziele, die den größten Lernerfolg versprechen. Weiterhin wird auf der Basis des Bayes'schen Netzes entschieden, ob eine aktive Hilfestellung notwendig erscheint. Die Einschätzung eines Schülers hinsichtlich potenzieller Fehlannahmen stellt eine Klassifikationsaufgabe dar. Die in den Spielszenarien beobachteten Ergebnisse dienen als Symptome einer Klassifikation der Fehler. Im Wesentlichen entspricht die Modellierung einer

Zeitscheibe des dynamischen Bayes'schen Netzes dem naiven Bayes'schen Klassifizierer (vgl. Abschnitt 2.1.4), der um Knoten erweitert wird, die im Rahmen der Spielszenarien zur Bestimmung der auszuwählenden Aufgaben benötigt werden.

Im Verlauf des Konstruktionsprozesses des intelligenten Lehr-/Lernsystems führten die Autoren eine Studie zu alternativen Methoden der Erstellung des Bayes'schen Netzes durch. Sie untersuchten und verglichen (i) die Erstellung des gesamten Netzes durch Experten, (ii) das Lernen der bedingten Wahrscheinlichkeiten der durch Experten spezifizierten Struktur mittels Standardlernverfahren und (iii) das maschinelle Lernen des kompletten Bayes'schen Netzes (Struktur und CPTs). Zusammenfassend führte jede der Einzelmethoden zu brauchbaren Netzen. Allerdings konnte insbesondere durch das Lernen der CPTs eine signifikante Verbesserung der Performanz des Netzes erzielt werden. Strukturlernverfahren unterstützten die Experten bei der semimanuellen Spezifikation einer adäquaten Struktur. Das Einbringen von (kausalem) Hintergrundwissen in Form von strukturellen Vorgaben, wie z.B. einer kausalen Ordnung der Variablen für den Lernprozess, erwies sich als nützlich.

2.6.11 Diskussion

Die Tabellen 2.4 und 2.4 fassen die im Vergleich zu den in den Übersichten von Jameson (1996) und Schäfer (1998) vorgestellten aktuelleren Arbeiten (inklusive des Abschnitt 1.1.2 beschrieben READY-Systems) zu benutzeradaptiven Systemen, die in wesentlichen Komponenten Bayes'sche Netze verwenden, zusammen. Die Auswahl an Systemen verdeutlicht die vielseitige Anwendbarkeit Bayes'scher Netze zur erfolgreichen Lösung unterschiedlicher Problemstellungen zur Unsicherheitsbehandlung. Die Flexibilität dieses probabilistischen Ansatzes wird dokumentiert durch das breit gefächerte Spektrum der Anwendungsgebiete, das von webbasierten Systemen über Office-Anwendungen bis hin zu intelligenten Lehr-/Lernsystemen reicht.

In der Mehrzahl der vorgestellten Systeme sind empirisch ermittelte Daten in irgendeiner Weise in den Konstruktionsprozess der verwendeten Bayes'schen Netze eingeflossen, sei es in Form von Erkenntnissen, die in Wizard-of-Oz-Studien gewonnen wurden, oder durch gesammelte Daten zum Interaktionsverhalten der Benutzer, die durch Anwendung von Standardlernverfahren in die Benutzermodelle eingebracht wurden. In allen Fällen handelt es sich bei den erstellten Modellen um Kombinationen aus manuell kodiertem (Hintergrund-)Wissen und im Rahmen von maschinellen Lernmethoden extrahierten empirisch basierten Informationen. Typischerweise wird vorhandenes Hintergrundwissen in Form der Struktur des Bayes'schen Netzes eingebracht und die CPTs mit Hilfe maschineller Lernverfahren automatisch gelernt. Dabei kommen in den vorgestellten Arbeiten nur unmodifizierte existierende Lernverfahren zum Einsatz, die nicht die speziellen Eigenschaften des Benutzermodellierungskontexts berücksichtigen (vgl. Abschnitt 1.3). Eine Behandlung der entsprechenden Fragestellungen wie sie in dieser Arbeit vorgestellt wird wurde bislang nicht vorgenommen.

Die bisherigen Ansätze zur Verwendung maschineller Lernverfahren Bayes'scher Netze in benutzeradaptiven Systemen beschränken sich auf einen Einsatz in der Entwurfs- und Implementationsphase. Zur Laufzeit wird zur Adaption an den Benutzer meist auf dynamische Bayes'sche Netze zurückgegriffen, die in der Lage sind, eine größere Menge an Beobachtungen zu verschieden Interaktionszeitpunkten zum aktuellen Benutzer im Rahmen des Schlussfolgerungsprozesses zu berücksichtigen. Alternative Adaptionstechniken, die beispielsweise in der Lage sind, die Komplexitätsproblematik dynamischer Bayes'scher Netze zu umgehen, spielen hier bisher kaum eine Rolle.

System	**Domäne**	**Aufgabe**	***Konstruktion der Netze***	***Eingesetzte Lernverfahren***	**Besonderheiten**
Horvitz et al. (1998): LUMIÈRE	Office-Anwendungen	Hilfe / Assistenz	*manuell*	—	Wizard-of-Oz-Studien zur empirischen Fundierung
Albrecht et al. (1998)	MUD-Spiele	Planerkennung, Vorhersage von Aktionen bzw. Positionen der Spieler	*Lernen der CPTs bei Vorgabe alternativer Strukturen*	*Maximum-Likelihood (relative Häufigkeiten)*	Hohe Komplexität der Domäne
Billsus und Pazzani (1999): NEWSDUDE	personalisierte WWW-Nachrichten	Klassifikationsaufgabe	*Lernen/Adaption der CPTs eines naiven Bayes'schen Klassifizierers*	*Maximum-Likelihood (relative Häufigkeiten/Beta-Verteilungen)*	Hybrides Benutzermodell in Form von Langzeit- und Kurzzeitbenutzermodell
Lau und Horvitz (1999)	WWW-Suchanfragen	Planerkennung bzgl. Suchzielen, Vorhersage der nächsten Aktion	*Lernen der CPTs bei Vorgabe alternativer Strukturen*	*Maximum-Likelihood (relative Häufigkeiten)*	—
Conati und Van-Lehn (1999)	Lehr-/Lernsystem	Erkennung von Wissensdefiziten anhand von Selbsterklärungen	*manuell*	—	—
Horvitz et al. (1999)	Office-Anwendungen	Situativ adäquate Benachrichtigung, Vorhersage der Verfügbarkeit	*manuell, Lernen der Struktur und CPTs*	*Maximum-Likelihood (relative Häufigkeiten)*	Kombination mit Einflussdiagrammen, Berücksichtigen des situativen Kontexts durch entsprechende Selektion der Lerndaten

***Tabelle 2.4**: Überblick benutzeradaptiver Systeme auf der Basis Bayes'scher Netze unter Berücksichtigung des Einsatzes maschineller Lernverfahren - Teil 1*

System	**Domäne**	**Aufgabe**	***Konstruktion der Netze***	***Eingesetzte Lernverfahren***	**Besonderheiten**
Paek und Horvitz (2000): BAYESIAN RECEPTIONIST, DEEPLISTENER	Infokiosk, Office-Anwendungen	Dialogführung, Erkennung natürlichsprachlicher Kommandos	*(Lernen der CPTs bei Vorgabe der Struktur)*	*(Maximum-Likelihood)*	Kombination von Daten verschiedener Sensoren
Zukerman (2001)	Argumentation	Erkennung von und adäquate Reaktion auf unterschiedliche Überzeugungen zwischen System und Benutzer	*manuell*	—	Dynamischer Aufbau des Bayes'schen Netzes
Bunt und Conati (2001)	Lehr-/Lernsystem	Erkennung von Defiziten und adäquate Unterstützung beim explorativen Lernen	*manuell*	—	—
Nicholson et al. (2001)	Lehr-/Lernsystem	Erkennung von Fehlannahmen, Generierung adäquater Lernstrategien	*manuell, Lernen der CPTs bei Vorgabe alternativer Strukturen, Lernen der Struktur*	*Maximum-Likelihood, CaMML*	Fallstudie / Vergleich zur Verwendung maschineller Lernverfahren Bayes'scher Netze im Konstruktionsprozess
Jameson et al. (2001): READY	Mobiles Dialogsystem	Erkennung und Adaption an kognitive Ressourcenbeschränkungen	*Lernen/Adaption der CPTs und der Struktur*	*Entwicklung und Verwendung von an die spezifischen Anforderung der Benutzermodellierung angepassten Lernverfahren*	Durchführung psychologisch motivierter Experimente zur Datenerhebung

Tabelle 2.5*: Überblick benutzeradaptiver Systeme auf der Basis Bayes'scher Netze unter Berücksichtigung des Einsatzes maschineller Lernverfahren - Teil 2*

Zusammenfassend kann gesagt werden, dass die Verwendung und insbesondere die Entwicklung maschineller Lernverfahren für Bayes'sche Netze im Kontext benutzeradaptiver Systeme bisher nicht im Fokus der Forschung standen, obwohl in vielen Domänen empirische Daten vorhanden sind. Werden entsprechende Verfahren eingesetzt, so handelt es sich meist um bereits existierende Lernverfahren, die in keiner Weise hinsichtlich der Verwendung in benutzeradaptiven Systemen optimiert waren. Einen Schritt zum Schließen dieser Lücke stellt die vorliegende Arbeit dar.

3 MASCHINELLES LERNEN IN BENUTZERADAPTIVEN SYSTEMEN

Ziel dieses Kapitels ist es, einen informellen, allgemeinen Überblick über die Verwendung maschineller Lernverfahren in benutzeradaptiven Systemen und den damit verbundenen Problemstellungen zu vermitteln. Die für diese Arbeit relevanten spezifischen Algorithmen werden ausführlich im folgenden Kapitel vorgestellt. Die allgemeine Formulierung des maschinellen Lernenproblems wird auf den spezifischen Kontext benutzeradaptiver Systeme übertragen, wobei eine ausführliche Diskussion der damit verknüpften Fragestellungen im Vordergrund steht. Es folgt eine Besprechung generischer Benutzermodellierungsumgebungen, die explizit den Einsatz maschineller Lernverfahren unterstützen bzw. entsprechende Verfahren zur Verfügung stellen. Weiterhin wird die in benutzeradaptiven Empfehlungssystemen wichtige Unterscheidung in kollaborative bzw. inhaltlich basierte Lernansätze der Benutzermodellierung beleuchtet. Den Abschluss des Kapitels bildet ein Überblick maschineller Lernverfahren, die bereits erfolgreich in benutzeradaptiven Systemen zur Anwendung gekommen sind.

3.1 Problemformulierung

3.1.1 Definition des allgemeinen maschinellen Lernproblems

Mitchell (1997) definiert das maschinelle Lernproblem in allgemeiner Form, derart, dass jedes System, das seine Performanz hinsichtlich einer bestimmten Aufgabe auf der Basis gesammelter Erfahrungen verbessern kann, maschinelles Lernen implementiert:

Definition 3.1 (Maschinelles Lernen) *Ein Computer-Programm* lernt, *wenn sich seine Performanz hinsichtlich der von ihm zu bearbeitenden Aufgabe A bezüglich eines Performanzmaßes Q mit der gesammelten Erfahrung E verbessert.*

Mit dieser Formulierung des maschinellen Lernproblems wird implizit zwischen zwei Komponenten eines Systems, das maschinelles Lernen realisiert, unterschieden: (a) der *Lernkomponente*, die die Erfahrungen bzw. Daten E auswertet, um das Ergebnis der im Rahmen der (b) *Performanzkomponente* behandelten Aufgabe zu verbessern (siehe Abbildung 3.1).[1] Betrachtet man als Beispiel ein lernendes Schachprogramm, so sind dies einerseits (a) das Modul, das gespielte Partien

[1] Im allgemeinen Modell eines maschinellen Lernsystems werden zwei weitere Module unterschieden, die in dieser Arbeit nicht von Interesse sind (siehe z.B. Mitchell, 1997, Kap. 1).

hinsichtlich vielversprechender Strategien analysiert, und andererseits (b) das Modul, das während des Spiels den jeweiligen besten nächsten Zug bestimmt.

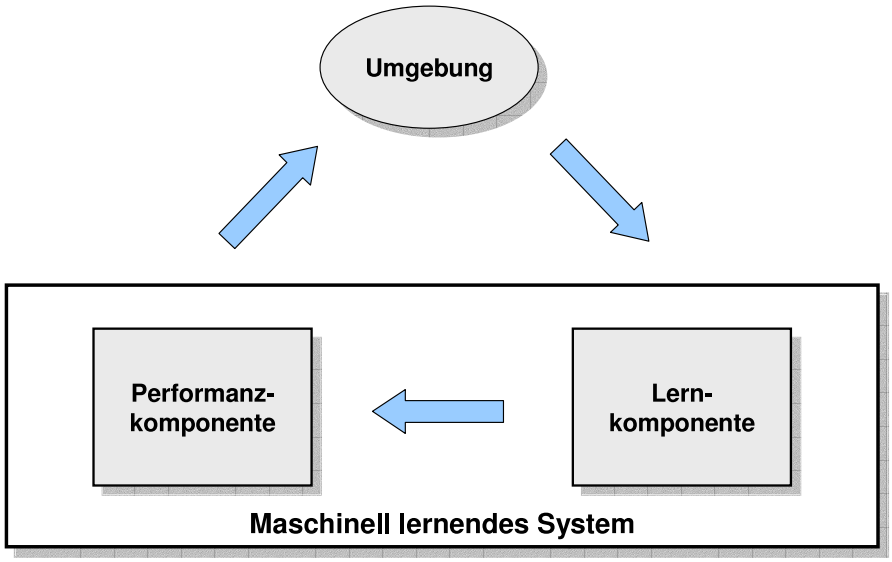


***Abbildung 3.1**: Prototypische Architektur eines maschinellen Lernsystems*
(Die Pfeile geben den Informationsfluss an.)

Zur Formulierung eines wohl-definierten maschinellen Lernproblems müssen gemäß Definition 3.1

1. die zu behandelnde Aufgabe A
2. die (Art der) verfügbaren bzw. zu sammelnden Erfahrungen (im Weiteren auch als *Trainings-* bzw. *Adaptionsdaten* bezeichnet) E, und
3. das zu verwendende Performanzmaß Q

spezifiziert werden. Neben diesen Entscheidungen zur formalen Definition des maschinellen Lernproblems ist auch die Festlegung der notwendigen Methoden zur Sammlung und Aufbereitung der Trainingsdaten in der Entwurfsphase eines solchen Systems angesiedelt. Langley (1997, 1999) weist unter Berücksichtigung von Erfahrungen aus der Praxis der Entwicklung maschineller Lernsysteme darauf hin, dass den Vorarbeiten bzw. Entwurfsentscheidungen zur Formulierung des Problems eine sehr große Bedeutung zukommt. Oftmals hat die letztendliche Wahl des verwendeten Lernalgorithmus nur einen geringen Einfluss auf die Performanz des Systems.

Das Ergebnis des Prozesses des maschinellen Lernens ist ein Modell[2] der Domäne, das oft in Kooperation mit Domänenexperten evaluiert wird. Wird eine den Anforderungen entsprechende Qualität gemäß Q erzielt, kann das erlernte Modell im System zum Einsatz kommen. Üblicherweise handelt es sich bei der Entwicklung eines maschinellen Lernsystems um einen iterativen Prozess, dessen Phasen einige Male durchlaufen werden, bis eine genügend hohe Qualität des Lernergebnisses erzielt wird. Wird ein Modell einer Domäne, das als die Realisierung einer Funktion

[2]Im Gebiet des maschinellen Lernens wird in diesem Zusammenhang anstelle von ‘Modell’ auch häufig der Begriff ‘Hypothese’ verwendet. Die Entscheidung für die Verwendung des Begriffs ‘Modell’ ist begründet in der großen Bedeutung des Konzepts des Benutzer*modells* in dieser Arbeit.

zwischen Ein- und Ausgabewerten interpretiert werden kann, anhand einer Menge von Trainingsdaten maschinell gelernt, spricht man auch von *induktivem* maschinellen Lernen. Die zugrunde liegende Annahme dieses Ansatzes ist, dass ein Modell, das anhand von Trainingsdaten gelernt wurde, in der Lage ist, noch nicht gesehene Daten adäquat zu modellieren.

Das induktive maschinelle Lernproblem kann als Suchproblem interpretiert werden: Basierend auf den verfügbaren Trainingsdaten soll ein Modell gefunden werden, das möglichst optimale Vorhersagen in der betrachteten Domäne machen kann. Dabei ist zu beachten, dass nicht unbedingt das Modell gewählt werden muss, welches die Trainingsdaten optimal repräsentieren kann, sondern ein Modell, das mit einer hohen *Generalisierungfähigkeit* ausgestattet ist, d.h., das in der Lage ist, gute Vorhersagen für ihm noch nicht bekannte Situationen zu liefern. Der Effekt, dass in vielen Situationen maschinelle Lernverfahren Modelle liefern, die zwar die Trainingsdaten sehr genau modellieren, aber lediglich eine eingeschränkte Generalisierungsfähigkeit besitzen, wird als *Übertraining* bzw. *Overfitting* bezeichnet. Meist tritt diese Problematik dann auf, wenn nur wenige Trainingsdaten zur Verfügung stehen. Dann ermittelt der Lernalgorithmus ein Modell, das zu stark auf die in diesen Daten auftretenden Eigenschaften spezialisiert ist—obwohl die beobachteten Eigenschaften bei geringen Datenmengen meist nicht repräsentativ für die gesamte Domäne sind. Typische Eigenschaften, die im Modell berücksichtigt werden sollten, lassen sich meist nur anhand einer relativ großen Menge an Trainingsdaten erkennen.

Zur Eingrenzung des Suchraumes kann oft zusätzliche Information in Form von A-priori-Wissen in die Lernprozedur eingebracht werden. In vielen Fällen kann auf diese Weise sowohl die Qualität insbesondere unter dem Aspekt der Generalisierbarkeit verbessert, als auch die Komplexität des Lernvorgangs reduziert werden. Eine Möglichkeit besteht diesbezüglich in der Vorgabe eines Startpunktes der Suche. Mit ihm spezifiziert man ein Ausgangsmodell, das das a priori vorhandene Wissen kodiert. Einer solchen Vorgehensweise liegt die Annahme zugrunde, dass das gesuchte Modell sich nur noch in wenigen Dimensionen vom Startmodell unterscheidet und sich somit in dessen „Umgebung“ im Suchraum befinden sollte.

Man unterscheidet zwischen *Batchlern*- und *Adaptionsverfahren*. Im ersten Fall wird eine (genügend) große Menge an Trainingsdaten in einem Arbeitsschritt zum Erlernen eines Modells genutzt. Da entsprechende Verfahren oft eine hohe Komplexität besitzen und/oder mit einer großen Menge an Daten umgehen müssen, werden sie in vielen Fällen vor der eigentlichen Laufzeit des Systems in einem Vorverarbeitungsschritt *offline* eingesetzt. Die (sequentielle) Aktualisierung eines vorhandenen Modells bezeichnet man als *Adaption*—bezüglich der verwendeten Daten spricht man von *Adaptionsdaten*. Adaptionsverfahren werden typischerweise *online*, d.h. zur Laufzeit des Systems eingesetzt. Die Unterscheidung ist recht subtil, da beispielsweise innerhalb eines Adaptionsverfahrens auf (effiziente) Batchlernverfahren zur Lösung von Teilaufgaben zurückgegriffen werden kann. Die beiden Arten der Algorithmen unterscheiden sich in der Art wie sie eine Menge von Daten verwenden: in einem Arbeitsschritt gemeinsam oder sequentiell. In der beschriebenen Situation werden zwar zur Lösung des vom Batchlernverfahren bearbeiteten Teilproblems alle aktuell vorhandenen Adaptionsdaten in einem Schritt genutzt, insgesamt stellt diese Menge der Adaptionsdaten aber nur eine Teilmenge der verfügbaren Daten dar. Die bereits in früheren Adaptionsschritten ins Modell eingebrachten Daten werden zum aktuellen Zeitpunkt dann nicht mehr verwendet.

Maschinelle Lernverfahren werden nach Mitchell (1997) im Wesentlichen aus drei Gründen angewendet:

- *Data-Mining / Wissensentdeckung* (engl. *Knowledge Discovery*): In vielen Fällen liegen große Datenmengen vor, die implizit Regelmäßigkeiten enthalten, welche mit automatisch operierenden Methoden des induktiven maschinellen Lernens entdeckt und anschließend ausgenutzt werden können. Beispielsweise wird in großen Firmen in dieser Weise im Rahmen einer so genannten Warenkorbanalyse das Konsumentenverhalten analysiert, um eine Optimierung der Verkaufsstrategien zu erreichen.

- *Schwierig handhabbare Domänen:* Einige Anwendungsgebiete des maschinellen Lernens zeichnen sich durch die Eigenschaft aus, dass es für einen menschlichen Experten schwierig ist, explizite Methoden zur Lösung des Problems zu entwickeln. In solchen Fällen kann oftmals auf Werkzeuge wie künstliche neuronale Netze zurückgegriffen werden, die anhand vorliegender Daten auf die Approximation nicht-linearer Funktionen trainiert werden können. Typische Beispiele für solche Anwendungen sind Gesichts- und Handschrifterkennung. Ein weiteres Beispiel sind Hidden Markov Modelle (HMM) in der Spracherkennung (siehe z.B. Wahlster, 2000).

- *Adaption:* Die Adaption eines Systems an eine sich verändernde Umgebung ist eine weitere wichtige Problemstellung, die mit Methoden aus dem Gebiet des maschinellen Lernens behandelt werden kann. Wie im Folgenden diskutiert werden wird, spielt dies eine wichtige Rolle im Szenario benutzeradaptiver Systeme.

3.1.2 Übertragung der Definition des maschinellen Lernproblems auf benutzeradaptive Systeme

Definition 3.1 des maschinellen Lernproblems legt eine direkte Übertragung auf den Kontext benutzeradaptiver Systeme in folgender Weise nahe (vgl. Langley, 1997, 1999):[3]

Definition 3.2 (Benutzeradaptives System) *Ein* benutzeradaptives System *stellt ein interaktives Softwaresystem dar, das seine Interaktionsfähigkeit mit seinen Benutzern auf der Basis von Erfahrungen mit (möglicherweise anderen) Benutzern verbessert.*

Das bedeutet hinsichtlich der Formulierung des allgemeinen maschinellen Lernproblems in Definition 3.1: Die Aufgabe A entspricht der Interaktion mit den Benutzern, das Performanzmaß Q ist ein Maß der Qualität der Interaktion und die Erfahrungen bzw. Trainingsdaten E sind die Interaktionsdaten, die entweder zum aktuellen Benutzer und/oder zu früheren Benutzern gesammelt wurden.

Man unterscheidet in diesem Kontext (Wahlster & Kobsa, 1989; Jameson, 2002)—in Analogie zur Unterscheidung zwischen Lern- und Performanzkomponente beim allgemeinen maschinellen Lernen—zwischen den beiden Komponenten zur (i) *Akquisition* des Benutzermodells (engl. *user model acquisition*) und (ii) der *Anwendung* des Benutzermodells (engl. *user model application*). Damit repräsentiert das Benutzermodell die Schnittstelle zwischen Lern- und Performanzkomponente in benutzeradaptiven Systemen (vgl. Abbildung 3.2).

In den weiteren Kapiteln der vorliegenden Arbeit liegt der Fokus auf der Aquisitionskomponente benutzeradaptiver Systeme, die typischerweise das von der Anwendungskomponente benötigte Wissen über den Benutzer aus den Interaktionsdaten extrahiert und generalisiert. Dieses

[3] Langley (1997, 1999) formuliert die Definition spezieller, indem er den Fokus auf einen einzigen Benutzer einschränkt. Im Rahmen dieser Arbeit erscheint es allerdings sinnvoller, die Formulierung hinsichtlich mehrerer Benutzer zu verallgemeinern.

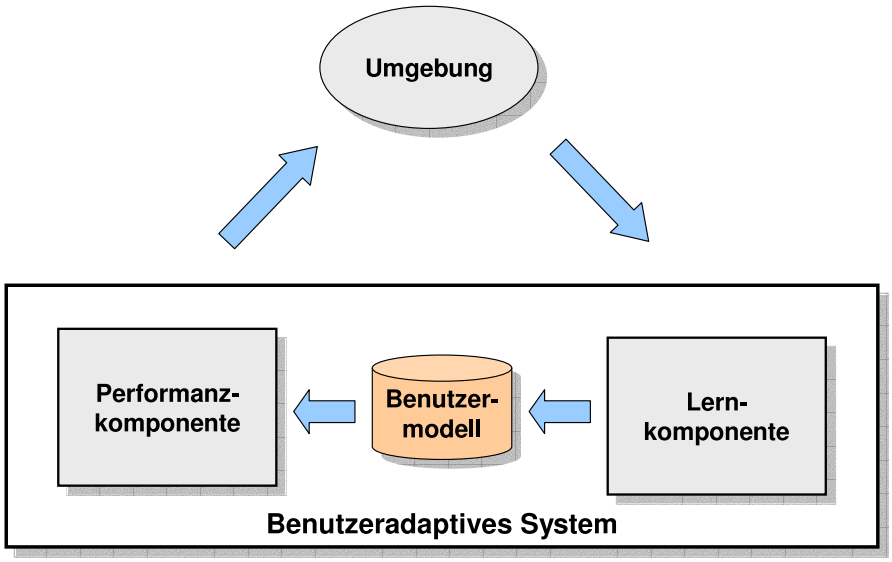


***Abbildung 3.2**: Prototypische Architektur eines benutzeradaptiven Systems aus der Sichtweise des maschinellen Lernens*

Wissen wird im Benutzermodell des benutzeradaptiven Systems repräsentiert, das den Benutzer aus unterschiedlichen Perspektiven beschreiben kann, was sich in der Aufgabenstellung der Lernverfahren widerspiegelt. So können (a) die kognitiven Prozesse, die dem Handeln des Benutzers zugrunde liegen, (b) Unterschiede zwischen Benutzern, (c) Muster des Handels bzw. Vorlieben des Benutzers und/oder (d) Besonderheiten des Benutzers modelliert werden (vgl. Webb et al., 2001). Im Speziellen werden im Rahmen dieser Arbeit die auftretenden Fragestellungen bei der Verwendung maschineller Lernverfahren zum induktiven Lernen sowie der Adaption solcher Benutzermodelle in Form Bayes'scher Netze betrachtet. Problemstellungen im Zusammenhang mit den früher im Entwurf- und Designprozess angesiedelten Phasen bei der Entwicklung eines benutzeradaptiven Systems im Sinne der Diskussion zu Definition 3.1 werden lediglich am Rande betrachtet. Darunter fallen z.B. die Entscheidungen hinsichtlich der Art der zu nutzenden Trainingsdaten sowie der Wahl des Performanzkriteriums zur Bewertung der Interaktionsfähigkeit des zu entwickelnden benutzeradaptiven Systems.

Weiterhin ist in diesem speziellen Anwendungsgebiet des maschinellen Lernens zu unterscheiden, ob entweder (i) ein *individuelles Benutzermodell* eines einzelnen Benutzers gelernt werden soll oder ob (ii) eine Gruppe von Benutzern das Ziel der Modellierung in Form eines *allgemeinen Benutzermodells* ist. In der Praxis werden auch in letzterem Fall meist mehrere Modelle, die wie im Stereotyp-Ansatz von Rich (1979, 1989) unterschiedlichen kohärenten Benutzergruppen zugeordnet sind, gelernt. Die meisten benutzeradaptiven Systeme—eine wichtige Ausnahme sind Empfehlungssysteme auf Basis des kollaborativen Filterns (siehe Abschnitt 3.3)—bedienen sich zur Zeit des ersten Modellierungsansatzes. Dies entspricht vordergründig dem übergeordneten Ziel eines benutzeradaptiven Systems, sich möglichst optimal an einen (aktuellen) individuellen Benutzer anzupassen. Nachteil einer solchen Vorgehensweise ist allerdings, dass allgemein gültige Informationen zu den Benutzern für jeden einzelnen Benutzer aufs Neue gelernt werden (müssen) und nicht aufgrund der Informationen der anderen Benutzer(modelle) an zentraler Stelle unter Vermeidung von Redundanz repräsentiert werden können.

Analog zum allgemeinen Fall lassen sich die Gründe der Verwendung maschineller Lernverfahren in benutzeradaptiven Systemen formulieren (vgl. Abschnitt 3.1.1):

- *Wissensentdeckung:* Bei der Entwicklung eines benutzeradaptiven Systems stellt die Phase der Wissensakquisition—wie in den meisten wissensbasierten Systemen—einen arbeits- und zeitintensiven Prozess dar. Meist wird in Zusammenarbeit mit Experten das benötigte Wissen zusammengetragen und in Form entsprechender Modelle dem System als Adaptionsgrundlage zur Verfügung gestellt. Zur Verringerung des Arbeitsaufwandes bieten sich an dieser Stelle Techniken des maschinellen Lernens an, um verfügbare Datensammlungen zu analysieren und die Ergebnisse einer Interpretation der Experten zu unterziehen. So können Charakteristika der Domänen erkannt sowie potenzielle Fehleinschätzungen der Experten im Rahmen des Modellierungsprozesses vermieden werden (vgl. die in Abschnitt 2.6.10 beschriebene Studie von Nicholson et al., 2001). Häufig können die erlernten Modelle auch direkt ohne Modifikation und/oder Erweiterung durch Experten im benutzeradaptiven Zielsystem eingesetzt werden. Beispiele hierzu sind die Arbeiten von Heckerman, Chickering, Meek, Rounthwaite und Kadie (2000) sowie Paliouras, Karkaletsis, Papatheodorou und Spyropoulos (1999), die Verfahren zur Visualisierung und Analyse von Zusammenhängen zwischen verschiedenen Aspekten der Domäne bzw. zur Identifikation typischer Benutzergruppen beschreiben. Teile von Kapitel 7 der vorliegenden Arbeit, die das Lernen der Struktur eines Bayes'schen-Netz-Benutzermodells behandeln, können dieser Art des maschinellen Lernens zugeordnet werden. Das Erlernen interpretierbarer Strukturen trägt u.a. zu einem tieferen Verständnis des Benutzerverhaltens bei. Solche Erkenntnisse lassen sich bei der Konstruktion der endgültigen Modelle berücksichtigen.

- *Schwierig zu modellierende Domänen:* Auch in einigen Domänen benutzeradaptiver Systeme ist es schwierig, ein (explizites) Modell durch Experten spezifizieren zu lassen. Oftmals macht eine ausgeprägte Diversität von Benutzereigenschaften, -interessen, -ziele usw. jeden Versuch der manuellen Konstruktion eines Benutzermodells zunichte. In solchen Situationen sind maschinelle Lernverfahren oft die einzige Möglichkeit benutzeradaptives Verhalten eines Systems zu realisieren. Typische Beispiele solcher Systeme sind Vertreter der Klasse der Empfehlungssysteme (vgl. Abschnitt 3.3), die z.B. Produkte wie Filme, Bücher und CDs zum Kauf vorschlagen.

- *Adaption des Benutzermodells:* Ein wichtiger Grund des Einsatzes maschineller Lernverfahren in benutzeradaptiven Systemen ist das automatische Anpassen der verwendeten Benutzermodelle im laufenden Betrieb an neue Gegebenheiten auf der Basis des Benutzerverhaltens. Diese Fähigkeit ist in vielen Anwendungssituationen entscheidend für den Erfolg des Systems. Insbesondere solche Systeme, die in dynamischen Domänen operieren, sind auf die kontinuierliche Adaption ihrer Benutzermodelle angewiesen, um potenzielle Veränderungen auch in ihrem adaptiven Verhalten adäquat abbilden zu können. In intelligenten Lehr-/Lernsystemen ist es beispielsweise gerade das Ziel, die Fähigkeiten des Benutzers, d.h. des Lernenden, hinsichtlich dessen Wissen—oder auch allgemeiner—bezüglich dessen Lernstrategien zu verbessern.

3.1.3 Problemestellungen beim maschinellen Lernen im Kontext benutzeradaptiver Systeme

Es lassen sich potenziell kritische Punkte bei der Anwendung maschineller Lernverfahren im Kontext benutzeradaptiver Systeme identifizieren, die die direkte Übertragung bzw. Anwendung entsprechender Algorithmen in einem solchen System erschweren oder sogar verhindern können. Sie werden in den folgenden Abschnitten identifiziert und diskutiert (vgl. Webb et al., 2001, für eine ähnliche Diskussion einer Teilmenge der angeführten Punkte). Diese Problemstellungen sind teilweise stark miteinander verknüpft und können mit ähnlichen Lösungsansätzen behandelt werden.

3.1.3.1 Wenige verfügbare Trainingsdaten

Ein Punkt, der einer direkten Anwendung von Standardalgorithmen aus dem Bereich des maschinellen Lernens im Kontext benutzeradaptiver Systeme im Weg steht, ist die Tatsache, dass in vielen Szenarien nur relativ wenige Trainingsdaten zur Akquisition des Benutzermodells zur Verfügung stehen. Wie schon im einleitenden Kapitel dieser Arbeit (Abschnitt 1.3) dargelegt, findet typischerweise nur eine begrenzte Anzahl an Interaktionen zwischen Benutzer und System statt. Außerdem sollen möglichst von Beginn der Interaktion an, sinnvolle Adaptionsentscheidungen getroffen werden können, um ein adaptives Verhalten des Systems zu ermöglichen. Ein Benutzer eines adaptiven Assistenzsystems wie beispielsweise NEWSDUDE (Abschnitt 2.6.3) möchte nicht gezwungen sein, erst eine Vielzahl an persönlichen Angaben gegeben oder aufwendige Bewertungen zu Beispielartikeln machen zu müssen, bevor er die Funktionalität des Systems nutzen kann. Damit ist die Verfügbarkeit von Algorithmen, die in der Lage sind, eine möglichst schnelle adäquate Adaption zu ermöglichen, von entscheidender Bedeutung für den Erfolg von benutzeradaptiven Systemen in vielen der potenziellen Einsatzszenarien.

Beim Einsatz von Batchlernverfahren zur Akquisition der Benutzermodelle muss in solchen Situationen, die sich durch geringe Mengen an Trainingsdaten auszeichnen, die Overfitting-Problematik berücksichtigt werden. Betrachtet man den Fall allgemeiner Benutzermodelle, d.h., Modelle, die anhand der Daten einer Menge von (anderen) Benutzer erlernt wird, so kann eine zu starke Spezialisierung solcher Modelle insbesondere dann auftreten, wenn die erhobenen Daten auf einer begrenzten Anzahl von Benutzern basieren oder die Ausprägungen der Nutzereigenschaften sehr heterogenen Charakter aufweisen. Es ist dann im Allgemeinen nur in sehr eingeschränkter Form möglich, ein allgemeines Modell des (typischen) Benutzerverhaltens zu erlernen. In analoger Weise kommt es meist zu Overfitting, wenn mit wenigen Daten zum betrachteten Benutzer ein individuelles Modell erlernt wird. Die geringe Menge an Daten ist nicht in der Lage, alle relevanten Benutzereigenschaften zu repräsentieren, zusätzlich können anhand der kleinen Datenmenge vom Lernverfahren fälschlicherweise Eigenschaften erkannt werden, die sich bei genauerer Betrachtung—etwa durch das Erheben von zusätzlichen Trainingsdaten—als nicht typisch für den Benutzer herausstellen. Beispielsweise sollte aus dem Kauf einer Klassik-CD nicht alleine auf ein entsprechendes Interesse des Kunden geschlossen werden, möglicherweise hat er die CD als Geschenk für einen Bekannten gekauft und ist selbst in keiner Weise an dieser Musikrichtung interessiert.

Es existieren zumindest die folgenden Möglichkeiten, das Problem einer geringen Menge verfügbarer Trainingsdaten anzugehen:

- Es bietet sich an, die Problematik durch die Adaption eines allgemeinen „Ausgangsbenutzermodells“ zu behandeln, das auf der Basis neuer Beobachtungen an den individuellen,

aktuellen Benutzer angepasst wird. Die Voraussetzung bei dieser Vorgehensweise ist allerdings, dass das den aktuellen Benutzer adäquat modellierende Modell dem Ausgangsmodell einigermaßen ähnlich ist. Damit kann schon von Beginn an—oder zumindest nach wenigen Interaktionen—eine sehr gute Adaptionsleistung erzielt werden. Hinsichtlich der Akquisition des Ausgangsmodells können neben der manuellen Konstruktion durch Experten auch Techniken des maschinellen Lernens (Batchlernverfahren) verwendet werden. Letzteres bietet sich insbesondere dann an, wenn Interaktionsdaten einer ausreichend großen Menge anderer Benutzer vorhanden sind. Mit ihrer Hilfe kann das allgemeine Modell induktiv erlernt werden. Alternativ können bereits vorhandene Benutzermodelle anstelle der Interaktionsdaten verwendet werden, um durch eine Analyse der Einzelmodelle ein „Durchschnittsmodell“ zu erstellen.

- Einige Lernverfahren können in bestimmten Situationen schon nach einer geringen Anzahl von Interaktionsschritten gute Ergebnisse erzielen. Dazu zählt beispielsweise das Verfahren der nächsten Nachbarn (engl. nearest neighbors, vgl. Abschnitt 3.4.4), das unter der Voraussetzung, dass die neuen Beobachtungen den Trainingsdaten nicht fundamental widersprechen, erfahrungsgemäß schon sehr schnell brauchbare Ergebnisse erzielen kann. Diese Technik wird z.B. im Kurzzeitbenutzermodell des NEWSDUDE-Systems verwendet, um Nachrichtenartikel zu bestimmen, die ein Thema behandeln, an dem der Benutzer aktuelles Interesse gezeigt hat. Dies kann z.B. bei aktualisierten (Folge-)Meldungen zu den Auswirkungen einer Naturkatastrophe der Fall sein kann.

Ein verwandtes Problem entsteht, wenn zu wenige Informationen zu anderen Aspekten der Domäne, die nicht direkt relevant für das Benutzermodell sind, verfügbar sind, um eine Adaptionsentscheidung treffen zu können. Diese Situation tritt z.B. in Empfehlungssystemen auf, wenn ein neues Produkt ins Angebotssortiment aufgenommen wird. Auch hier existieren zunächst keine Informationen zu den Beziehungen zwischen den Benutzerwünschen und dem neuen Produkt, auf deren Basis Empfehlungen generiert werden können.

3.1.3.2 Inter-individuelle Unterschiede zwischen Benutzern

Charakteristisch für benutzeradaptive Systeme sind die auftretenden inter-individuellen Unterschiede zwischen einzelnen Benutzern. Ohne sie gäbe es keine Existenzberechtigung für einen Großteil der benutzeradaptiven Systeme: Jeder potenzielle Benutzer könnte in gleicher Art und Weise behandelt werden. Deshalb müssen—soweit vorhanden—existierende maschinelle Lernverfahren verwendet oder speziell auf den Benutzermodellierungskontext zugeschnittene Methoden entwickelt werden, die in der Lage sind, genau diese Unterschiede zu erkennen und im Rahmen der Adaption auszunutzen.

Im einfachsten Fall kann ein allgemeines Benutzermodell durch Anwendung von Standardlernverfahren mit einer auf einer großen Anzahl von Benutzern basierenden Datensammlung erlernt werden, das bezüglich der individuellen Benutzercharakteristika parametrisiert ist. Zum Zeitpunkt der Interaktion mit einem Benutzer kann das System versuchen, die benutzerspezifischen Werte der Parameter einzuschätzen und mit ihrer Hilfe im allgemeinen Modell entsprechende Schlussfolgerungen ziehen. Beispiele solcher Parameter sind die in der Analyse in Abschnitt 2.4 verwendeten individuellen Parametervariablen. Ein weiterer Vorteil eines solchen Ansatzes ist, dass die Parameterwerte eines Benutzers gespeichert und bei der nächsten Interaktion mit eben diesem Benutzer wieder verwendet werden können. In diesem Fall kann von Beginn an eine adäquate Adaptionsleistung des Systems erreicht werden.

Oft bietet es sich an, bereits bekannte Ansätze der Repräsentation individueller Unterschiede in Benutzermodellen um eine Komponente zu ihrem Erlernen zu erweitern. So können beispielsweise Stereotypen unter Verwendung von Clusteringmethoden bestimmt werden, die automatisch ähnliche Datensätze einer Datenbank zusammenfassen und auf diese Weise zur Identifikation von Benutzergruppen beitragen (siehe z.B. Paliouras et al., 1999).

In Domänen, die sich durch geringe Unterschiede zwischen den einzelnen Benutzer(modelle)n auszeichnen, kann bevorzugt die im vorherigen Abschnitt beschriebene Vorgehensweise zur Behandlung der individuellen Unterschiede herangezogen werden: Ein allgemeines Modell (ohne Parameter) dient als Ausgangspunkt und wird sukzessive an die individuellen Eigenschaften des Benutzers auf Basis der Interaktion angepasst. Somit kann unter Ausnutzung der weitgehenden Übereinstimmung der Benutzer von Beginn an eine relativ hohe Performanz erzielt werden, einzelne individuelle Aspekte des Benutzermodells werden im Zuge des Adaptionprozesses nach und nach erkannt.

3.1.3.3 Dynamische Domänen

Eine weitere charakteristische Eigenschaft der Domänen benutzeradaptiver Systeme ist die Tatsache, dass die Eigenschaften, Interessen, Ziele usw. der Benutzer oftmals zeitabhängigen Veränderungen unterliegen. Das Problem ist im maschinellen Lernen unter dem englischen Begriff '*concept drift*' (Widmer & Kubat, 1996) bekannt. Als Beispiel hierfür kann wieder das NEWSDUDE-System dienen: Sowohl hinsichtlich des Langzeit- als auch des Kurzzeit-Benutzermodells können zeitabhängige Interessensverschiebungen auf der Benutzerseite auftreten, z.B. kann sich vor einer anstehenden Wahl trotz eines eigentlich nur moderaten Interesses an Politik ein Interessensschwerpunkt auf Nachrichten aus dieser Kategorie bilden, der nach dem Wahltermin wieder rapide an Gewicht verliert, wohingegen gleichzeitig eine zeitnah stattfindende Fußballweltmeisterschaft in den Fokus des Interesses rückt.

Ein verwandtes Problem ist das Erkennen neuer Interessen eines Benutzers. Man muss hier unterscheiden, ob es sich (a) um eine Eigenschaft handelt, die der Benutzer schon besessen hat, die aber bislang noch nicht vom System beobachtet wurde, oder (b) die es zwar beobachtet, aber bereits wieder „vergessen" hat, oder (c) ob es tatsächlich eine sich neu entwickelte Eigenschaft des Benutzers ist.

Es existieren verschiedene Ansätze, diese Problemstellungen zu behandeln, u.a.:

- Ältere Trainingsdaten bekommen im Rahmen der Lernprozedur ein geringeres (relatives) Gewicht zugewiesen als aktuellere Daten. Der kritische Punkt eines solchen Ansatzes ist die Bestimmung der Gewichtung. Andererseits ist die Annahme, dass weiter zurückliegende Daten von geringerer Bedeutung sind, nicht immer korrekt. Oft treten—wie angesprochen—in unregelmäßigen Abständen immer wieder ähnliche Situationen auf, die ähnliche Verhaltensweisen des benutzeradaptiven Systems erfordern. Hat ein Kunde beispielsweise über einen längeren Zeitraum hinweg keine Jazz-CD mehr gekauft, bedeutet dies nicht unbedingt, dass er sein Interesse an diesem Genre verloren hat.

- Eine ähnliche Verfahrensweise stellen Zeitfenster-Techniken dar, die nur Daten innerhalb eines gewissen Zeitraumes zur Konstruktion bzw. Pflege des Benutzermodells in Betracht ziehen. In diesem Fall entstehen ähnliche Probleme wie beim vorhergehenden Lösungsansatz. Wiederkehrende Eigenschaften bzw. Interessen des Benutzers können durch einen

Rückgriff auf zurückliegende Zeitfenster behandelt werden, in denen bereits die entsprechenden Aspekte des Benutzerverhaltens beobachtet werden konnten (siehe z.B. Koychev, 2001).

- Wie in NEWSDUDE können hybride Methoden angewendet werden, in denen verschiedenen (Teil-)Benutzermodelle zu Modellierung unterschiedlicher Betrachtungszeiträume unterschieden werden. Die entscheidende und schwierige Frage eines solchen Ansatzes besteht in der adäquaten Auswahl des Teil-Benutzermodells in einer bestimmten Situation. Wann soll welches Modell im Inferenzprozess herangezogen werden?

3.1.3.4 Komplexität der Lernverfahren / Effizienz zur Laufzeit

Gerade in webbasierten oder mobilen Szenarien ist die Effizienz der verwendeten Verfahren ein entscheidendes Kriterium der Einsetzbarkeit eines benutzeradaptiven Systems. Webshops, wie der des Online-Buchhändlers AMAZON, müssen in der Lage sein, täglich Millionen von Anfragen zu beantworten. Mobile Systeme zeichnen sich oft durch begrenzte Ressourcen wie geringere Rechenleistung und Speicherkapazität aus. In solchen Situationen müssen die dem Adaptionsmechanismus zugrunde liegenden Verfahren extrem kurze Antwortzeiten bzw. einen geringen Bedarf an Rechenaufwand garantieren, um einen sinnvollen Einsatz zu gewährleisten.

Diese Problematik führt dazu, dass in kommerziellen Anwendungen oft nur ein Teil der potenziell möglichen und wünschenswerten Adaptionsfähigkeiten realisiert werden kann. Vielfach wird in solchen Situationen bislang komplett auf automatische Anpassung durch das System verzichtet und dem Benutzer die Möglichkeit geboten, sich bestimmte Aspekte des Systemverhaltens manuell zu konfigurieren. Man spricht dann von *adaptierbaren* (im Gegensatz zu adaptiven) Systemen. Beispiele solcher Systeme finden sich in vielen Portalseiten wie z.B. bei YAHOO!,[4] wo sich der Benutzer eine seinen Wünschen entsprechende Startseite zusammenstellen kann, indem er aus einem Angebot an verschiedenen Diensten wie Börsennachrichten, Sportnachrichten, Wetter usw. auswählt. Eine weitergehende automatische Anpassung findet nicht statt.

Um auch in problematischen Domänen maschinelle Lernverfahren einsetzen zu können, muss versucht werden, möglichst viele Teilkomponenten dieser Verfahren zu identifizieren, die in Vorverarbeitungsschritte ausgelagert werden können, und z.B. (semi-)offline zwischen zwei Benutzersitzungen (möglicherweise auf zusätzlicher Hardware) bearbeitet werden können. Die Adaptionskomponenten, die im Laufzeitbetrieb des Systems verbleiben, müssen die geforderten Antwortzeiten garantieren. Man hat hier meist zwischen der Genauigkeit der eingesetzten Verfahren und den geforderten Antwortzeiten des Systems abzuwägen. In vielen Fällen stellt sich diese Frage allerdings dennoch nicht, da die Komplexität vieler Lern- bzw. Adaptionsverfahren deutlich zu hoch ist, um überhaupt für einen Einsatz zur Laufzeit des System in Erwägung gezogen zu werden.

3.1.3.5 Interpretierbarkeit der erlernten Benutzermodelle

In benutzeradaptiven Systemen spielt die *Interpretierbarkeit* der Benutzermodelle eine wichtige Rolle. Diese Eigenschaft ist eng verbundenen mit den beiden Begriffen der *Vorhersagbarkeit* und der *Transparenz* des Systemverhaltens (vgl. Wahlster, 1981 und Jameson, 2002). Herlocker et al.

[4] www.yahoo.com

(2000) führen die folgenden Gründe an, die für die Verwendung interpretierbarer Benutzermodelle als Grundlage vorhersagbarer, transparenter Systeme sprechen:

- *Begründung:* Auf der Basis interpretierbarer Modelle, kann der Benutzer entscheiden, wieviel Vertrauen er in die Adaptionsentscheidungen des Systems setzt.
- *Einbeziehung des Benutzers:* Durch eine Einbeziehung des Benutzers in den Schlussfolgerungsprozess kann dieser sein Wissen in den Entscheidungsprozess einbringen. Dies ist nur dann möglich, wenn der Benutzer sein Wissen in den Kontext des interpretierbaren Modells einordnen kann.
- *Verständnis:* Versteht der Benutzer den Schlussfolgerungsprozess des Systems, so kann er dessen Stärken und Schwächen erkennen.
- *Akzeptanz:* Aufbauend auf den bisher angeführten Gründen, erhöht sich die Akzeptanz eines benutzeradaptiven Systems mit einem interpretierbaren Benutzermodell, da seine Grenzen und sein Potential erkennbar sind und die Entscheidungen eines solchen Systems begründet werden können (vgl. Wahlster, 1981; Teach & Shortliffe, 1984; Cook & Kay, 1994; Herlocker et al., 2000).

Diese Gründe spielen in ähnlicher Form nicht nur bei der Anwendung des Benutzermodells sondern auch beim Entwurf und der Konstruktion eines benutzeradaptiven Systems eine wichtige Rolle. Die Verwendung interpretierbarer Benutzermodelle versetzt die Systementwickler in die Lage, Fehler in den Modellen zu lokalisieren und diese gegebenenfalls durch Modifikation der Modelle zu beheben.

Da in den meisten Anwendungsszenarien maschineller Lernverfahren die (prediktive) Genauigkeit der Modelle im Vordergrund steht, müssen im Sonderfall benutzeradaptiver Systeme neue Verfahren entwickelt bzw. existierende angepasst werden, um die Interpretierbarkeit der erlernten Benutzermodelle zu gewährleisten oder den Grad der bereits vorhandenen Interpretierbarkeit zu verbessern. Erst dann können aufbauend auf solchen Verfahren Erklärungskomponenten—wie in Abschnitt 2.1.7 am Beispiel Bayes'scher Netze beschrieben—realisiert werden, welche benutzeradaptive Systeme in die Lage versetzen, die angeführten potenziellen Vorteile auszunutzen. Es existieren deutliche Unterschiede zwischen verschiedenen maschinellen Lernverfahren hinsichtlich der prinzipiellen Interpretierbarkeit der mit ihrer Hilfe erlernten Modelle. Ein Verfahren, dass sich relativ gut als Basis einer Erklärungskomponente eignet, ist z.B. die Anwendung regelbasierter Methoden. Andere Methoden, die das erlernte Wissen implizit kodieren, wie etwa neuronale Netze, eignen sich ohne zusätzliche Erweiterung nicht zur Repräsentation interpretierbarer Benutzermodelle. Solche auf wissensbasierten Techniken aufbauende Methoden, die die Interpretierbarkeit des Inferenzprozesses künstlicher neuronaler Netze erhöhen, werden z.B. von Cloete und Zurada (1999) beschrieben. Weitere maschinelle Lernverfahren werden in Abschnitt 3.4 hinsichtlich dieser Eigenschaft untersucht.

3.1.3.6 Eigenschaften der Trainingsdaten

Im Wesentlichen können zwei Arten von Trainingsdaten für das maschinelle Lernen von Benutzermodellen unterschieden werden (vgl. Kobsa et al., 2001; Jameson, 2002): (a) solche *expliziten* Charakters, die beispielsweise vom Benutzer selbst z.B. durch Ausfüllen eines Fragebogens oder durch Bewerten von (Test-)Objekten dem System zur Verfügung gestellt werden, und (b)

Daten, *impliziten* Charakters, die indirekt, anhand der Interaktionsdaten, unter Anwendung spezieller Verfahren behandelt werden (müssen). Allgemein spricht man in diesem Zusammenhang von (a) *überwachtem* (engl. *supervised*) bzw. (b) *unüberwachtem* (engl. *unsupervised*) maschinellem Lernen. In letztere Kategorie fallen beispielsweise Informationen, die durch eine Analyse der vom Benutzer gekauften Waren oder seines Navigationsverhaltens auf den Web-Seiten des Online-Shops indirekt bestimmt werden können, wie etwa seine Interessen. Kobsa et al. (2001) und Jameson (2002) stellen detailliertere Kategorisierungen der in benutzeradaptiven Systemen im Rahmen des Adaptionsprozesses relevanten Datenausprägungen vor.

Insbesondere der zweite Fall ist typisch für die Situation in vielen benutzeradaptiven Systemen, denn oft soll eine für den Benutzer aufwendige Befragungsphase vor der eigentlichen Interaktion vermieden werden, um die Hemmschwelle für die potenziellen Benutzer so niedrig wie möglich zu halten. Gleichzeitig sind Rückmeldungen über den Erfolg der Adaption in vielen Fällen nicht direkt möglich. Implizite Daten sind naheliegenderweise oft stark mit Unsicherheit bzw. Rauschen behaftet, was von den eingesetzten maschinellen Lernmethoden bei der Modellakquisition berücksichtigt werden sollte.

Ein eng verwandtes Problem in der Benutzermodellierung sind *fehlende Daten* (engl. *missing data*), d.h., unvollständige Datensätze, in denen Werte zu einzelnen Variablen nicht aufgezeichnet wurden. Dieses Fehlen kann unterschiedliche Gründe haben: Oftmals ist es technisch nicht möglich die Daten zu erheben, beispielsweise bedingt durch das Fehlen entsprechender Sensoren, in anderen Fällen ist es prinzipiell unmöglich die Werte von Variablen zu beobachten, wie dies z.B. bei solchen Variablen wie KOGNITIVE BELASTUNG des Netzes in Abschnitt 2.2.1.4 der Fall ist. Man spricht dann von so genannten (für den Lernprozess) *verborgenen Variablen* (engl. *hidden variables*).

Oft kann diese Problematik (zum Teil) dadurch bearbeitet werden, indem während der Konstruktionsphase des Systems Benutzerstudien zum Aufbau einer Datenbasis durchgeführt werden. Solche Studien bieten den Vorteil, dass in ihrem Verlauf viele Parameter bzw. Variablen der Kontrolle der Versuchsleiter unterstehen und somit verlässliche Daten erhoben werden können. Benutzermodelle, die auf der Grundlage solcher Daten gelernt wurden, spiegeln allerdings möglicherweise nicht exakt die Anwendungssituation wider, können aber in vielen Fällen als Ausgangspunkt des Adaptionsvorgangs dienen. Weiterhin können fehlende Daten durch zusätzliches Einbringen von Hintergrundwissen über die entsprechenden Variablen—soweit von den verwendeten Verfahren unterstützt—(teilweise) ausgeglichen werden. Im Beispiel der Variable KOGNITIVE BELASTUNG bedeutet dies, dass man Informationen zum Zusammenspiel dieser Variable mit anderen im Netz ausnutzen kann, um den Lernprozess positiv zu beeinflussen. Man weiß etwa, dass eine erhöhte kognitive Belastung im Allgemeinen zum vermehrten Auftreten von Fehlern führt.

3.1.3.7 Integration von a priori verfügbarem Wissen

Das Einbringen von verfügbarem A-priori-Wissen ist eine nicht nur zur Kompensation von fehlenden Daten häufig eingesetzte Möglichkeit zur Verbesserung der Ergebnisse eines maschinellen Lernprozesses (siehe z.B. Mitchell, 1997, Kap. 12). Auf diese Weise ist es in vielen Fällen möglich, die Menge an Trainingsdaten—und damit die Komplexität des Lernprozesses—, die zum Erlernen eines adäquaten Modells benötigt wird, deutlich zu reduzieren. Solche Ansätze bieten sich somit zur Behandlung der Problematik weniger verfügbarer Trainingsdaten in benutzeradaptiven Systemen an (vgl. Abschnitt 3.1.3.1), um möglichst rasch sinnvolle Adaptionsentscheidungen treffen zu können. Zusätzlich erhöht das eingebrachte Vorwissen in den meisten Fällen die Interpretierbarkeit der Resultate im Sinne der Diskussion in Abschnitt 3.1.3.5.

Jameson (2002) unterscheidet in diesem Zusammenhang *daten-basierte* sowie *theorie-basierte* Ansätze zur Konstruktion der Benutzermodelle. Die Integration von A-priori-Wissen und maschinellem Lernen stellt in dieser Kategorisierung eine Kombination dieser beiden Alternativen dar.

Insbesondere im Fall der Interpretation des maschinellen Lernproblems von Benutzermodellen als Suchproblem erscheinen folgende Ansätze des Einbringens von Hintergrundwissen vielversprechend (vgl. Mitchell, 1997):

- *Konstruktion eines Startmodells für die Suche:* Im Gegensatz zu Abschnitt 3.1.3.1 dient hier ein (von einem Domänenexperten) manuell konstruiertes „Ausgangsmodell“ als Startpunkt der Suchprozedur (im Gegensatz zur Adaptionsprozedur). Die in den meisten Fällen gültige, zugrunde liegende Annahme ist hierbei wie in Abschnitt 3.1.1 beschrieben, dass dieses Ausgangsmodell eine hinreichend gute Modellierung darstellt, d.h., dass das „echte“ Benutzermodell sich nur in beschränktem Umfang von ihm unterscheidet. Unter diesen Voraussetzungen liegt das Ergebnis der Lernprozedur in einer Nachbarregion im Suchraum und kann recht schnell vom Lernalgorithmus „gefunden“ werden.

- *Modifikation des Suchkriteriums:* Durch eine entsprechende Berücksichtigung bekannter Informationen im in der Formulierung des wohl-definierten maschinellen Lernproblems verwendeten Performanzmaß kann die Generalisierungsfähigkeit des erlernten Benutzermodell oft gesteigert werden. Meist wird dazu ein zusätzlicher Term in die Bewertungsfunktion eingebracht, der beim Vorliegen einer zu großen Diskrepanz zwischen dem potenziellen Modell und dem verfügbaren Hintergrundwissen zu einer niedrigeren Gesamtbewertung führt. Anschaulich gesprochen wird damit der Suchprozess durch den Suchraum „geführt“ indem potenziell schlechte Bereiche gemieden werden.

- *Modifikation der potenziellen Suchschritte:* Hintergrundwissen kann genutzt werden, um den Mechanismus der Suchprozedur direkt zu beeinflussen. Im Extremfall können beispielsweise potenzielle Suchschritte explizit verboten bzw. erlaubt werden. Ebenso kann eine Sequenz aufeinander folgender Suchschritte in einem einzigen Schritt aggregiert werden, um die Effizienz des Verfahrens zu erhöhen.

3.1.3.8 Evaluation

Ein wichtiges—wenn auch bis vor kurzer Zeit von vielen Forschern teilweise vernachlässigtes—Thema ist die Evaluation benutzeradaptiver Systeme (siehe Chin, 2001; Weibelzahl, 2001). Sie ermöglicht (a) das Nachprüfen, ob die Eigenschaft der Benutzeradaptivität eines Systems tatsächlich einen Mehrwert darstellt und (b) die Detektion von Systemdefiziten und -fehlern.

Einordnung in das Ebenenmodell Konzeptuell bieten sich *Ebenenmodelle* (engl. *layered evaluation*) zur Durchführung der Evaluation benutzeradaptiver Systeme an (siehe z.B. Weibelzahl & Weber, 2002). Dieser Ansatz besteht in der sukzessiven Evaluation der verschiedenen, aufeinander aufbauenden, abstrakten Datenverarbeitungsprozesse im untersuchten benutzeradaptiven System, um den Erfolg der Adaptivität zu bewerten. Die Ebenen werden einzeln betrachtet, wobei eine erfolgreiche Evaluation einer Ebene die Voraussetzung einer erfolgreichen Evaluation der übergeordneten Ebene und schließlich des Gesamtsystems ist. Weibelzahl und Weber (2002) schlagen ein Modell bestehend aus vier Ebenen vor (Abbildung 3.3): (i) *Akquisition der Interaktionsdaten*,

(ii) die auf diesen Daten basierende *Konstruktion bzw. Akquisition des Benutzermodells*, (iii) die Bestimmung der *Adaptionsentscheidungen* und (iv) die *Realisierung der Adaption*. Auf der ersten Ebene wird u.a. die Qualität der Datenerhebung bzw. die Güte der Daten bewertet. Nur Daten, die in entsprechender Weise geeignet sind, können als Grundlage der Akquisition brauchbarer Benutzermodelle dienen. Die Performanz der Benutzermodelle wird auf Ebene (ii) betrachtet. Dies kann u.a. anhand „echter" Daten der Domäne geschehen (wie z.B. bei Müller et al., 2001) oder aber anhand hypothetischer, zu erwartender typischer Fälle (siehe z.B. Berthold & Jameson, 1999). Auf Ebene (iii) wird der Mechanismus zur Ermittlung der Adaptionsentscheidungen evaluiert, der auf den im Benutzermodell repräsentierten Eigenschaften der Benutzer aufbaut. Die oberste Ebene behandelt den Erfolg der Interaktion zwischen System und Benutzer. Ein typisches Kriterium ist hier die (subjektive) Zufriedenheit des Benutzers mit den adaptiven Eigenschaften des Systems. Weitere Beispiele für Evaluationen auf den einzelnen Ebenen geben Weibelzahl und Weber (2002).

Da sich die vorliegende Arbeit mit dem induktiven Lernen bzw. der Adaption der Benutzermodelle beschäftigt, sind die im weiteren Verlauf der Arbeit vorgestellten Evaluationen maschinell erlernter Benutzermodelle auf der zweiten Ebene dieses Modells einzuordnen.

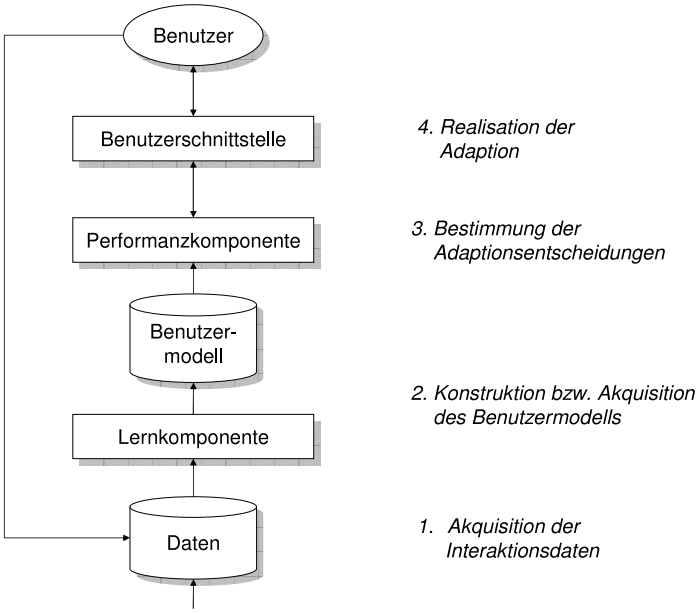


***Abbildung 3.3**: Ebenenmodell der Evaluation benutzeradaptiver Systeme (aus der Sichtweise des maschinellen Lernens)*
(Die Pfeile geben an, welche Komponenten welchen anderen Komponenten Informationen bzw. Daten zur Verfügung stellen.)

Evaluationskriterien Evaluationen, die auf dieser zweiten Ebene durchgeführt werden, bedienen sich Performanzkriterien aus dem Gebiet des maschinellen Lernens oder—je nach Anwendungsszenario—dem Information Retrieval (vgl. Zukerman & Albrecht, 2001):

- *Recall / Precision:* Diese beiden Performanzmetriken werden zur Evaluation der Leistungsfähigkeit von Empfehlungssystemen eingesetzt. *Recall* gibt den Prozentsatz der korrekt empfohlenen Objekte bezogen auf alle verfügbaren Objekte an, die korrekterweise emp-

fohlen werden müssten. *Precision* misst den Anteil korrekter Ergebnisse in der gesamten Resultatsmenge. Im optimalen Fall sollte ein System sowohl einen hohen Recall- als auch Precision-Wert erzielen, in der Praxis muss meist im Rahmen eines Tradeoffs zwischen diesen beiden Eigenschaften eines Systems abgewogen werden.

- *Vorhersagegenauigkeit / -wahrscheinlichkeit:* Diese beiden Klassen von Metriken dienen der Evaluation von Systemen bzw. Modellen, die Vorhersagen über das Benutzerverhalten, die Interessen, usw. liefern. Typischerweise entspricht die *Vorhersagegenauigkeit* dem Verhältnis der vom Modell mit den höchsten Wahrscheinlichkeiten vorhergesagten Hypothesen und den tatsächlich zutreffenden. Alternativen sind diesbezüglich bekannte Fehlermaße, wie der quadratische oder logarithmische Verlust, wie sie auch in der vorliegenden Arbeit zur Anwendung kommen (vgl. Abschnitte 6.4.1 und 7.3.5.2). Die *Vorhersagewahrscheinlichkeit* gibt die durchschnittliche vom Modell bestimmte Wahrscheinlichkeit des tatsächlich eintretenden Ereignisses an.

Kreuzvalidierung Maschinell erlernte Modelle werden unter Verwendung von *Testdaten* evaluiert. Dabei handelt es sich um Datensätze, die nicht innerhalb des Lernprozesses als Trainingsdaten zur Verfügung gestanden haben. Damit soll die Generalisierbarkeit des Modells hinsichtlich neuer, noch nicht gesehener Daten beurteilt werden. In der Praxis wird meist die komplette Menge an verfügbaren Daten (zufällig) in Trainings- und Testdaten partitioniert, beispielsweise in einem Verhältnis von 80 zu 20.

Problematisch ist dies, wenn nur eine geringe Datenmenge vorliegt—wie es im Kontext benutzeradaptiver Systeme häufig der Fall ist (vgl. Abschnitt 3.1.3.1). Einerseits sollten möglichst viele Daten dem Lernverfahren zur Verfügung stehen, d.h., die Testmenge wird aus sehr wenigen Datensätzen bestehen. Andererseits können solche sehr kleinen Datensätze die Gesamtheit der Eigenschaften der analysierten Domäne nicht erfassen. In solchen Situationen wird beim maschinellen Lernen die Methode der (k-fachen) *Kreuzvalidierung* angewendet: Dabei wird der gesamte Datenbestand in k Datenmengen partitioniert. Die Vereinigung von $k-1$ dieser Mengen wird als Trainingsmenge des Lernvorgangs benutzt. Danach wird das erlernte Modell unter Verwendung der verbleibenden k-ten Datenmenge als Testmenge evaluiert. Dies wird für alle k Kombinationen von Trainings- und Testmengen durchgeführt. Das Gesamtresultat der Evaluierung ergibt sich als Durchschnitt der Resultate der k Teilevaluationen.

Berücksichtigt man die Tatsache, dass Datensammlungen im Umfeld von benutzeradaptiven Systemen in den meisten Fällen Daten verschiedener Personen beinhalten, so bietet sich eine Partitionierung der Daten nach Benutzern wie in der in Abschnitt 2.4 vorgestellten Analyse als eine hinsichtlich benutzeradaptiven Systemen adäquate Alternative der Evaluation an. D.h., die Daten von $k-1$ Benutzern werden zum Erlernen eines allgemeinen Benutzermodells verwendet, welches anschließend hinsichtlich der Daten des verbliebenen k-ten Benutzers bewertet wird. Es handelt sich hierbei um eine *Leave-one-out-Kreuzvalidierung* auf der Ebene der Benutzer. Schließlich ist es gerade das Ziel eines benutzeradaptiven Systems, eine möglichst optimale Performanz bezüglich des jeweiligen (neuen) individuellen Benutzers zu erzielen. In anderen Fällen, in denen beispielsweise die langfristige Performanz eines Systems, das sich über einen großen Zeitraum an einen einzigen Benutzer anpasst, untersucht werden soll, müssen der abweichenden Problemstellung entsprechende Methoden angewendet werden. In einem solchen Fall könnten etwa die erhobenen Interaktionsdaten des individuellen Benutzers bis zu einem bestimmten Zeitpunkt als Trainingsdaten und die restlichen als Testdaten eingesetzt werden. Aber auch in diesem Fall soll-

ten die Analysen wie üblich mit einer größeren Anzahl an Benutzern durchgeführt werden, um Verzerrungen der Ergebnisse durch Sonderfälle zu vermeiden.

Nach der Diskussion der zu behandelnden Problemstellungen beim maschinellen Lernen in benutzeradaptiven Systemen wird in den folgenden Abschnitten ein Überblick über Verfahren und Werkzeuge des Einsatzes maschineller Lerntechniken im Zusammenhang mit benutzeradaptiven Systemen gegeben.

3.2 Integrative generische Ansätze zum maschinellen Lernen in benutzeradaptiven Systemen

Zur Verringerung des Entwicklungsaufwandes benutzeradaptiver Systeme wurden in Analogie zu Expertensystemshells (siehe z.B. Beierle & Kern-Isberner, 2000) so genannte *Benutzermodellierungsshells* entwickelt, die dem Entwickler eine Vielzahl generischer Werkzeuge zur Repräsentation, Pflege und Anwendung von Benutzermodellen zur Verfügung stellen.[5] Üblicherweise bieten solche Umgebungen Implementationen von Standardverfahren der Benutzermodellierung wie z.B. den Stereotypen-Ansatz oder logik-basierte Schlussfolgerungsmechanismen. Mit der Verwendung von Benutzermodellierungsshells soll hauptsächlich—neben der klaren Trennung von eigentlicher Systemfunktionalität und den Komponenten zur Generierung der Benutzeradaptivität—wie bereits angedeutet einerseits der Konstruktionsprozess benutzeradaptiver Systeme erleichtert werden und andererseits im Rahmen von Client-Server-Architekturen der Aufbau und das Ausnutzen gemeinsamer Benutzermodelle durch verschiedene Applikationen ermöglicht werden.

Im Folgenden werden zwei Forschungsprojekte dieser Art vorgestellt, die maschinelle Lernverfahren zur Akquisition und Adaption der Benutzermodelle anbieten. Im Vordergrund steht dabei die Integration der (verschiedenen) Lernverfahren in das Gesamtkonzept der jeweiligen generischen Umgebung. Verweise auf Ansätze, die im Wesentlichen auf wissensbasierten Techniken fußen, finden sich bei Kobsa (2001a).

3.2.1 Orwant (1993 – 1995): DOPPELGÄNGER

DOPPELGÄNGER (Orwant, 1995) ist ein generisches Benutzermodellierungssystem, dessen Hauptaugenmerk auf der verteilten Akquisition und Verwendung der Benutzermodelle liegt. DOPPELGÄNGER sammelt Information zu Benutzern anhand von *Datenströmen* verschiedener Sensoren. Dabei kann es sich sowohl um Software- als auch um Hardware-Sensoren handeln. Diese Datenströme werden vom System unter Verwendung maschineller Lernverfahren analysiert und zum Aufbau bzw. zur Pflege expliziter, von den Lernverfahren getrennt verwalteter Benutzermodelle verwendet. Unterschiedliche Datenströme müssen mit unterschiedlichen, ihren Eigenschaften entsprechenden, Methoden behandelt werden. Durch die Abtrennung der Repräsentation der Benutzermodelle von den spezifischen Lernverfahren wird die Anwendung der erworbenen Modelle in einer Vielzahl unterschiedlicher Anwendungen ermöglicht.

Orwant (1995) beschreibt folgende Lernverfahren, die für die Behandlung prototypischer Aufgaben in benutzeradaptiven Systemen in DOPPELGÄNGER integriert wurden:

[5] Kobsa (2001a) gibt einen ausführlichen Überblick über die Entwicklung und den aktuellen Stand der Forschung zu solchen generischen Benutzermodellierungssystemen.

- *Bayes'sches Lernen mit Beta-Verteilungen:* Um einfache Annahmen über die Benutzerinteressen modellieren zu können, stellt das DOPPELGÄNGER-System dem Entwickler Beta-Verteilungen zur Verfügung. Mit ihrer Hilfe können Wahrscheinlichkeiten für Benutzerinteressen inklusive einem Maß für die Zuverlässlichkeit dieser Einschätzung des Systems repräsentiert werden. Diese Technik wird beispielsweise auch in NEWSDUDE verwendet, um den naiven Bayes'schen Klassifizierer anhand der Benutzerrückmeldungen zu adaptieren. Eine ausführliche Beschreibung des Bayes'schen Lernens mit Beta-Verteilungen wird in Kapitel 4 gegeben.

- *Lineare Vorhersage:* In benutzeradaptiven Systeme ist es häufig notwendig, das Eintreten wiederkehrender Ereignisse vorherzusagen. Beispielsweise kann es sinnvoll sein, den Zeitpunkt zu antizipieren, zu dem der Benutzer sich eine vom System zusammengestellte Übersicht über die aktuellen Nachrichten anschauen, und wie lange er sich dafür voraussichtlich Zeit nehmen wird. Zur Bearbeitung solcher Aufgaben bietet DOPPELGÄNGER das Werkzeug der linearen Vorhersage. Anhand der Informationen des zugeordneten Datenstroms wird ein Modell gelernt, das zyklische Muster erkennt und entsprechende Ereignisse vorhersagen kann.

- *Markov'sche Modelle:* Zur Modellierung temporaler Aspekte des Benutzerverhaltens nutzt DOPPELGÄNGER Markov'sche Modelle. Sie eignen sich gut zur Vorhersage eines Ereignisses, das vom Eintreten anderer Ereignisse in der Vergangenheit abhängt. Dynamische Bayes'sche Netze sind ein Spezialfall der Klasse der Markov'schen Modelle. Orwant (1995) setzt diesen Ansatz u.a. ein, um den Aufenthaltsort des Benutzers vorherzusagen.

- *Unüberwachtes Clustering:* Zur automatischen Konstruktion von Benutzergruppen bestehend aus „ähnlichen" Benutzern wird unüberwachtes Clustering verwendet. Die so maschinell erlernten Gruppen dienen als Informationsquelle im Adaptionsprozess, wenn Informationen zum individuellen Benutzer fehlen. Dann kann im Sinne von Default-Annahmen wie im Stereotyp-Ansatz auf die entsprechenden Werte aus dem Gruppenmodell zurückgegriffen werden.

Die zentrale Aussage des DOPPELGÄNGER-Ansatzes zur Integration unterschiedlicher maschineller Lernverfahren in einem generischen Benutzermodellierungssystem lässt sich folgendermaßen zusammenfassen: Die Gesamtheit der Interaktion wird in eigenständige Interaktionsdatenströme aufgespalten, die individuell im Rahmen adäquater Lernalgorithmen bearbeitet werden. Das Benutzermodell stellt somit ein hybrides Konglomerat unterschiedlicher, maschinell erlernter Teilmodelle dar. Es muss folglich eine Kontrollinstanz existieren, die über die Zuordnung von Datenstrom und Verfahren entscheidet. Idealerweise sollte diese Systemkomponente autonom arbeiten—ohne die Notwendigkeit des Eingriffs des Systementwicklers.

3.2.2 Pohl et al. (1997 – 1999): LABOUR

In der LABOUR[6]-Architektur (Pohl, Schwab & Koychev, 1999; Pohl & Nick, 1999) steht die Integration traditioneller, wissensbasierter Methoden und Verfahren des maschinellen Lernens im Vordergrund. Sie baut auf den im DOPPELGÄNGER-System realisierten Ideen auf und führt den

[6] LABOUR ist das Akronym für '**L**earning **ABO**ut the **U**ser'

hybriden Ansatz konsequent hinsichtlich einer Erweiterung um Methoden der traditionellen Wissensakquisition fort.

In der LABOUR-Architektur werden Beobachtungen über das Benutzerverhalten entweder von *Akquisitions*-Komponenten oder *Lern*-Komponenten entgegengenommen. Eine Akquisitionskomponente implementiert Methoden wie sie in wissensbasierten Systemen häufig zum Einsatz kommen, um eine Wissensbasis aufzubauen. Dabei kann es sich beispielsweise um Heuristiken zur Interpretation der Beobachtungen handeln. Die Lernkomponenten bestehen aus maschinellen Lernverfahren sowie Mechanismen zur Transformation der im Kontext eines speziellen Lernverfahrens erzielten Ergebnisse in explizite, verfahrensunabhängige Informationen, die in das Benutzermodell einfließen können und dann zusammen mit aus anderen Lernkomponenten gewonnenen Informationen als Grundlage der Adaptionsentscheidungen dienen. Die Architektur erlaubt auch Adaptionsentscheidungen auf der Basis der untransformierten Ergebnisse der Lernkomponenten. Analog zum DOPPELGÄNGER-Ansatz werden den Eigenschaften der Datenströme bzw. den verfolgten Zielen entsprechende Algorithmen verwendet. Wie in DOPPELGÄNGER wird der automatische Aufbau von Benutzergruppen, beispielsweise durch Clustering-Verfahren, unterstützt.

Die Schwerpunkte des generischen LABOUR-Ansatzes zum Aufbau benutzeradaptiver Systeme bestehen (a) in der deutlichen gegenseitigen Trennung von Lern- und Akquisitions-Komponenten, expliziter Repräsentation des Benutzermodells und Entscheidungskomponenten sowie (b) in der Kombination von Methoden wissensbasierter Systeme mit maschinellen Lernverfahren, um potenzielle Synergieeffekte auszunutzen.

3.2.3 Diskussion

Der Fokus der beiden vorgestellten Ansätze generischer Benutzermodellierungsumgebungen liegt auf der Integration unterschiedlicher Verfahren sowohl aus dem Bereich des maschinellen Lernens, als auch—im Fall des LABOUR-Projekts—wissensbasierter Systeme in einer einheitlichen Architektur. Die Anpassung dieser Verfahren an die Anforderungen des Benutzermodellierungskontextes spielen nur dann ein Rolle, wenn sie für das Gesamtkonzept von Bedeutung sind, wie z.B. die Transformation der Lernergebnisse in explizite, von speziellen Verfahren unabhängige, Benutzermodelle.

Das Ziel der vorliegenden Arbeit kann als ein Ansatz interpretiert werden, der maschinelle Lernverfahren für Bayes'sche Netze auf den Kontext benutzeradaptiver Systeme überträgt bzw. erweitert, und dieses Werkzeug zur Repräsentation von und Inferenz unter Unsicherheit in generischer Art und Weise für den Einsatz in solchen Benutzermodellierungsumgebungen verfügbar macht. Dies erscheint insbesondere unter Berücksichtigung der wachsenden Akzeptanz und Bedeutung Bayes'scher Netze als Inferenzmechanismus in benutzeradaptiven Systemen von Interesse.

3.3 Kollaborative vs. inhaltlich-basierte Ansätze

Die bereits mehrfach angesprochenen *Empfehlungssysteme* (engl. *recommender systems*) als ein ausgezeichnetes Teilgebiet innerhalb der Forschung benutzeradaptiver Systeme haben sich aufgrund des explosionsartigen Wachstums des WWW im Laufe der letzten Jahre als einer der Schwerpunkte des erfolgreichen Einsatzes von Benutzermodellierungsmethoden in kommerziel-

len Szenarien entwickelt.[7] Benutzer kommerzieller Angebote im WWW stehen typischerweise einer für sie unüberschaubaren Fülle von Informationen bzw. Produkten gegenüber. Es ist sowohl für den Erfolg solcher Angebote als auch für einen potenziellen Kunden von entscheidender Bedeutung, dass er möglichst einfach die für ihn interessanten Produkte bzw. Informationen im Gesamtangebot lokalisieren kann. Diese Aufgabe übernehmen Empfehlungssysteme, indem sie versuchen, anhand eines impliziten oder expliziten Benutzermodells entsprechende maßgeschneiderte Zusammenstellungen von Produkten oder Informationen zu unterbreiten. Da diese Systeme eine wichtige Klasse der Anwendung maschineller Lernverfahren in benutzeradaptiven Systemen darstellen, werden im Folgenden kurz die relevanten Aspekte und Verfahren beleuchtet. Dabei ist zu beachten, dass es sich um einen Meta-Ansatz handelt, der mit verschiedenen (elementaren) Algorithmen des maschinellen Lernens ausgefüllt werden kann. Typischerweise kommen die Methode der nächsten Nachbarn und der naive Bayes'sche Klassifizierer zum Einsatz. Im Zusammenhang mit letzterem Verfahren können die in den Kapiteln 6 und 7 der vorliegenden Arbeit entwickelten Verfahren angewendet werden.

In diesem Zusammenhang werden *inhaltlich-basierte* Methoden und *kollaboratives Filtern*[8] (Alspector, Kolcz & Karunanithi, 1997; Konstan et al., 1997) unterschieden. Eine detailliertere Klassifizierung von Empfehlungssystemen, die neben diesen beiden angesprochenen, am häufigsten eingesetzten Methoden weitere elementare sowie hybride Ansätze betrachtet, wird von Burke (2002) vorgenommen.

Abbildung 3.4 visualisiert die in Empfehlungssystemen relevanten Informationen. Ziel ist es, den Inhalt des rechts unten angeordneten (blau schraffierten) Kastens, d.h., die Bewertung des aktuellen Objekts durch den aktuellen Benutzer anhand der verfügbaren Information vorherzusagen.

Der inhaltliche Ansatz (vgl. die grün markierten Bereiche (rechtes aufrechtstehendes Rechteck) in Abbildung 3.4) baut darauf auf, charakteristische Merkmale bzw. Attribute (engl. *features*) der potenziell relevanten Objekte mit den Interessen des Benutzers zur Deckung zu bringen. In den Fällen, in denen eine genügend große Überdeckung erreicht wird, kann das Objekt dem Benutzer empfohlen werden. Um eine solche Vorgehensweise zu ermöglichen, müssen die Benutzerinteressen dem System explizit bekannt sein. Dies kann entweder durch Angaben des Benutzers geschehen oder anhand der Analyse seines (Kauf-)Verhaltens unter Verwendung entsprechender Inferenz- bzw. Lerntechniken erschlossen werden. Weiterhin stellt sich die Frage, welche der Merkmale der Objekte zur Generierung der Empfehlungen herangezogen werden sollten (engl. *feature-selection problem*). Weiterhin ist es oft nicht einfach, die Merkmale eines Objekts maschinell zu extrahieren, wie z.B. im Falle von multimedialen Objekten. Soll ein solches Empfehlungssystem ohne eine vorgeschaltete Phase der Benutzerbefragung zur Interessensbestimmung auskommen, d.h., soll ein entsprechendes inhaltliches Modell mit Methoden des maschinellen Lernens aus dem Benutzerverhalten extrahiert werden, hat dieser Ansatz den Nachteil, eine gewisse Zeit zu benötigen, um ein brauchbares Benutzermodell zu erlernen.

Das kollaborative Filtern realisiert die Idee, Empfehlungen für den aktuellen Benutzer anhand der Informationen zu anderen früheren Systemnutzern, die ähnliche Interessen bzw. Eigenschaften gezeigt haben, zu generieren. Objekte, die von diesen Nutzern als interessant empfunden wurden, sollten dann im Normalfall auch für den aktuellen Benutzer von Interesse sein. Prinzipiell muss ein

[7] Einen aktuellen Überblick über Systeme und eingesetzte Techniken gibt Burke (2002).

[8] Die Terminologie ist hier in der Literatur nicht eindeutig. Oft wird kollaboratives Filtern wegen der überragenden Bedeutung diese Ansatzes mit Empfehlungssystemen gleichgesetzt (siehe z.B. Breese, Heckerman & Kadie, 1998). Außerdem erscheint die Wahl des Ausdrucks 'kollaboratives Filtern' nicht zutreffend, weshalb zunehmend verschiedene andere Bezeichnungen dieses Verfahrens verwendet werden (z.B. 'cliquen-basiert', siehe Kobsa et al., 2001).

kollaboratives Filtersystem drei Schritte implementieren: (i) Bestimmen eines Ähnlichkeitsmaßes zwischen dem aktuellen und den früheren Benutzern, (ii) Auswahl einer Menge ähnlicher Benutzer und (iii) Berechnen einer Empfehlung auf der Basis der ausgewählten ähnlichen Benutzer. Zur Implementation der Einzelschritte existiert eine Vielzahl an verschiedenen Alternativen. Einige Möglichkeiten werden von Breese et al. (1998) empirisch verglichen.

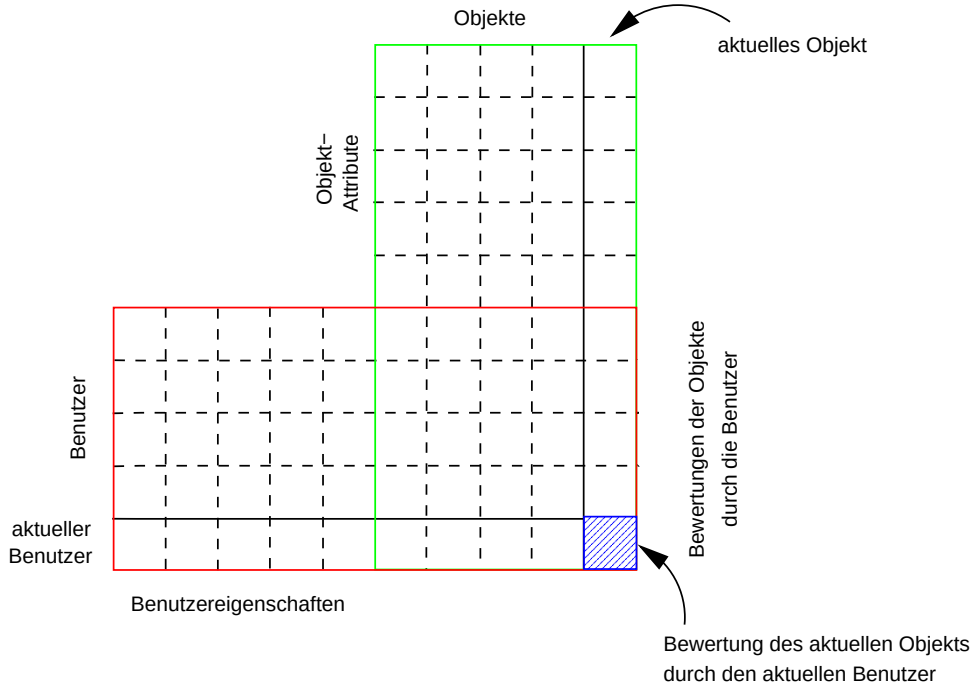


***Abbildung 3.4**: Empfehlungssysteme - inhaltlich-basierter und/oder kollaborativer Ansatz (Nach Jameson, Konstan & Riedl, 2002, weitere Erläuterungen im Text)*

Aus der Perspektive der Effizienz zur Systemlaufzeit unterscheidet Breese et al. (1998) zwei Klassen innerhalb der Verfahren des kollaborativen Filterns:

- *Speicher-basiert* (engl. *memory-based*): Verfahren dieser Klasse nutzen die komplette Benutzer-Datenbank, um Empfehlungen zu bestimmen. Typischerweise werden die Empfehlungen als gewichtete Kombinationen der Informationen der anderen Benutzer ermittelt, wobei das Ähnlichkeitsmaß in die Festlegung des Gewichts einfließt.

- *Modell-basiert* (engl. *model-based*): Im Gegensatz dazu nutzen modell-basierte Ansätze die Benutzer-Datenbank zur Konstruktion eines expliziten Modells, das dann zur Erzeugung der Empfehlungen dient. Der Vorteil hierbei ist, dass sobald dem System ein Modell zur Verfügung steht, die Benutzer-Datenbank zur Laufzeit nicht mehr konsultiert werden muss. In diesem Fall muss allerdings sichergestellt werden, dass das Modell (in regelmäßigen Abständen) aktualisiert wird, um sowohl neuen Objekten als auch neuen Benutzern Rechnung zu tragen.

Kollaboratives Filtern eignet sich im Normalfall besser für WWW-Empfehlungssysteme als inhaltlich-basierte Ansätze: Typischerweise sind in einem solchen Szenario genügend Daten von

vielen Benutzern verfügbar, die das kollaborative Verfahren erleichtern. Da keine inhaltlichen Aspekte betrachtet werden, entfällt die Abhängigkeit von (potenziell fehleranfälligen) maschinellen Methoden zur semantischen Analyse der Objekte, sowie das Feature-Selection-Problem bzw. die Modellierung komplexer inhaltlicher Konzepte. Dadurch sind kollaborative Filterverfahren weitgehend unabhängig von der Domäne, was ihren Einsatz in unterschiedlichen Systemen deutlich erleichtert. Andererseits stellen kollaborative Verfahren meist eine „Black Box“ dar. Es mangelt ihnen an Interpretierbarkeit bzw. Transparenz (vgl. Abschnitt 3.1.3.5 sowie Herlocker et al., 2000). Problematisch ist ebenso das so genannte *Problem der ersten Bewertung* (engl. *first-rater-problem*), das die Situation bezeichnet, in der ein neues Objekt in das System aufgenommen wird, und zu Beginn keine Bewertungen der Benutzer zu ihm verfügbar sind. Analog tritt dieses Problem auf, wenn ein neuer Systembenutzer mit dem System interagiert, zu dem keine Informationenvorliegen, um das Ähnlichkeitsmaßes zu berechnen.

Die beiden genannten Nachteile des kollaborativen Filterns können in vielen Fällen unter temporärer Aufgabe einiger Vorteile durch eine Kombination mit inhaltlichen Methoden zu einem hybriden Ansatz reduziert werden (Schnittbereiche der beiden Rechtecke der Abbildung). So kann beispielsweise in der initialen Phase, nach der ein Objekt in das System eingeführt wird, eine inhaltlich-basierte Verfahrensweise angewendet werden, die die Objektmerkmale mit Benutzerinteressen abgleicht (siehe beispielsweise Balabanovic, 1998).

3.4 In benutzeradaptiven Systemen eingesetzte maschinelle Lernverfahren

Im Folgenden werden einige maschinelle Lernverfahren, die bereits erfolgreich in benutzeradaptiven Systemen eingesetzt werden, diskutiert, ohne detailliert auf technische Aspekte einzugehen. Es handelt sich dabei nicht um eine erschöpfende Auflistung aller Verfahren, sondern um einen Querschnitt, der typische Einsatzszenarien und Lösungsansätze veranschaulichen soll. Die Entscheidung für ein spezielles Verfahren ist immer an die von der Domäne gestellten Anforderungen abhängig zu machen, aber es existieren allgemeine Vor- bzw. Nachteile der unterschiedlichen Methoden, die mit ihrem Einsatz im Kontext benutzeradaptiver Systeme verbunden sind. Eine entsprechende Einordnung Bayes’scher Netze folgt im nächsten Kapitel im Rahmen der Vorstellung der dieser Arbeit zugrunde liegenden Konzeption zum maschinellen Lernen Bayes’scher Netze für benutzeradaptive Systeme.

In vielen Fällen wird versucht, die Nachteile eines Verfahrens durch entsprechende Weiterentwicklungen bzw. zum Teil domänenabhängige Anpassungen zu beheben. Es ist klar, dass eine solche Einordnung nach den vorliegenden Kriterien einen sehr subjektiven Charakter und keinen Anspruch auf Allgemeingültigkeit besitzt.

3.4.1 Entscheidungsbäume

Eines der in der Praxis am erfolgreichsten eingesetzten maschinellen Lernverfahren sind *Entscheidungsbäume* (engl. *decision trees (DT)*, siehe beispielsweise Quinlan, 1986; Russell & Norvig, 1995). Zur Beliebtheit dieser Verfahren, die der Klassen der überwachten Lernverfahren angehören, trägt bei, dass sie vergleichsweise einfach zu implementieren und somit in Anwendungen zu integrieren sind. Ein Entscheidungsbaum liefert zu Objekten, die durch Attribut-Wert-Paare beschrieben werden, eine Klassifikationsentscheidung. Mit dem ID3-Algorithmus und seiner Wei-

terentwicklung C4.5 (Quinlan, 1993) stehen maschinelle Lernverfahren zur Verfügung, die bereits in vielen kommerziellen Anwendungen mit teilweise sehr großem Erfolg eingesetzt wurden. Ein erlernter Entscheidungsbaum kann als Grundlage der Konstruktion einer Regelmenge dienen, was insbesondere hinsichtlich des Lernens expliziter Benutzermodelle von Interesse ist.

Entscheidungsbäume werden auch in benutzeradaptiven Systemen häufig erfolgreich verwendet. Einer der ersten so genannten *persönlichen Assistenten*, ein von Mitchell, Caruana, Freitag, McDermott und Zabowski (1994) entwickeltes System, nutzt beispielsweise Entscheidungsbaumlernverfahren, um die Planung und Eintragung eines Termins in einen Kalender zu erleichtern. Bei Angabe einiger relevanter Informationen, wie z.B. den Teilnehmern, dem Ort u.Ä. ist das System anhand der erlernten Entscheidungsbäume beispielsweise in der Lage, die voraussichtliche Dauer vorzuschlagen, um einen entsprechenden Eintrag im Kalender vorzunehmen. Paliouras et al. (1999) verwendeten erlernte Entscheidungsbäume zur Klassifikation von Nachrichtenartikeln in solche Gruppen, die für Benutzerstereotypen von potenziellem Interesse bzw. uninteressant sind. Ebenfalls eine Klassifikationsaufgabe lösen Semeraro, Ferilli, Fanizzi und Abbattist (2001) in einen digitalen Bibliotheksszenario. Mit erlernten Entscheidungsbäumen werden die Benutzer nach Stereotypen klassifiziert, die je nach Systemkenntnis unterschiedliche, ihren Anforderungen entsprechende Systemoberflächen zur Durchführung ihrer Recherchen angeboten bekommen.

Für einen Einsatz maschineller Lernverfahren für Entscheidungsbäume in benutzeradaptiven Systemen spricht neben der Verfügbarkeit einfacher Standardverfahren insbesondere der hohe Grad an Interpretierbarkeit der Inferenz in Entscheidungsbäumen durch eine mögliche Transformation in Regelmengen. Anhand der Regeln kann eine Erklärungskomponente die Entscheidungen des Systems für den Benutzer transparent gestalten. Problematisch sind in der ursprünglichen Form des Erlernens von Entscheidungsbäumen neben der Einschränkung auf Klassifikationsaufgaben die Behandlung von Unsicherheit, das Einbringen von A-priori-Wissen in den Lernvorgang, die Behandlung fehlender Daten und das Modellieren dynamischer Domänen.

3.4.2 Künstliche neuronale Netze

Ein weiteres bekanntes und erfolgreich kommerziell eingesetztes Werkzeug des maschinellen Lernens stellen *künstliche neuronale Netze* dar (siehe z.B. Mitchell, 1997, Kap. 4). Da sie in der Lage sind mehrdimensionale nicht-lineare Funktionen zu approximieren, eignen sie sich besonders gut zur Interpretation von Sensordaten. Im Zusammenhang mit neuronalen Netzen spricht man anstelle von ‘Lernen’ von *Training*, um die Charakteristik des prinzipiellen Vorgehens zu veranschaulichen: Trainingsfälle werden sequentiell benutzt, um die freien Parameter des Netzes zu adjustieren. Somit bieten sich (Lern- bzw. Trainingsverfahren für) neuronale Netze zur Verwendung in dynamischen Domänen an.

In benutzeradaptiven Systemen werden künstliche neuronale Netze häufig für Klassifikationsaufgaben eingesetzt. So nutzt Höppner (2001) sie beispielsweise zur Klassifikation von Situationen der virtuellen Arbeitsumgebung eines Telearbeitsszenarios. Um den für die Motivation der Mitarbeiter wichtigen sozialen Kontakt mit Kollegen—z.B. der kleine „Plausch“ an der Kaffeemaschine—in solchen Systemen zu ermöglichen, muss einem Benutzer des virtuellen Büros vom System angedeutet werden, ob der gewünschte Gesprächspartner zur Zeit überhaupt geneigt ist, sich auf eine Plauderei einzulassen. Zur Bewertung der „Plauderneigung“ der Mitarbeiter werden in diesem System neuronale Netze eingesetzt, die anhand verfügbarer Informationen wie Keyboardnutzung und Umgebungsgeräuschen die Arbeitssituation klassifizieren. Beobachtungen des Systems dienen als Trainingsdaten für die neuronalen Netze. Auch Goren-Bar, Kuflik, Lev und

Shoval (2001) setzen neuronale Netze zur Klassifikation ein. Sie beschreiben einen persönlichen Agenten, der die Archivierung von Dokumenten vornimmt. Anhand einer Datensammlung von Beispielklassifikationen des Benutzers wird ein neuronales Netz erlernt, das diese Aufgabe übernehmen kann. Ahman und Waern (2001) verwenden neuronale Netze in einem hybriden Ansatz des inhaltlich-basierten und kollaborativen Filterns von Nachrichtenartikeln.

Maschinelle Lernverfahren für neuronale Netze bieten sich wie bereits angesprochen insbesondere aufgrund der Fähigkeit mit dynamischen Domänen umzugehen für benutzeradaptive Systeme an. Meist werden sie zur Behandlung von Klassifikationsaufgaben eingesetzt. Größter Nachteil ist, dass sie hinsichtlich der Interpretierbarkeit eine „Black Box“ darstellen. Es ist aufgrund ihres subsymbolischen Charakters nicht ohne weiteres möglich, den Inferenzprozess nachzuvollziehen. Es wurden in den vergangenen Jahren Erweiterungen des Konzepts neuronaler Netze entwickelt, die diese Problematik behandeln (siehe z.B. Cloete & Zurada, 1999), und es ermöglichen (interpretierbare) Regelmengen zu dem im neuronalen Netz kodierten Wissen zu erstellen. Das Einbringen von A-priori-Wissen und das explizite Modellieren individueller Unterschiede werden im Grundkonzept nicht unterstützt.

3.4.3 Induktives logisches Programmieren

Induktives logisches Programmieren (engl. *inductive logic programming, ILP*) (Muggleton, 1991) ist ein Ansatz zum Erlernen von Mengen bestehend aus Regeln erster Stufe. Er ist somit mächtiger als die Entscheidungsbaumlernverfahren aus Abschnitt 3.4.1, die lediglich Regeln ohne Variablen produzieren können. Bei diesem Verfahren wird Hintergrundwissen über die Domäne vor dem eigentlichen Lernvorgang in Form von Regeln vorgegeben, um anhand der verfügbaren Daten, neue Regeln hinzuzufügen.

Techniken des induktiven logischen Programmierens werden in den letzten Jahren verstärkt auch in benutzeradaptiven Systemen eingesetzt. Jacobs und Blockeel (2001) erzeugen mit ihrer Hilfe eine Regelmenge, die zur Erzeugung benutzerspezifischer Makros wiederholt verwendeter Sequenzen von UNIX-Kommandos genutzt wird. Kay und McCreath (2001) entwickeln im MUMILP-Projekt ein Verfahren zur automatischen Konstruktion von (komplexen) Filterregeln, wie sie in Mail-Clients zum Einsatz kommen. Die Spezifikation solcher Filter stellt oft hohe Anforderungen an typische Benutzer, die in vielen Fällen nur fähig sind, einfache, sub-optimale Regeln zu formulieren. Maschinell erlernte Regelmengen besitzen in diesem Anwendungsszenario ein großes Potential. Mit OYSTER beschreibt Müller (2002) das Konzept einer benutzeradaptiven ILP-basierten WWW-Suchmaschine. Die Besonderheit dieser Suchmaschine besteht darin, dass sie eine inhaltliche Kategorisierung in Form einer Ontologie der Domäne verwendet, die sowohl zur Akquisition der Benutzermodelle als auch zum Auffinden thematisch relevanter Informationen dient.

Der Grund des verstärkten Interesses am induktiven logischen Programmieren in der Benutzermodellierung liegt in der Möglichkeit explizite, erklärbare Modelle in Form von Regelmengen zu erlernen. Weiterhin ist es mit diesem Ansatz sehr einfach, verfügbares Hintergrundwissen in Regeln erster Stufe zu kodieren, um den Lernvorgang zu erleichtern und die erzielten Ergebnisse zu verbessern. Problematisch ist die Repräsentation von Unsicherheit. Meist wird versucht diese durch annotierte Regeln oder zusätzliche Prädikate im Schlussfolgerungsprozess zu berücksichtigen. Ebenso wird die Behandlung von dynamischen Domänen durch eine Adaption der Regelmenge anhand neuer Beobachtungen in diesem Ansatz nicht direkt durch die Standardverfahren unterstützt. Dazu muss ein zusätzlicher Mechanismus implementiert werden, um die Regelmenge entsprechend zu aktualisieren.

3.4.4 Methode der nächsten Nachbarn

In Situationen, in denen Entscheidungen anhand der Ähnlichkeit von Objekten getroffen werden können, wird häufig die Methode der *nächsten Nachbarn* (engl. *nearest neighbors*) verwendet. Mittels eines spezifizierten Ähnlichkeitsmaßes werden ein oder mehrere nächste Nachbarobjekte bestimmt, deren Eigenschaften als Grundlagen des Entscheidungsprozesses dienen. Auf den Benutzermodellierungskontext übertragen, kann es sich bei den betrachteten Objekten sowohl um Benutzer als auch um andere relevante Aspekte der modellierten Domänen handeln. Insbesondere in Empfehlungssystemen nimmt das Verfahren der nächsten Nachbarn eine bedeutetende Rolle ein. Dabei kann es in beiden Varianten—kollaborativ oder inhaltlich-basiert – eingesetzt werden: Einmal basiert die Empfehlung auf dem ähnlichen Verhalten von Benutzern, im anderen Fall liegt die Annahme zugrunde, dass ein einzelner Benutzer ähnliche Dinge bevorzugt, wie z.B. CDs eines bestimmten Genres.

Beispiele solcher Systeme sind das bereits mehrfach erwähnt NEWSDUDE (Abschnitt 2.1.4), bei dem Nächste-Nachbarn-Verfahren im Kurzzeitgedächtnis zur Identifikation sehr ähnlicher Nachrichtenartikel zum Einsatz kommen, und CASPER (Bradley, Rafter & Smyth, 2000), einem personalisierten Agenten zur Arbeitsstellensuche. Eine Erweiterung des allgemeinen Verfahrens hinsichtlich Domänen, in denen nur positive Rückmeldungen des Benutzers zur Verfügung stehen, d.h., wenn lediglich bekannt ist, welche Objekte den Benutzer interessieren, nicht welche für ihn uninteressant sind, stellen Schwab und Kobsa (2002) im Rahmen des Projektes ELFI vor, das Wissenschaftler bei der Suche nach neuen Akquisitionsmöglichkeiten von Forschungsprojekten unterstützt.

Nächste-Nachbarn-Verfahren werden in benutzeradaptiven Systemen oft wegen ihrer Eigenschaft, bereits nach wenigen Beobachtungen akzeptable Ergebnisse zu erzielen, eingesetzt. Auch erste Ansätze, sie zur Basis von Erklärungskomponenten heranzuziehen, werden in der Forschung verfolgt (Herlocker et al., 2000). Der entscheidende Punkt einer erfolgreichen Anwendung ist die adäquate Spezifikation des Ähnlichkeitsmaßes. Meist können die benötigten Parameter nur in empirischen Studien ermittelt werden. Nächste-Nachbarn-Verfahren werden—teilweise aufgrund ihrer einfachen Konzeption—fast ausschließlich in Empfehlungssystemen eingesetzt. Problematisch ist hingegen die Initialisierung des Modells. Ebenso ist Hintergrundwissen nur schwer in diesem Ansatz in die impliziten Benutzermodelle einzubringen. Eine Möglichkeit, die aber nicht in allen Einsatzszenarien zu realisieren ist, sind „hypothetische Nachbarn“, die manuell—beispielsweise von einem Domänenexperten—vorgegeben werden.

3.4.5 Fall-basiertes Schließen

Ein mit dem Verfahren der nächsten Nachbarn verwandter Ansatz ist das *fall-basiertes Schließen* (engl. *case-based reasoning, CBR*). Es verwendet ebenfalls ein Ähnlichkeitsmaß zum Vergleich von Fallbeschreibungen, die hier von deutlich komplexerer Natur sein können. Fälle werden in diesem Ansatz meist mit symbolischen Methoden der Wissensrepräsentation beschrieben. Ähnliche Fälle werden zur Lösung eines neuen Problems herangezogen, wobei dazu gegebenfalls die Lösungswege der alten Fälle an die neue Situation angepasst werden.

Die Flexibilität des CBR-Ansatzes wird auch in der Verwendung in unterschiedlichsten benutzeradaptiven Systemen deutlich. So stellt Gervas (2001) ein System vor, dass die Vorlieben eines Benutzers bezüglich Gedichten erkennt und diese Informationen zur (semi-)automatischen

Erzeugung neuer Gedichte ausnutzen kann. Auch CASPER, der personalisierte Job-Agent, der seine Benutzer bei der Suche eines neuen Arbeitsplatzes unterstützt, setzt CBR-Methoden ein. Waszkiewicz, Cunningham und Byrne (1999) entwickelten einen auf CBR-Techniken basierenden personalisierten Agenten, der zur Unterstützung bei der Reiseplanung eingesetzt wird.

Fall-basiertes Schließen kann zur adaptiven Unterstützung von Benutzern bei komplexen Aufgaben eingesetzt werden. Ein solches System kann anhand der gesammelten Fälle sein aktuelles Vorgehen in Analogie zum früheren Vorgehen (anderer Benutzer) erklären. Ähnlich wie beim Nächsten-Nachbarn-Ansatz stellt die Fallsammlung ein implizites Benutzermodell dar. Wie dort stellt sich auch hier die Frage der Initialisierung eines solchen Systems, wenn es noch nicht möglich war, genügend Fälle zu analysieren. Prinzipiell können auch in diesem Ansatz „hypothetische Fälle“ zur Initialisierung und dem Einbringen von A-priori-Wissen genutzt werden.

3.4.6 Diskussion

Tabelle 3.1 beinhaltet eine Zusammenfassung der Bewertung der in diesem Abschnitt diskutierten Verfahren bezüglich der Eignung für benutzeradaptive Systeme. In einem Vorgriff auf den ersten Abschnitt des anschließenden Kapitels wurden Bayes’sche Netze in der letzten Spalte mit aufgenommen. Die einzelnen Bewertungen beziehen sich wie angesprochen auf die jeweiligen Grundversionen der Methoden.

Man erkennt, dass jedes einzelne Verfahren Vor- und Nachteile besitzt. Die Entscheidung für ein bestimmtes ist abhängig von den Anforderungen und Gegebenheiten der zu modellierenden Domäne. Dies lässt sich über die in diesem Abschnitt vorgestellte Auswahl an Verfahren hinaus verallgemeinern. Oftmals können einige der Nachteile einer Methode durch domänenspezifische Erweiterungen bzw. Modifikationen ihrer Grundversion behoben werden.

Auch Bayes’sche Netze bzw. die zugehörigen maschinellen Lernverfahren erfüllen nicht alle potenziellen Anforderungen benutzeradaptiver Systeme, obwohl sie durch die in Abschnitt 1.2 angeführten Punkte in vielen Szenarien gut als Inferenzmechanismus geeignet sind. Mit den in der vorliegenden Arbeit entwickelten Verfahren wird ein Beitrag zur Verbesserung der Eignung maschineller Lernverfahren für Bayes’sche Netze in benutzeradaptiven Systemen geleistet.

Kriterien	**DT**	**KNN**	**ILP**	**NN**	**CBR**	**BN**
Wenige Trainingsdaten	$\oplus$	$\ominus$	$\oplus$	$\oplus$	$\odot$	$\odot$
Fehlende Daten	$\ominus$	$\odot$	$\oplus$	$\odot$	$\ominus$	$\oplus$
Inter-individuelle Unterschiede	$\odot$	$\odot$	$\oplus$	$\odot$	$\odot$	$\oplus$
Dynamische Domänen	$\odot$	$\oplus$	$\ominus$	$\odot$	$\ominus$	$\oplus$
Komplexität / Effizienz im Online-Betrieb	$\oplus$	$\odot$	$\ominus$	$\oplus$	$\odot$	$\odot$
Interpretierbarkeit	$\oplus$	$\ominus$	$\oplus$	$\odot$	$\odot$	$\odot$
Integration von A-priori-Wissen	$\odot$	$\ominus$	$\oplus$	$\odot$	$\odot$	$\oplus$

***Tabelle 3.1**: Eignung verschiedener maschinelller Lernverfahren für benutzeradaptive Systeme*
*(Abkürzungen der Verfahren: Entscheidungsbäume **(DT)**, künstliche neuronale Netze **(ANN)**, induktives logisches Programmieren **(ILP)**, nächste Nachbarn **(NN)**, fall-basiertes Schließen **(CBR)**, Bayes'sche Netze **(BN)**; Bewertungen $\oplus$: positiv, $\odot$: neutral bzw. nicht zu bewerten, $\ominus$: negativ)*

4 Maschinelles Lernen Bayes'scher Netze für benutzeradaptive Systeme - Konzeption und grundlegende Verfahren

Im Folgenden wird die der Arbeit zugrunde liegende Gesamtkonzeption des maschinellen Lernens Bayes'scher Netze für benutzeradaptive Systeme vorgestellt, die den Rahmen für die in den anschließenden Kapiteln im Detail beschriebenen und evaluierten Verfahren bildet. Es schließt sich eine Diskussion der Eignung existierender maschineller Lernverfahren für Bayes'sche Netze unter Berücksichtigung der in Kapitel 3 identifizierten kritischen Aspekte sowie der diesbezüglichen Beiträge dieser Arbeit an. Nach einigen allgemeinen Bemerkungen und der Festlegung der Notation zum maschinellen Lernproblem Bayes'scher Netze werden die grundlegenden Algorithmen sowohl des Batchlernens als auch der Adaption eingeführt.

4.1 Eine integrative Konzeption des maschinellen Lernens Bayes'scher Netze für benutzeradaptive Systeme

Wie in Abschnitt 2.6 dargelegt wurde, werden zwar in einigen benutzeradaptiven Systemen maschinelle Lernverfahren für Bayes'schen Netze eingesetzt, es existiert aber bislang kein integrativer Ansatz, der entsprechende existierende Verfahren für den Benutzermodellierungskontext anpasst und/oder neue Methoden bereitstellt, die in der Lage sind, mit den in Abschnitt 3.1.3 diskutierten Anforderungen umzugehen. Typischerweise werden bislang überwiegend Standardverfahren verwendet—meist beschränkt auf den wichtigen (Teil-)Fall des Lernens der bedingten Wahrscheinlichkeiten. Auch das Erlernen der kausalen Struktur einer Domäne kann in benutzeradaptiven Systemen von Interesse sein, um die Zusammenhänge verschiedener Aspekte der Benutzermodelle zu identifizieren und bei den Adaptionsentscheidungen entsprechend berücksichtigen zu können.

Im folgenden Abschnitt wird eine generische, integrative Konzeption zum maschinellen Lernen Bayes'scher Netze für benutzeradaptive Systeme vorgestellt, die in den folgenden Kapiteln mit alternativen—teilweise neu entwickelten—Verfahren instanziiert wird.

4.1.1 Überblick

Die in Abbildung 4.1 schematisch dargestellte Konzeption (Wittig, 2002) lässt sich hinsichtlich verschiedener *Dimensionen* charakterisieren, die im Weiteren beleuchtet werden. Dabei werden typischerweise von einem potenziellen benutzeradaptiven Zielsystem nur eine eingeschränkte Auswahl der angebotenen Optionen genutzt. Das mit dieser Gesamtkonzeption verfolgte Ziel ist es, eine Sammlung separat anwendbarer Methoden im Sinne des „Werkzeugkastenprinzips“ bereitzustellen, die bei Bedarf kombiniert werden können.

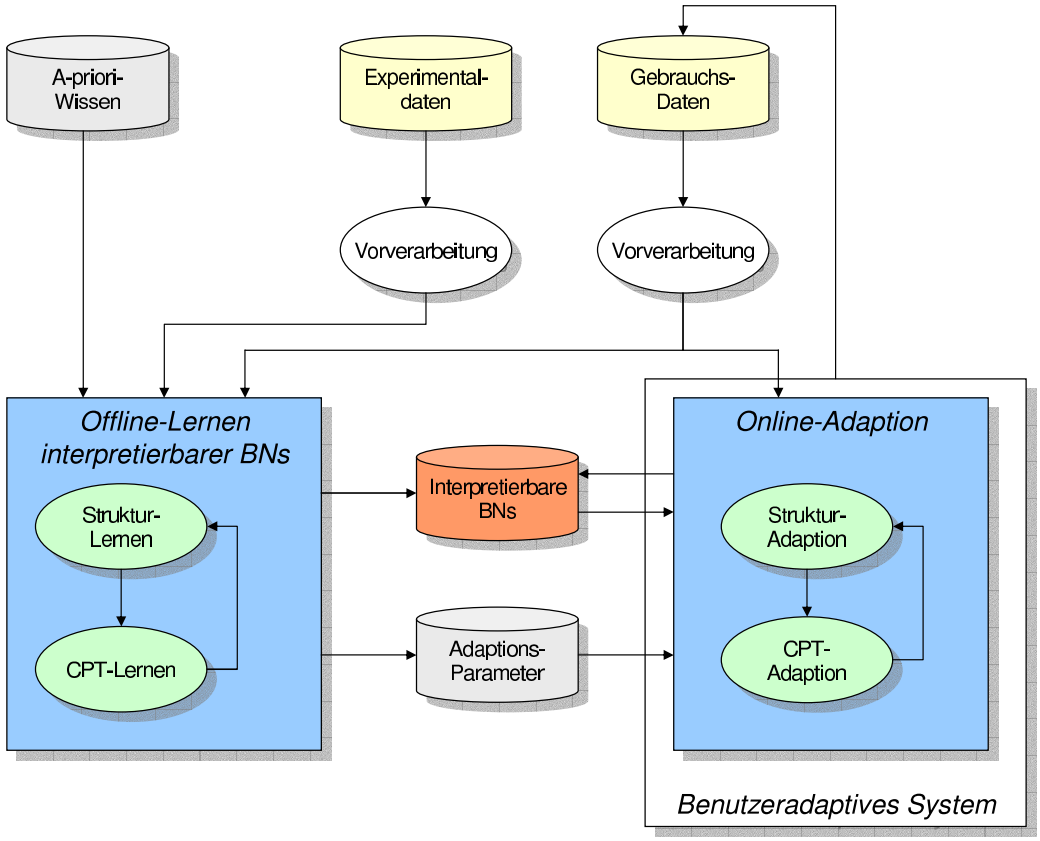


***Abbildung 4.1**: Eine integrative Konzeption zum maschinellen Lernen Bayes'scher Netze für benutzeradaptive Systeme*
(Die Pfeile repräsentieren den Informationsfluss zwischen den einzelnen Komponenten. Ellipsen stellen Algorithmen dar, Zylinder modellieren Daten bzw. Informationen und Rechtecke symbolisieren größere Einheiten, die weitere Komponenten umfassen können—zusätzlich zu den abgebildeten. Inhaltlich eng zusammenhängende Teile der Konzeption sind in gleicher Farbe (Grauschattierung) kodiert.)

Offline-Lernen und Online-Adaption Die ersten beiden (in der Abbildung dunkelgrau unterlegten) Dimensionen sind—in Analogie zur allgemeinen Situation im maschinellen Lernen (vgl. Abschnitt 3.1.1)—das *(Offline-)Lernen* und die *(Online)-Adaption*. Im Verlauf einer Offline-Phase wird üblicherweise ein allgemeines Benutzermodell auf der Basis der verfügbaren Daten einer

Vielzahl von Benutzern erlernt, welches als Ausgangspunkt der Interaktion des Zielsystems mit einem neuen Benutzer dient (vgl. Abschnitt 3.1.3.1). Das initiale, allgemeine Bayes'sche Netz wird unter Verwendung entsprechender Adaptionsverfahren an den individuellen Benutzer angepasst und kann nach Beendigung der Interaktion für die zukünftige Verwendung mit diesem Nutzer gespeichert werden. Alternativ kann bei Verfügbarkeit entsprechender Daten anstelle des allgemeinen Modells ein individuelles initiales Modell erlernt werden, das im Rahmen des Adaptionsvorgangs gegebenenfalls an den aktuellen Kontext angepasst wird. Ein Beispiel eines solchen Systems ist SWIFTFILE (Segal & Kephart, 2000), ein Assistenzsystem, das die (semi-)automatische Archivierung eingegangener E-Mails ermöglicht. Beim Installationsvorgang von SWIFTFILE werden die bestehenden E-Mail-Ordner analysiert, um ein initiales Modell des Ablageverhaltens des Benutzers zu erlernen, das im weiteren Einsatz verfeinert bzw. an veränderte Verhaltensweisen des Benutzers angepasst wird.

Zusätzlich zum (allgemeinen) Benutzermodell kann die Offline-Phase Parameter für den Mechanismus der Adaption an den individuellen Benutzer und/oder den aktuellen Kontext liefern. Die zugrunde liegende Idee ist dabei, dass unterschiedliche Teile des Bayes'schen Netzes mit unterschiedlicher Geschwindigkeit an den individuellen Benutzer adaptiert werden. Eine Beobachtung im Anweisungsexperiment (Abschnitt 2.2.1) war u.a., dass die Versuchspersonen weniger unterschiedliche Fehlerhäufigkeiten produzierten, aber hinsichtlich der Ausführungsgeschwindigkeiten sehr stark individuell differierten. Deshalb erscheint es sinnvoll, den Teil des Netzes, der das Verhalten der Benutzer bezüglich der Ausführungsgeschwindigkeiten modelliert, schneller an den individuellen Benutzer anzupassen, als jenen Teil, der für die Fehlerraten zuständig ist.

Experimentelle Daten und Gebrauchsdaten Zwei weitere Dimensionen der Konzeption betreffen die Art der verwendeten bzw. verfügbaren Lerndaten. Man unterscheidet diesbezüglich zwischen *experimentellen Daten* und *Gebrauchsdaten* (siehe oberer Teil der Abbildung). Experimentelle Daten werden in kontrollierten Umgebungen wie beispielsweise den in Abschnitt 2.2 beschriebenen Experimenten gesammelt. Gebrauchsdaten werden im Rahmen der „echten" Interaktion zwischen Benutzer und System erhoben. Solche Daten zeichnen sich oft durch unvollständige Datensätze und eine schlechte Eignung zur Behandlung selten auftretender Situationen aus. Um ein adäquates (Teil-)Modell zur Bearbeitung solcher Situationen zu erlernen, ist eine große Menge an Daten notwendig. Kleine Datensätze führen diesbezüglich zu wenig robusten Modellierungen. Dagegen spiegeln im Rahmen von Experimenten gesammelte Daten häufig nicht adäquat die reale Anwendungssituation wider. Oftmals ist eine Kombination der beiden Datenformen vorhanden. Sie kann mit dem beschriebenen Offline-/Online-Ansatz—nach entsprechender Vorverarbeitung (z.B. zur Diskretisierung kontinuierlicher Variablen)—beispielsweise durch (a) Lernen eines (allgemeinen) Benutzermodells anhand experimenteller Daten mit (b) anschließender Adaption auf der Basis von Gebrauchsdaten behandelt werden.

Lernen der bedingten Wahrscheinlichkeiten und der Struktur Aufgrund des Aufbaus Bayes' scher Netze bestehend aus zwei Teilkomponenten sind auch die Lern- und Adaptionsaufgabe 2-dimensional: (a) das Lernen bzw. die Adaption der Struktur und (b) das Lernen bzw. die Adaption der bedingten Wahrscheinlichkeiten der CPTs. Um die in Abschnitt 3.1.3 formulierten Probleme zu behandeln, spielt das Einbringen von a priori vorhandenem Wissen eine große Rolle in der vorliegenden Konzeption. Insbesondere stehen hierbei—wegen ihrer entscheidenden Bedeutung für den erfolgreichen Einsatz maschineller Lernverfahren in benutzeradaptiven Systemen—die Ver-

besserung der Interpretierbarkeit und die Problematik von zu wenigen verfügbaren Trainingsdaten im Vordergrund des Interesses.

Grad der Interpretierbarkeit Wie in Abschnitt 3.1.3.5 argumentiert, spielt die Eigenschaft der Interpretierbarkeit der erlernten Benutzermodelle in Form der Bayes'schen Netze aus mehreren Gründen eine wichtige Rolle. Diesem Ziel wird in der vorgeschlagenen Konzeption im Wesentlichen durch (neu entwickelte) Methoden zum Einbringen des vorhandenen Domänenwissens Rechnung getragen. Selbst wenn das Ausnutzen des A-priori-Wissens keine Verbesserung der Performanz der erlernten Benutzermodelle bewirkt, lohnt es sich möglicherweise alleine aufgrund der erhöhten Interpretierbarkeit der erzielten Resultate.

4.1.2 Eignung existierender Verfahren des maschinellen Lernens Bayes'scher Netze für den Einsatz in benutzeradaptiven Systemen

Die Eigenschaften Bayes'scher Netze als Inferenzmechanismus, die für einen Einsatz in benutzeradaptiven Systemen relevant sind, sowie Beispiele solcher Systeme wurden bereits in den Abschnitten 1.2 bzw. 2.6 aufgelistet und ausführlich diskutiert. An dieser Stelle sollen einige Eigenschaften maschineller Lernverfahren für Bayes'sche Netze bezüglich der in Abschnitt 3.1.3 formulierten Problemstellungen detailliert beleuchtet werden. Neben maschinellen Lernverfahren werden auch dynamische Bayes'sche Netze wegen ihrer großen Bedeutung in benutzeradaptiven Systemen in die Diskussion einbezogen. Der Fokus dieser Arbeit liegt dennoch auf der induktiven Lernaufgabe zur Ermittlung des Benutzermodells in Form der Struktur und den zugehörigen bedingten Wahrscheinlichkeiten. Die Details zu den entsprechenden Standardverfahren sind Inhalt der folgenden Abschnitte dieses Kapitels.

Die Problematik einer relativ *geringen Menge an verfügbaren Trainingsdaten* zur Akquisition des Benutzermodells kann mit Bayes'schen Netzen in vielen Fällen wie in Abschnitt 3.1.3.1 vorgeschlagen behandelt werden. Ein entweder manuell auf der Basis theoretischer Überlegungen spezifiziertes oder anhand der Daten anderer Benutzer maschinell erlerntes Bayes'sches-Netz-Benutzermodell wird als Ausgangspunkt des Adaptionsvorgangs an den individuellen Interaktionspartner verwendet. Es stehen Adaptionsverfahren für die bedingten Wahrscheinlichkeiten Bayes'scher Netze zur Verfügung, die anhand einer einzelnen Beobachtung eine Modellanpassung vornehmen können. Schwierig zu behandeln sind allerdings Situationen, in denen weder Hintergrundwissen noch Daten anderer Benutzer vorliegen. Dann eignen sich Bayes'sche Netze schlecht für die Benutzermodellierungsaufgabe, da es im Vergleich zu anderen Verfahren wie z.B. dem Nächsten-Nachbarn-Ansatz relativ lange dauert, bis im Rahmen der Adaptionsverfahren ein Modell erlernt wurde, das brauchbare Ergebnisse liefert.

Lernverfahren für Bayes'sche Netze eignen sich im Allgemeinen gut zur Berücksichtigung *inter-individueller Unterschiede* zwischen den einzelnen Benutzern. Eine einfache Möglichkeit ist die Aufnahme von expliziten individuellen Parametervariablen wie im Bayes'schen Netz im Beispiel aus Abschnitt 2.4.2. Einmal ermittelt, können die Werte dieser Parametervariablen vom System benutzerspezifisch abgelegt und für zukünftige Interaktionen mit diesem Benutzer verwaltet werden. Weiterhin kann wie beschrieben ein allgemeines Ausgangsnetz erlernt werden, das danach mit Hilfe der Adaptionsalgorithmen für Bayes'sche Netze an die einzelnen Benutzer angepasst wird. Hier erhält man allerdings keine derart kompakte explizite Repräsentation der individuellen Unterschiede, wie dies beim Einsatz von Parametervariablen der Fall ist. Außerdem berücksichtigen die existierenden Adaptionsverfahren bislang keinerlei Informationen zu einzelnen spe-

zifischen Aspekten der Benutzermodelle. So existieren—wie bereits beispielhaft beschrieben—Eigenschaften, in denen alle Benutzer weitestgehend übereinstimmen, so dass lediglich geringe Anpassungen vorgenommen werden müssen, hinsichtlich anderer Teile der Benutzermodelle kann es allerdings notwendig sein, radikalere Veränderungen der Modelle im Rahmen des Anpassungsprozesses durchzuführen.

Schäfer und Weyrath (1997) und Schäfer (1998) haben gezeigt, dass Benutzermodelle in Form dynamischer Bayes'scher Netze ein adäquates Mittel zur *Repräsentation temporaler Aspekte* in Domänen benutzeradaptiver Systeme sind. Die existierenden CPT-Adaptionstechniken Bayes' scher Netze bieten die Möglichkeit, ältere Trainingsdaten „zu vergessen". Ein handhabbares Verfahren, um die Struktur Bayes'scher Netze in dynamischen Domänen unter Berücksichtigung der besonderen Anforderungen benutzeradaptiver Systeme anzupassen, existiert bislang nach Wissen des Autors nicht.

Hinsichtlich der *Komplexität* der induktiven maschinellen Lernverfahren für Bayes'sche Netze müssen vier Fälle unterschieden werden: (a) das Lernen der bedingten Wahrscheinlichkeiten mit vollständigen Daten, (b) das Lernen der bedingten Wahrscheinlichkeiten mit unvollständigen Daten, (c) das Lernen der Struktur (inklusive der bedingten Wahrscheinlichkeiten) mit vollständigen Daten und (d) das Lernen der Struktur mit unvollständigen Daten. Für Fall (a) existieren effiziente, einfache Lernverfahren. Die restlichen Lernaufgaben (b) - (d) erfordern aufwendigere Methoden, die im Allgemeinen nicht zum Einsatz zur Laufzeit eines Systems geeignet sind, sondern in einen Vorverarbeitungsschritt ausgelagert werden müssen. Die existierenden Adaptionsverfahren zur Anpassung der bedingten Wahrscheinlichkeiten stellen in dieser Hinsicht aufgrund ihrer Effizienz kein Hindernis dar. In diesem Zusammenhang sind wiederum Methoden zur Anpassung der Strukturen (beispielsweise durch wiederholtes Neulernen) problematisch, die nicht für einen Einsatz in Laufzeitszenarien geeignet sind.

Die kausale Interpretation der erlernten Strukturen Bayes'scher Netze eignet sich als Grundlage für Erklärungskomponenten in benutzeradaptiven Systemen, um die *Interpretierbarkeit* bzw. Transparenz des Systemverhaltens zu verbessern (vgl. Abschnitt 2.1.7). Da das Lernproblem Bayes'scher Netze als hochdimensionales Suchproblem mit typischerweise vielen lokalen Optima angesehen werden kann, können Standardverfahren Ergebnisse liefern, die—wenn überhaupt—nur sehr schwer zu interpretieren sind, obwohl sie hohe numerische Genauigkeiten bei der Inferenz erzielen können. Die existierenden Lernverfahren berücksichtigen diese Problematik bisher nur in geringem Maße.

Maschinelle Lernverfahren für Bayes'sche Netze sind gut geeignet, um mit unsicheren Daten umzugehen, wie sie häufig im Kontext benutzeradaptiver Systeme vorkommen. *Daten impliziten Charakters* führen im Falle von Klassifikationsproblemen zur Verwendung komplexer *unüberwachter Lernverfahren* (meist im Zusammenspiel mit einem naiven Bayes'schen Klassifizierer, vgl. Abschnitt 2.1.4), stellen aber solange sie in einer Vorverarbeitungsphase bearbeitet werden kein Hindernis dar. Ein typisches Beispiel hierfür ist ein benutzeradaptives Nachrichtensystem, das keine oder nur unvollständige Rückmeldungen des Benutzers darüber erhält, ob ein vom System als interessant eingestufter Artikel tatsächlich vom Benutzer als interessant bewertet wird, oder lediglich implizite Rückmeldungen verfügbar sind, wie etwa die Lesedauer, Scrollaktionen usw., die im Normalfall zwar in Korrelation mit dem Benutzerinteresse stehen, aber zum Teil dennoch stark mit Unsicherheit behaftet sind. Mit unüberwachtem Lernen kann es trotzdem möglich sein, ein Modell zur Klassifikation von Nachrichtenartikeln in die beiden Gruppen 'interessant' vs. 'nicht interessant' zu erlernen.

Das Einbringen von *a priori vorhandenem Wissen* ist in einem Bayes'schen-Netz-Lernszenario auf unterschiedliche Art und Weise möglich. Dabei spielen im Wesentlichen zwei Aspekte eine Rolle: (a) die kausale Interpretation der Kanten und (b) der Bayes'sche Lernansatz, auf den detailliert in Abschnitt 4.2.3 eingegangen wird. Anschaulich beschrieben bedeutet der Bayes'sche Lernansatz, dass man eine Vorstellung eines Modells mit einer quantifizierbaren Konfidenz hat und diese Einschätzung dann im Licht neuer Informationen (Trainings- bzw. Adaptionsdaten) entsprechend anpasst. Somit stellt der Bayes'sche Ansatz eine natürliche Form der Kombination von A-priori-Wissen und (maschinell gelernter) neuer Information dar. Daneben sind die in Abschnitt 3.1.3.7 aufgezeigten allgemein üblichen Methoden auch im Zusammenhang mit maschinellen Lernverfahren Bayes'scher Netze anwendbar.

4.2 Grundkonzepte des maschinellen Lernens Bayes'scher Netze

Im Folgenden wird die Aufgabe des maschinellen Lernens Bayes'scher Netze in den zugehörigen allgemeinen Konstruktionsprozess eingeordnet und formalisiert. Es werden wichtige Aspekte des Lernens Bayes'scher Netze betrachtet, die bei der Entwicklung der Lernalgorithmen in den anschließenden Kapiteln eine Rolle spielen.

4.2.1 Prototypischer Konstruktionsprozess

Bouckaert (1995) beschreibt den prototypischen Konstruktionsprozess eines Bayes'schen Netzes. Schäfer (1998) tut dies für den Spezialfall des Einsatzes dynamischer Bayes'scher Netze in benutzeradaptiven Dialogsystemen. Mahoney und Laskey (1996) beschreiben den Konstruktionsprozess komplexer Netze als Knowledge-Engineering-Problem. Zur Einordnung der potenziellen Anwendung maschineller Lernverfahren innerhalb des Konstruktionsprozesses wird im Folgenden die erste, allgemeinste der Varianten einer (manuellen) Konstruktion (von oder in Zusammenarbeit mit Experten) diskutiert.

Abbildung 4.2 zeigt die vier Stufen des prototypischen Konstruktionsprozesses eines Bayes' schen Netzes, der als „Lebenszyklus“ angesehen werden kann.

1. *Spezifikation der Variablen:* Der erste Schritt der Konstruktion besteht in der Festlegung der Variablen des Bayes'schen Netzes. Wie Heckerman (1998) diesbezüglich unterstreicht, handelt es sich dabei um ein nicht-triviales Problem. Heckerman weist darauf hin, dass zur Spezifikation der Variablen u.a. die mit dem Einsatz des zu konstruierenden Systems verfolgten Ziele zu berücksichtigen sind und dass entschieden werden muss, welche Teilmenge der potenziell möglichen Variablen in das Modell aufgenommen werden soll, um die vorgegebenen Modellierungsziele zu erreichen.

2. *Spezifikation der Struktur:* Danach muss die Struktur des Bayes'schen Netzes festgelegt werden. Hierbei wird häufig—wie auch in dieser Arbeit—die Heuristik der kausalen Interpretation der Kanten angewendet. Existiert ein direkter kausaler Zusammenhang zwischen zwei Variablen (nach der Meinung des/der Experten), so wird die entsprechende Kante in die Struktur eingefügt. In den meisten praktisch relevanten Anwendungssituationen führt diese Vorgehensweise zu einer Struktur, die die in der Domäne vorhandenen bedingten Unabhängigkeiten im Sinne des d-Separationskriteriums (siehe Abschnitt 2.1.2) widerspiegelt

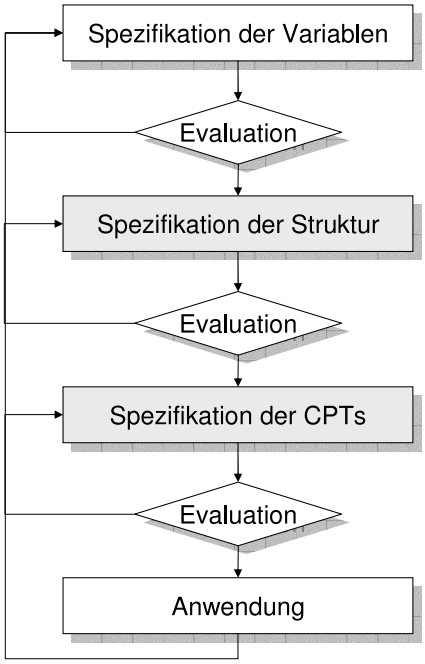


***Abbildung 4.2**: Konstruktionsprozess eines Bayes'schen Netzes*
(Die Pfeile geben die Bearbeitungsreihenfolge an.)

(Heckerman, 1998). Dabei muss insbesondere beachtet werden, dass keine Zyklen entstehen.[1]

3. *Spezifikation der CPTs:* Anschließend wird der zweite zentrale Bestandteil eines Bayes' schen Netzes, die CPTs, festgelegt. Da die Anzahl der bedingten Wahrscheinlichkeiten der CPT einer Variable exponentiell mit der Anzahl der Elternvariablen wächst, kann dieser Schritt sehr aufwendig oder im Extremfall nicht mehr praktikabel manuell durchführbar sein. Darüber hinaus hat sich gezeigt, dass selbst Experten oft Probleme mit der Spezifikation exakter bedingter Wahrscheinlichkeiten haben (Kahneman et al., 1982; Druzdzel & van der Gaag, 2000). Obwohl unterstützende Techniken (von Winterfeldt & Edwards, 1986; Morgan & Henrion, 1990; van der Gaag et al., 1999) entwickelt wurden, stellt dieser dritte Schritt innerhalb des (manuellen) Konstruktionsprozesses oft das entscheidende Teilproblem dar.

4. *Anwendung:* Nach Abschluss der vorangehenden Konstruktionsschritte kann das Bayes'sche Netz im Zielsystem eingesetzt werden. Es können Erfahrungen und/oder Daten gesammelt werden, die wiederum zur Evaluation und gegebenenfalls zur Revision oder Adaption des konstruierten Netzes verwendet werden, um nicht antizipierte Abweichungen oder—nach längerem Einsatz—Veränderungen der Eigenschaften der Domäne Rechnung zu tragen. Oft stellt sich zu diesem Zeitpunkt erst heraus, ob das konstruierte Bayes'sche Netz in der vorliegenden Form im Zusammenspiel mit den verwendeten Inferenzverfahren den Zeitanforderungen der Anwendungssituation entspricht.

[1] Eine Möglichkeit, Zyklen zu vermeiden, ist der Einsatz von dynamischen Bayes'schen Netzen. (Indirekte) Einflüsse einer Variablen auf sich selbst können oft durch die Modellierung als dynamische Variable behandelt werden.

Die Problematik der *Diskretisierung* der Variablen, d.h., die Festlegung der Anzahl und der Intervalle der Variablenzustände im Falle kontinuierlicher Wertebereiche, kann sowohl im ersten, zweiten und/oder dritten Schritt behandelt werden. Es existieren bereits Verfahren, die automatisch (im Rahmen von Lernverfahren) eine möglichst optimale Diskretisierung ermitteln (Friedman & Goldszmidt, 1996; Kozlov & Koller, 1997). Die Lösung dieser Problemstellung kann eine sehr wichtige Rolle bezüglich der Qualität des Bayes'schen Netzes spielen. Wird eine ungeeignete Diskretisierung gewählt, so können im Extremfall möglicherweise die probabilistischen Zusammenhänge vom Netz nicht mehr repräsentiert werden. Da Standardverfahren existieren, um Einflüsse der genannten Verfahren auf die Ergebnisse auszuschließen, wird bezüglich der in dieser Arbeit durchgeführten Analysen von bereits diskretisierten Datensätzen ausgegangen.

Nach jeder Stufe des beschriebenen „Lebenszykluses" kann der aktuelle Stand des Konstruktionsprozesses bzw. die aktuelle Performanz des Netzes evaluiert werden. Wird entschieden, dass in einer der Konstruktionsstufen Nachbesserungen oder eine komplette Revision der Modellierung empfehlenswert sind, kann auf die früheren Stufen des Designprozesses zurückgekehrt werden. In der Praxis werden üblicherweise mehrere Schleifen (innerhalb) des Prozesses durchlaufen bis eine zufriedenstellende Lösung erzielt wird.

Betrachtet man den beschriebenen Konstruktionsprozess unter dem Aspekt eines potenziellen Einsatzes maschineller Lerntechniken, so bieten sich der zweite und dritte Schritt (in der Abbildung grau unterlegt) an. Insbesondere das automatische, maschinelle Lernen der CPT-Einträge (Schritt 2) anhand einer Datenbank von Lernfällen sollte in den meisten Fällen zu einer erheblichen Vereinfachung und Beschleunigung des Konstruktionsprozesses führen. Zusätzlich geht damit erfahrungsgemäß häufig auch eine Verbesserung der Verlässlichkeit bzw. Qualität des konstruierten Bayes'schen Netzes einher—sofern genügend Daten der zu modellierenden Domäne zur Verfügung stehen. Auswirkungen subjektiver (Fehl-)Annahmen der Experten auf die Performanz des Modells können in dieser Weise im Rahmen des Konstruktionsprozesses verringert oder gänzlich vermieden werden. Obwohl es durch die kausale Interpretation der Kanten oft recht einfach möglich ist, die Struktur des Bayes'schen Netzes zu spezifizieren, kann es in manchen Fällen interessant sein, die Ergebnisse der Strukturlernverfahren im Sinne der Wissensentdeckung zu analysieren, um so möglicherweise bislang noch nicht erkannte Eigenschaften der Domäne zu identifizieren und im weiteren Konstruktionsprozess zu berücksichtigen. In Schritt 3 bietet es sich beispielsweise auch an, eine manuell erstellte Struktur durch den Einsatz maschineller Lernverfahren zu modifizieren. Dadurch fließen die vorhandenen empirischen Daten in das Ergebnis ein und erhöhen im Allgemeinen die Performanz des manuell konstruierten Bayes'schen Netzes. Maschinelle (Online-)Adaptionsverfahren als Spezialfall maschineller Lernverfahren können auch im Rahmen der Adaption des konstruierten Bayes'schen Netzes nach bzw. während der Anwendung im Zielsystem eingesetzt werden. Dazu müssen im Systembetrieb Daten gesammelt werden, die sequentiell ausgewertet werden können, um das verwendete Netz adäquat an auftretende Veränderungen anzupassen. In Abbildung 4.2 entspricht dies einer Schleife bestehend aus den Schritten 4 - 2 - 3. Im Gegensatz zum Offline-Konstruktionsprozess kommen hier in den Schritten 2 und 3 keine Batchlernmethoden sondern Adaptionsverfahren zum Einsatz.

4.2.2 Formulierung des Lernproblems

Für den weiteren Verlauf der Arbeit ist es notwendig, das allgemeine maschinelle Lernproblem für Bayes'sche Netze zu formalisieren und die zugehörige Notation einzuführen:

Definition 4.1 (Maschinelles Lernproblem für Bayes'sche Netze) *Gegeben eine Menge $\boldsymbol{D} = \{D_1, \ldots, D_s\}$ bestehend aus s Trainingsfällen D_i und ein Performanzmaß Q, finde ein Bayes'sches Netz $B = (G, \boldsymbol{\theta})$, d.h., finde eine Struktur G und eine assoziierte Menge von Tabellen bedingter Wahrscheinlichkeiten $\boldsymbol{\theta}$, die Q optimieren. Jeder Trainingsfall D_i besteht dabei aus Zuweisungen von Zuständen zu einer Teilmenge der Variablen von B.*[2]

Es folgen einige Konsequenzen, die sich aus Definition 4.1 ergeben, sowie wichtige Grundlagen, bevor im Anschluss die in dieser Arbeit relevanten Algorithmen detailliert beschrieben werden. Einen Überblick maschineller Lernverfahren für Bayes'sche Netze geben beispielsweise Heckerman (1995), Buntine (1996) und Heckerman (1998).

4.2.3 Frequentistischer vs. Bayes'scher Ansatz

Es existieren zwei alternative Ansätze der Interpretation des Wahrscheinlichkeitsbegriffs: auf der einen Seite der so genannte *frequentistische*, auf der anderen der *Bayes'sche* Ansatz. Obwohl in dieser Arbeit grundlegende Begriffe der Wahrscheinlichkeitstheorie als bekannt vorausgesetzt werden, wird die Unterscheidung dieser beiden Schulen innerhalb der Wahrscheinlichkeitstheorie wegen der großen Bedeutung im Rahmen der Lernverfahren an dieser Stelle kurz erläutert.

Der frequentistische Ansatz interpretiert den Wahrscheinlichkeitsbegriff als eine physikalische Eigenschaft der Domäne, die im Prinzip auf der Basis beliebig oft wiederholbarer, von einander unabhängiger Zufallsexperimente ermittelt werden kann. In diesem Zusammenhang wird oft von *objektiven* Wahrscheinlichkeiten gesprochen. Als Begründer dieser Schule gilt R. Fisher (Fisher, 1912, 1922).

Im *Bayes'schen* Ansatz, der auf Thomas Bayes (Bayes, 1763) zurückgeht, werden Wahrscheinlichkeiten als Maß für die Einschätzung einer Person (engl. *degree of belief*) hinsichtlich des Eintreffens eines bestimmten Ereignisses aufgefasst. Eine Wahrscheinlichkeit ist damit keine physikalische, objektive Eigenschaft mehr, sondern eine *subjektive*, von der Person abhängige, Größe. Dies schließt allerdings nicht aus, dass eine Person eine Wahrscheinlichkeit im Sinne der Frequentisten als subjektive Einschätzung übernimmt.

Entscheidender Vorteil des Bayes'schen Ansatzes ist, dass auch Ereignissen eine Wahrscheinlichkeit zugewiesen werden kann, für die es nicht möglich ist, wiederholte Zufallsexperimente durchzuführen. Beispielsweise kann man modellieren, dass es eine 20-prozentige Wahrscheinlichkeit dafür gibt, dass Deutschland bei der nächsten Fußball-Weltmeisterschaft den Titel erringt. Dies ist eine Festlegung, die aus der frequentistischen Sichtweise nicht möglich ist. Solche Situationen treten wie in Kapitel 3 diskutiert im Zusammenhang mit benutzeradaptiven Systemen häufig auf, was den bedeutenden Einfluss des Bayes'schen Ansatzes im Kontext der Benutzermodellierung begründet. Wegen der Objektivitätseigenschaft und der damit verbundenen Beweisbarkeit der Korrektheit hat die frequentistische Schule ihre Bedeutung in streng (natur-)wissenschaftlichen Domänen, wo diese Eigenschaften entscheidenden Charakter besitzen.

[2] Die in dieser Arbeit verwendeten Lernverfahren basieren auf mehreren Standardannahmen bezüglich der Eigenschaften der freien Parameter der zu erlernenden Bayes'schen Netze sowie der Trainingsdaten, die beispielsweise in (Geiger, Heckerman & Meek, 1996) aufgeführt werden. Sie werden bei der Forschung zum maschinellen Lernen Bayes'scher Netze immer als gültig vorausgesetzt und spielen für den Fokus der vorliegenden Arbeit keine zentrale Rolle. Deshalb wird auf eine ausführliche Diskussion verzichtet. Die beiden wichtigsten sind die *globale Unabhängigkeit der Parameter* und *Modularität der Parameter*.

Der auf der frequentistischen Schule basierende Lernansatz besteht in der Maximierung der *Likelihood der Daten*[3] unter Betrachtung der möglichen Modelle (*Maximum-Likelihood-Methode*). Die Likelihood ist die Wahrscheinlichkeit der Daten konditioniert auf das Modell, im Falle eines Bayes'schen Netzes $P(\boldsymbol{D} \mid B)$.

$$P(\boldsymbol{D} \mid B) = \prod_{l=1}^{s} P(D_l \mid B). \tag{4.1}$$

Mit der Maximierung der Likelihood wird die Modellierung der gemeinsame Wahrscheinlichkeitsverteilung optimiert. Das Resultat ist eine erlernte gemeinsame Wahrscheinlichkeitsverteilung, die die Daten als Ganzes möglichst optimal repräsentiert. Das bedeutet, dass bestimmte Teile des Modells für sich betrachtet suboptimal modelliert sein können, was aber im Zusammenspiel mit anderen Aspekten der Kodierung gemeinsam dennoch zur Optimalität des kompletten Modells führt.[4]

Aus Gründen der praktischen Handhabbarkeit wird häufig der Logarithmus der Likelihood (*Log-Likelihood*) betrachtet:

$$\ln P(\boldsymbol{D} \mid B) = \sum_{l=1}^{s} \ln P(D_l \mid B). \tag{4.2}$$

Die Lernaufgabe lässt sich damit wie folgt formulieren:[5]

$$B = \arg\max_{B} P(\boldsymbol{D} \mid B) = \arg\max_{B} \ln P(\boldsymbol{D} \mid B), \tag{4.3}$$

d.h., es soll das Modell im Lernprozess ermittelt werden, dem die größte Wahrscheinlichkeit zugeschrieben wird, die vorliegenden Daten erzeugt zu haben.

Der Bayes'sche Lernansatz im Besonderen als auch die Bayes'sche Schule der Wahrscheinlichkeitstheorie (inklusive Bayes'scher Netze) im Allgemeinen basieren auf dem folgenden Satz (hier formuliert in spezieller dem Kontext angepasster Version):

Satz 4.1 (Satz von Bayes)

$$P(B \mid \boldsymbol{D}) = \frac{P(\boldsymbol{D} \mid B)P(B)}{P(\boldsymbol{D})}. \tag{4.4}$$

Auf seiner Grundlage kann anhand einer *A-priori-Wahrscheinlichkeitsverteilung* der möglichen Modelle $P(B)$ in Kombination mit der Likelihood der Daten $P(\boldsymbol{D} \mid B)$ und der Wahrscheinlichkeit der Daten $P(\boldsymbol{D})$ die so genannte *A-posteriori-Wahrscheinlichkeitsverteilung* der Modelle $P(B \mid \boldsymbol{D})$ bestimmt werden. Die A-priori-Wahrscheinlichkeitsverteilung repräsentiert die subjektiven Wahrscheinlichkeiten, die eine Person den möglichen Modellen zuschreibt, solange sie keine empirischen Daten kennt. Ein wichtiger Unterschied zur frequentistischen Maximum-Likelihood-Methode besteht darin, dass nicht ein einziges (unter Berücksichtigung der Daten) wahrscheinlichstes Modell ermittelt wird, sondern eine Wahrscheinlichkeitsverteilung aller Modelle.

[3] Mangels einer in diesem Zusammenhang geeigneten deutschen Übersetzung des Begriffs der 'Likelihood' wird in dieser Arbeit der englische Ausdruck beibehalten.

[4] Es existieren Lernverfahren, die ein Bayes'sches Netz unter einer anderen Sichtweise lernen, beispielsweise um eine optimale Performanz bei bestimmten Typen von Anfragen an das Modell zu erzielen. Ein Beispiel hierfür ist der ELQ-Algorithmus von Greiner, Grove und Schuurmans (1997), der auch in Abschnitt 4.3.2.3 angesprochen wird.

[5] Da der Logarithmus monotonieerhaltend ist, kann er zur algorithmischen Vereinfachung der Maximierungsaufgabe verwendet werden.

Eine Approximation des in der Praxis wegen der Berechnung und Verwaltung der Wahrscheinlichkeits*verteilungen* über alle möglichen Netze oft schwierig zu handhabenden Bayes'schen Ansatzes stellt das *Maximum-a-posteriori*-Lernen (*MAP*-Lernen) dar. Anstatt der Ermittlung der A-posteriori-Wahrscheinlichkeitsverteilung wird lediglich das Modell bestimmt, das die größte A-posteriori-Wahrscheinlichkeit besitzt:

$$B = \arg\max_{B} P(\boldsymbol{D} \mid B)P(B). \tag{4.5}$$

Da die A-priori-Wahrscheinlichkeit der Daten $P(\boldsymbol{D})$ eine Konstante bezüglich aller potenziellen Modelle ist, kann sie bei der Maximierung vernachlässigt werden und es genügt, das Produkt aus A-priori-Wahrscheinlichkeit $P(B)$ und Likelihood $P(\boldsymbol{D} \mid B)$ zu maximieren. Insbesondere in Fällen, in denen die A-posteriori-Wahrscheinlichkeitsverteilung ein ausgeprägtes Maximum besitzt, ist die MAP-Methode als gute Approximation des Bayes'schen Lernansatzes anzusehen. Bei zunehmender Größe der verfügbaren Menge an Trainingsfällen konvergieren MAP- und Maximum-Likelihood-Methode gegeneinander, da der Einfluss der A-priori-Verteilung im MAP-Ansatz abnimmt.

4.2.4 Vier Lernsituationen

Aus Definition 4.1 ergeben sich *vier* Szenarien des maschinellen Lernens Bayes'scher Netze, die aufgrund ihrer unterschiedlichen Komplexität der Aufgabenstellung mit unterschiedlichen Methoden behandelt werden müssen (vgl. z.B. Russell & Norvig, 1995). Die verfügbaren Trainingsdaten können entweder *vollständig* oder *unvollständig* sein, und die Struktur des zu erlernenden Netzes kann entweder *bekannt* oder *unbekannt* und somit zu erlernen sein:

- *Bekannte Struktur, vollständige Trainingsdaten:* Diese Situation stellt das am einfachsten zu behandelnde Szenario dar. Es sind lediglich die bedingten Wahrscheinlichkeiten θ_{ijk} der CPTs $\boldsymbol{\theta}$ zu ermitteln. Da vollständige Trainingsfälle vorliegen, können die relativen Häufigkeiten der Zustandskombinationen der Eltern-Kind-Variablenpaare in der Datenmenge $\boldsymbol{D}$ ausgezählt werden, um im frequentistischen Ansatz Maximum-Likelihood-Schätzungen zu erhalten. Auch im Bayes'schen Ansatz existieren einfache Verfahren zur Berechnung der Posteriori-Werte der bedingten Wahrscheinlichkeiten.

- *Bekannte Struktur, unvollständige Trainingsdaten:* Dies ist der wohl am häufigsten in der Praxis—gerade auch in benutzeradaptiven Systemen—auftretende Fall: Die Struktur G wurde von Experten spezifiziert und es verbleibt das Lernen der bedingten Wahrscheinlichkeiten θ_{ijk} anhand unvollständiger Trainingsfälle $\boldsymbol{D}$. In dieser Situation muss in beiden Ansätzen—frequentistisch oder Bayes'sch—auf aufwendigere Verfahren zurückgegriffen werden. Die beiden bekanntesten und erfolgreichsten Methoden sind (a) die *Expectation-Maximization*-Methode (EM, Dempster, Laird & Rubin, 1977) und (b) der gradienten-basierte *Adaptive-Probabilistic-Networks*-Algorithmus (APN) von Binder, Koller, Russell und Kanazawa (1997).

- *Unbekannte Struktur, vollständige Trainingsdaten:* Die Aufgabe der Rekonstruktion der kausalen Struktur G der Domäne wird oft als hochdimensionales Suchproblem im Raum der möglichen Strukturen aufgefasst. Allgemeines Suchkriterium—unter Berücksichtigung des gewählten Ansatzes—ist die Fähigkeit der potenziellen Strukturen, die in den Daten

$\boldsymbol{D}$ vorhandenen (Un-)Abhängigkeiten zu modellieren. Dabei reduziert sich das Teilproblem des Lernens der CPTs $\boldsymbol{\theta}$ auf den ersten Fall mit gegebener Struktur und vollständigen Trainingsdaten. Hier werden oft lokale Suchverfahren wie Greedy-Hillclimbing-Verfahren eingesetzt.

- *Unbekannte Struktur, unvollständige Trainingsdaten:* Hierbei handelt es sich um den schwierigsten der vier Fälle, der die aufwendigsten Techniken erfordert. Mit dem *strukturellen EM*-Algorithmus von Friedman (1997, 1998) existiert ein praktikables, approximatives Verfahren, das eine Erweiterung des Standard-EM-Verfahrens zum Erlernen der bedingten Wahrscheinlichkeiten θ_{ijk} bei bekannter Struktur G und unvollständigen Daten $\boldsymbol{D}$ darstellt.

	Trainingsdaten	
Struktur	vollständig	unvollständig
bekannt	analytische Lösung	EM, APN
unbekannt	lokale Suche	Struktureller EM

***Tabelle 4.1**: Die vier Szenarien des maschinellen Lernens Bayes'scher Netze*

Tabelle 4.1 fasst die vier Lernszenarien und die entsprechenden (in dieser Arbeit) verwendeten Methoden, die im weiteren Verlauf dieses Kapitels detailliert vorgestellt werden, zusammen.

4.2.5 Verborgene Variablen

Ein Spezialfall unvollständiger Trainingsdaten tritt dann auf, wenn Variablen im Modell existieren, zu denen in *keinem* der Trainingsfälle $\boldsymbol{D}$ ein Wert existiert. Man spricht dann von *verborgenen Variablen*—im Gegensatz zu beobachteten Variablen.

Solche Variablen spielen insbesondere im Kontext benutzeradaptiver Systeme eine wichtige Rolle, da viele Benutzereigenschaften bzw. -interessen oft generell nicht empirisch beobachtet werden (können). In manchen Systemen wird diese Situation dadurch vermieden, dass beispielsweise eine explizite Angabe der Interessen durch den Benutzer gefordert wird. Ein typischer Fall einer verborgenen Variable, wie er bereits in Abschnitt 4.1.2 besprochen wurde, ist die Klassifikationsvariable beim unüberwachten Lernen eines naiven Bayes'schen Klassifizierers. Andere Beispiele solcher verborgenen Variablen sind die beiden Variablen TATSÄCHLICHE ARBEITSGEDÄCHTNISBELASTUNG und RELATIVE GESCHWINDIGKEIT DER SPRACHPRODUKTION der Bayes'schen Netze zur Modellierung der Versuchspersonen im Flughafenexperiment (Abbildung 2.7 (b)). Vorteile solcher verborgener Variablen sind u.a.:

- *Interpretierbarkeit:* Wie am Beispiel der Netze des Anweisungs- und Flughafenexperiments bereits verdeutlicht, dienen verborgene Variablen oftmals als erklärende Variablen, die die Interpretierbarkeit der Modelle erhöhen. Mit ihnen wird der potenzielle Nutzen des Einsatzes von Erklärungskomponenten verbessert, da das System damit in der Lage ist, Erläuterungen zu geben, wie etwa „Der Zeitdruck, unter dem die Versuchsperson steht, führt zu einer erhöhten tatsächlichen Arbeitsgedächtnisbelastung, was sich wiederum in kürzeren Äußerungen (weniger Silben) widerspiegelt.".

- *Repräsentation von Abhängigkeiten:* In manchen Situationen werden verborgene Variablen benötigt, um die bedingten (Un-)Abhängigkeiten, die in der Domäne vorliegen, korrekt abzubilden. Die Netzstruktur ohne verborgene Variablen aus Abbildung 2.7 (a) baut auf der

Annahme auf, dass alle Symptomvariablen bei bekannten Elternzuständen untereinander bedingt unabhängig sind. Die Struktur in Abbildung 2.7 (b) hingegen modelliert einen Tradeoff zwischen Geschwindigkeit und Qualität der Sprachproduktion, der komplexere Beziehungen zwischen den beobachtbaren Variablen impliziert. Ohne das Einbringen verborgener Variablen wäre ein solches komplexes Modell nur schwierig zu realisieren, insbesondere wenn die Interpretierbarkeit der Modelle gewährleistet sein soll. Friedman (1997) beschreibt eine Studie, in der das Einbringen verborgener Variablen in ein erlerntes Bayes'sches Netz aus diesem Grund zu einer Verbesserung der Modellierung der gemeinsamen Wahrscheinlichkeitsverteilung führt.

- *Kompaktheit:* Im Allgemeinen kann die Verwendung verborgener Variablen die Kompaktheit des Modells erhöhen (vgl. auch Russell, Binder, Koller & Kanazawa, 1995). Schon das relativ einfache, nur aus beobachteten Variablen bestehende Modell aus Abbildung 2.7 (a) benötigt eine Vielzahl von Kanten zwischen den unabhängigen und abhängigen Variablen. Würde das System weitere Symptomvariablen berücksichtigen (vgl. Müller, 2001; Kiefer, 2002), so wäre bald der Punkt erreicht, an dem die große Anzahl an Kanten das Modell sowohl aus theoretischer als auch aus praktischer Sicht unbrauchbar macht. Die Anzahl der benötigten bedingten Wahrscheinlichkeiten würde stark ansteigen, was bei gleichbleibender Menge an Trainingsdaten den Lernprozess schwieriger machen würde, da zum Erlernen pro Wahrscheinlichkeit weniger verwendbare Fälle zur Verfügung stehen.

- *Potenzielle Kombinationspunkte:* Eine verborgene Variable kann als Kombinationspunkt zweier getrennt erlernter Bayes'scher Netze dienen, beispielsweise eingebettet in einem objekt-orientiertem Ansatz wie in Abschnitt 2.5 beschrieben. Als Beispiel sei hier die Situation genannt, in der zwei unabhängige Lernprozesse durchgeführt wurden, die jeweils ein Netz zur Modellierung der Abhängigkeiten zwischen der Arbeitsgedächtnisbelastung und unterschiedlichen Sprachsymptomen geliefert haben. Dann ist es im Allgemeinen möglich, ein kombiniertes Netz zu erstellen, das eine (einzige) Variable zur Repräsentation der Arbeitsgedächtnisbelastung sowie die Vereinigungsmenge aller Symptomvariablen enthält.

Diese Vorteile werden durch eine erhöhte Komplexität der notwendigen Lernverfahren erkauft, so dass zwischen dem Mehrwert eines Modells mit verborgenen Variablen und den Laufzeitanforderungen des Systems abgewägt werden muss.

Im Zusammenhang mit verborgenen Variablen in Bayes'schen Netzen ist zu bemerken, dass die Bezeichnung einer solchen Variablen (lediglich) eine semantische Interpretation unter Berücksichtigung der Zusammenhänge mit den Eltern- und Kindvariablen in der Modellierung darstellt. Die Anwendung maschineller Lernverfahren führt nicht direkt dazu, dass etwas über die entsprechende Größe der Realität gelernt wird. Es ist nicht möglich, den Lernverfahren eine spezifische semantische Interpretation einer verborgenen Variablen vorzugeben, die im Lernvorgang berücksichtigt wird. Die Algorithmen erkennen lediglich das Vorhandensein einer verborgenen Variablen und Lernen anhand der Zusammenhänge mit den in den Trainingsdaten beobachteten Eltern- und Kindvariablen. So ist es im Beispiel der verborgenen Variable TATSÄCHLICHE ARBEITSGEDÄCHTNISBELASTUNG für Lernalgorithmen nicht zu erfassen, ob diese semantische Interpretation oder die Interpretation der Variable als FREIE TATSÄCHLICHE ARBEITSGEDÄCHTNISKAPAZITÄT intendiert ist. Noch grundlegender ist hier die Verbindung der Variable mit dem Konzept des Arbeitsgedächtnises. Durch die Anwendung maschineller Lernverfahren wird aus psychologischer Sichtweise im Allgemeinen wenig über das Arbeitsgedächtnis als solches gelernt. Erklären-

de verborgene Variablen sind im Wesentlichen ein Hilfsmittel, deren Verwendung aufgrund der angeführten Vorteile in benutzeradaptiven Systemen sinnvoll erscheint, die aber nur zu einem gewissen Grad in Relation zur modellierten Größe der Realität stehen.

4.3 Lernen der bedingten Wahrscheinlichkeiten

Das Lernen der bedingten Wahrscheinlichkeiten θ_{ijk} der CPTs $\boldsymbol{\theta}$ ist eine zentrale Aufgabenstellung im Rahmen des Lernproblems eines Bayes'schen Netzes $B = (G, \boldsymbol{\theta})$, da es in jeder der in Abschnitt 4.2.4 diskutierten Lernsituationen bearbeitet werden muss.

4.3.1 Vollständige Trainingsdaten

Die einfachste aller Lernsituationen tritt bei vollständigen Trainingsdaten $\boldsymbol{D}$ und fester, bereits spezifizierter Struktur G ein. In diesem Fall können die bedingten Wahrscheinlichkeiten θ_{ijk} von B *lokal*, d.h., für jede der Eltern-Kind-Variablenmengen separat, ermittelt werden. Dies folgt aus dem im Zusammenhang mit der Definition Bayes'scher Netze (Definition 2.1) vorgestellten Unabhängigkeitskriterium. Die Likelihood der Trainingsdaten $P(\boldsymbol{D} \mid B)$ aus Gleichung 4.3 wird durch die in den Daten auftretenden relativen Häufigkeiten der Variablenzustandskombinationen maximiert:

$$\hat{\theta}_{ijk} = \frac{N_{ijk}}{N_{ik}}, \tag{4.6}$$

dabei sind die N_{ijk} die in den empirischen Daten $\boldsymbol{D}$ auftretenden Häufigkeiten der den Indizes i, j, k entsprechenden Zustandskombinationen und $N_{ik} = \sum_{j=1}^{n_i} N_{ijk}$ die Anzahl der den Indizes i und j entsprechenden Kombinationen von Zuständen.

Der Bayes'sche Ansatz basiert in dieser Situation auf der Tatsache, dass es sich bei den durch Bayes'sche Netze mit diskreten Variablen modellierten gemeinsamen Wahrscheinlichkeitsverteilungen um *multinomiale* Verteilungen handelt (siehe z.B. Heckerman, 1995). Essentiell für die Handhabbarkeit der Berechnung der A-posteriori-Verteilung für $\boldsymbol{\theta}$ mit Hilfe des Satzes von Bayes aus A-priori-Wahrscheinlichkeitsverteilung $P(\boldsymbol{\theta})$ und Likelihood $P(\boldsymbol{D} \mid \boldsymbol{\theta})$ ist die Verwendung *konjugierter* Dichtefunktionen zur Modellierung der Wahrscheinlichkeitsverteilungen. Die entscheidende Eigenschaft solcher konjugierter Funktionen ist die Tatsache, dass die berechnete A-posteriori-Verteilung wiederum dieser Funktionenfamilie angehört (siehe z.B. DeGroot, 1970). Im vorliegenden Fall kommt die *Dirichlet-Verteilung* zum Einsatz. Für jede der Zustandskombination $pa_k(X_i)$ der Eltern einer Variablen X_i wird eine n_i-dimensionale Dirichlet-Verteilung $Dir(\alpha_1^{ik}, \ldots, \alpha_{n_i}^{ik})$ mit *Hyperparametern* $\alpha_1^{ik}, \ldots, \alpha_{n_i}^{ik}$ verwendet, um die zugehörigen bedingten Wahrscheinlichkeiten $\theta_{i1k}, \ldots, \theta_{1n_ik}$ einzuschätzen. Es gilt:

$$\theta_{ijk} = \frac{\alpha_j^{ik}}{\sum_{l=1}^{n_i} \alpha_l^{ik}}. \tag{4.7}$$

Abbildung 4.3 zeigt Beispiele 2-dimensionaler Dirichlet-Verteilungen,[6] die zur Modellierung der CPTs binärer Variablen eingesetzt werden. Die linke Spalte repräsentiert eine Situation, die durch eine höhere Unsicherheit bezüglich der A-priori-Einschätzung der Wahrscheinlichkeit gekennzeichnet ist (oberer Graph). Dies spiegelt sich in einer breiten Glockenform der Kurve wider.

[6] 2-dimensionale Dirichlet-Verteilungen werden auch als *Beta-Verteilungen* bezeichnet.

Es wird eine bedingte Wahrscheinlichkeit von 0.6 für das Eintreffen des repräsentierten Ereignisses angenommen. Analog zeigt die rechte Spalte eine Situation, in der eine geringere Unsicherheit in der A-priori-Verteilung modelliert wird, gekennzeichnet durch die schmale Glockenform der Kurve. Die beiden unteren Graphen stellen die A-posteriori-Verteilungen nach Berücksichtigung von je drei Trainingsfällen dar, in denen das betrachtete Ereignis eintraf. Man kann erkennen, dass einerseits in beiden Fällen der geschätzte Wert der bedingten Wahrscheinlichkeit erhöht wird, andererseits die Unsicherheit hinsichtlich der Einschätzung verringert wird (schmälere Glockenform). Man sieht, dass die unsichere A-priori-Einschätzung in einer stärkeren Anpassung der Wahrscheinlichkeit resultiert. Den empirischen Daten kommt in diesem Fall also eine höhere Bedeutung zu.

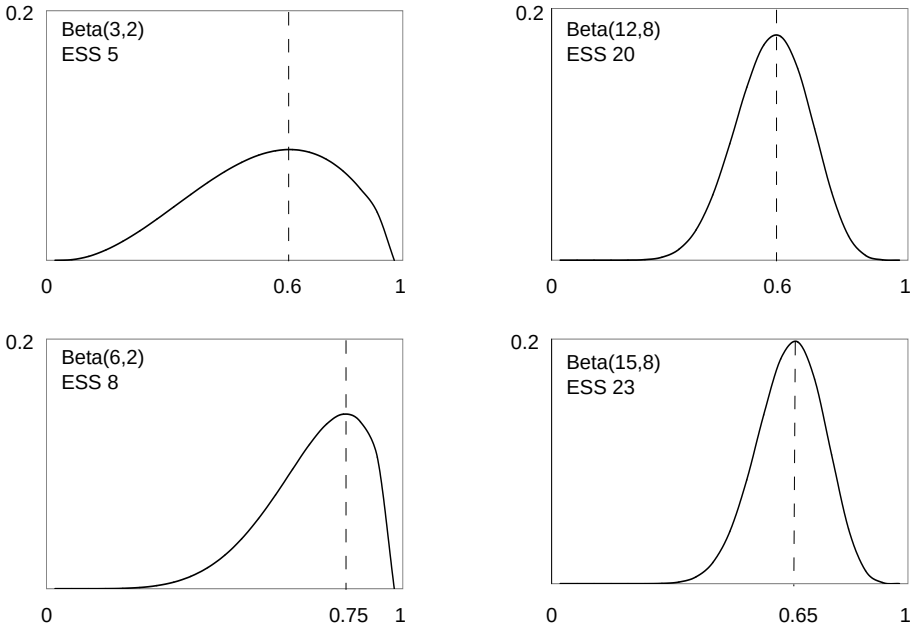


***Abbildung 4.3**: Beispiel für das Bayes'sche Lernen der bedingten Wahrscheinlichkeiten mit Dirichlet-Verteilungen*
(Erläuterungen im Text)

Das vorhandene Expertenwissen zu den bedingten Wahrscheinlichkeiten kann mit Dirichlet-Verteilungen somit durch die Angabe der Hyperparameter kodiert werden. Die Konfidenz einer Experteneinschätzung drückt sich in der Glockenform der Kurve aus: breite Kurven stellen unsichere Einschätzungen dar, schmale repräsentieren eine höhere Konfidenz.

Dirichlet-Verteilungen werden eingesetzt, da es bei ihnen sehr einfach ist aus der A-priori-Verteilung anhand der Daten $\boldsymbol{D}$ die A-posteriori-Verteilung zu berechnen. Es müssen lediglich die in den empirischen Daten auftretenden Häufigkeiten N_{ijk} zu den entsprechenden Hyperparametern der A-priori-Verteilung addiert werden:

$$P(\boldsymbol{\theta}_{ik} \mid \boldsymbol{D}) = Dir(\alpha_1^{ik} + N_{i1k}, \ldots, \alpha_{n_i}^{ik} + N_{in_ik}). \tag{4.8}$$

Der MAP-Schätzwert $\tilde{\theta}_{ijk}$ ergibt sich dann mit Gleichung 4.7 als:

$$\tilde{\theta}_{ijk} = \frac{\alpha_j^{ik} + N_{ijk}}{\sum_{l=1}^{n_i} \alpha_l^{ik} + N_{ik}}. \tag{4.9}$$

In den obigen Formeln können die Hyperparameter $\alpha_1^{ik}, \ldots, \alpha_{n_i}^{ik}$ der A-priori-Dirichlet-Wahrscheinlichkeitsverteilung als virtuelle Häufigkeiten der Zustandkombinationen interpretiert werden, weshalb ihre Summe

$$s_{ik} = \sum_{l=1}^{n_i} \alpha_l^{ik} \tag{4.10}$$

auch als *äquivalente Stichprobengröße* (engl. *equivalent sample size, ESS*) bezeichnet wird. Sie kann als Konfidenzmaß für die spezifizierte A-priori-Verteilung angesehen werden: Je höher der ESS-Wert, desto ausgeprägter ist das Maximum der Dirichlet-Verteilung (bei gleichem Verhältnis der Hyperparameter), d.h., desto höher ist die Sicherheit bezüglich der a priori spezifizierten Wahrscheinlichkeitsverteilung (vgl. Abbildung 4.3). Dies erscheint intuitiv plausibel, da Werte, die auf der Grundlage einer größeren Anzahl von Beobachtungen spezifiziert wurden, in der Regel verlässlicher sind als solche, die nur auf einer schwachen empirischen Basis fußen.

4.3.2 Unvollständige Trainingsdaten

Liegen dagegen keine vollständigen Trainingsfälle $\boldsymbol{D}$ vor, so ist es nicht möglich, die bedingten Wahrscheinlichkeiten θ_{ijk}, die die Bewertungsfunktionen maximieren, lokal und in geschlossener Form zu bestimmen (siehe z.B. Heckerman, 1995). In diesem Fall muss auf approximative (Such-)Verfahren höheren Komplexitätsgrades zurückgegriffen werden. Die beiden bekanntesten, in der Praxis am häufigsten mit Erfolg eingesetzten und in dieser Arbeit verwendeten Verfahren werden im Anschluss vorgestellt.

4.3.2.1 Expectation-Maximization

Der *Expectation-Maximization*- oder kurz *EM*-Algorithmus (Dempster et al., 1977) geht angewendet auf das Problem der Maximierung der (Log-)Likelihood folgendermaßen vor: Nachdem die θ_{ijk}-Werte mit Startwerten initialisiert worden sind, führt der Algorithmus je zwei Schritte iterativ durch: Im ersten Schritt, dem *Expectation*- oder *E*-Schritt, werden für die in den Trainingsfällen D_i fehlenden Werte die Erwartungswerte der relativen Häufigkeiten N_{ijk} ermittelt. (Diese Berechnung beinhaltet für jeden Trainingsfall D_i das Berechnen der Wahrscheinlichkeit des zu ermittelnden Wertes konditioniert auf die bekannten Werte in D_i, was mit Hilfe der Inferenzverfahren für Bayes'sche Netze vollzogen werden kann.) Das Resultat des E-Schrittes ist eine hypothetische Trainingsmenge $\boldsymbol{D}'$, die zusätzlich zu den beobachteten Werten Erwartungswerte der fehlenden Daten enthält:

$$E_{\boldsymbol{\theta}}[N_{ijk}] = \sum_{l=1}^{s} P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}). \tag{4.11}$$

Der zweite Schritt, der *Maximization*- oder kurz *M*-Schritt, bestimmt die neuen bedingten Wahrscheinlichkeiten der CPTs $\boldsymbol{\theta}'$, die die (Log-)Likelihood der hypothetischen Trainingsmenge $\boldsymbol{D}'$ (lokal) maximieren—eine Aufgabe, die wesentlich einfacher ist, als die Maximierung der (Log-)Likelihood des „echten" Datensatzes $\boldsymbol{D}$. Diese neuen θ'_{ijk}-Werte liefern immer eine (Log)-Likelihood der „echten" Trainingsmenge, die mindestens so hoch ist wie die der vorhergehenden Werte θ_{ijk}:

$$\theta'_{ijk} = \frac{E_{\boldsymbol{\theta}}[N_{ijk}]}{E_{\boldsymbol{\theta}}[N_{ik}]}. \tag{4.12}$$

Die beiden Schritte werden alternierend durchgeführt bis der Algorithmus gegen ein (lokales) Optimum der (Log-)Likelihood-Funktion konvergiert. Die so berechneten bedingten Wahrscheinlichkeiten $\hat{\theta}_{ijk}$ entsprechen Maximum-Likelihood-Schätzungen.

Die Bestimmung der MAP-Werte $\tilde{\theta}_{ijk}$ erfolgt analog durch folgende Modifikation innerhalb des M-Schritts:

$$\theta'_{ijk} = \frac{E_{\boldsymbol{\theta}}[N_{ijk}] + \alpha_j^{ik}}{E_{\boldsymbol{\theta}}[N_{ik}] + \sum_{l=1}^{n_i} \alpha_l^{ik}}. \tag{4.13}$$

Die Komplexität des EM-Algorithmuses wird dominiert durch die Komplexität der im E-Schritt wiederholt angewendeten—möglicherweise approximativen—Inferenzverfahren zur Ermittlung des Erwartungswerts. Die Konvergenz des Verfahrens wurde von Dempster et al. (1977) gezeigt.

4.3.2.2 Adaptive-Probabilistic-Networks

Eine Alternative zum EM-Algorithmus stellt die *Adaptive-Probabilistic-Networks*- oder *APN*-Methode dar (Russell et al., 1995; Binder et al., 1997). Dabei handelt es sich um einen gradientenbasierten Ansatz in Form eines Hillclimbing-Suchverfahrens.

Die Berechnung der neuen $\boldsymbol{\theta}'$-Werte wird durch die Durchführung (kleiner) Schritte in der Richtung des ermittelten Gradienten $\boldsymbol{\nabla} \ln P(\boldsymbol{D} \mid \boldsymbol{\theta})$ der Log-Likelihood bewerkstelligt:

$$\boldsymbol{\theta}' = \boldsymbol{\theta} + \alpha \boldsymbol{\nabla} \ln P(\boldsymbol{D} \mid \boldsymbol{\theta}), \tag{4.14}$$

wobei α die Schrittweite spezifiziert.

Die partiellen Ableitungen des Gradienten werden nach Russell et al. (1995) (wiederum unter Anwendung der Inferenzverfahren Bayes'scher Netze) wie folgt berechnet:

$$\nabla_{ijk}^{u} \ln P(\boldsymbol{D} \mid \boldsymbol{\theta}) = \sum_{l=1}^{s} \frac{P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta})}{\theta_{ijk}}. \tag{4.15}$$

Dabei gibt das hochgestellte ^{u} an, dass es sich hierbei noch um den unprojizierten Gradienten handelt, d.h., dieser muss noch auf die durch den Constraint $\sum_j \theta'_{ijk} = 1$ definierte Oberfläche projiziert werden, so dass auch die neuen bedingten Wahrscheinlichkeiten dieser fundamentalen Anforderung der Wahrscheinlichkeitstheorie genügen. Die Projektion wird von Binder et al. (1997) in Abschnitt 5.3 beschrieben.

Das Ergebnis nach der Konvergenz des Verfahrens ist ebenfalls—wie beim EM-Algorithmus—ein lokales Maximum der (Log-)Likelihood. Voraussetzung der Konvergenz ist—wie üblich bei gradienten-basierten Methoden—eine adäquate Wahl der Schrittweite.

Diese Grundform des Algorithmus kann durch bekannte, aufwendigere gradienten-basierte Verfahren wie z.B. das Verfahren der konjugierten Gradienten (siehe z.B. Press, 1992), das automatisch gute Schrittweiten bestimmt, optimiert werden.

Auch hier gilt wie beim EM-Algorithmus, dass die Komplexität im Wesentlichen von den im Rahmen der Berechnung des Gradienten verwendeten Inferenzverfahren dominiert wird.

4.3.2.3 Weitere Verfahren

Die beiden in den vorigen Abschnitten vorgestellten Standardalgorithmen erlernen die bedingten Wahrscheinlichkeiten Bayes'scher Netze, derart, dass die modellierte gemeinsame Wahrschein-

lichkeitsverteilung möglichst optimal zum vorhandenen Datensatz sowie gegebenenfalls den A-priori-Verteilungen „passt“. Dies ist eine wünschenswerte Eigenschaft solcher Netze, die flexibel eingesetzt werden, d.h., die in der Lage sein müssen, unterschiedlichste Anfragen zu beantworten. Für Einsatzszenarien, die durch weitestgehend gleichbleibende Anfragen charakterisiert sind, d.h., in denen beispielsweise immer die Wahrscheinlichkeiten der gleichen Variablen von Interesse sind (z.B. beim naiven Bayes'schen Klassifizierer) wurden spezialisierte Verfahren entwickelt. Solche Algorithmen zeichnen sich im Allgemeinen dadurch aus, dass die für den Suchprozess verwendete Bewertungsfunktion—die Likelihood der Daten in den Standardverfahren—eine Optimierung der Performanz der erlernten Netze bezüglich der tatsächlich auftretenden Anfragen bewirkt. Im Falle des naiven Bayes'schen Klassifizierers bedeutet dies beispielsweise, dass der Inferenzprozess zur Berechnung der Wahrscheinlichkeitsverteilung der Klassenzugehörigkeit anhand beobachteter Merkmalsvariablen optimiert wird. Ein entsprechendes Verfahren stellen z.B. Friedman, Geiger und Goldszmidt (1997) vor. Eine Methode zum Erlernen der bedingten Wahrscheinlichkeiten eines (strukturell beliebigen) Bayes'schen Netzes unter Berücksichtigung der relativen Häufigkeiten des Auftretens verschiedener Anfragen entwickelten Greiner et al. (1997) mit dem ELQ-Algorithmus. Diese Verfahren können sowohl bei vollständigen als auch bei unvollständigen Trainingsdaten eingesetzt werden.

4.4 Lernen der Struktur

Das Strukturlernproblem Bayes'scher Netze umfasst das im vorangehenden Abschnitt besprochene Lernen der bedingten Wahrscheinlichkeiten als Teilproblem. Das Ermitteln der kausalen Zusammenhänge einer Domäne als eine Form der Wissensentdeckung spielt häufig in einer frühen Phase der Systemkonstruktion eine bedeutende Rolle, um generelle Zusammenhänge von Interesse zu identifizieren. In vielen Anwendungsszenarien genügt es, sich im Anschluss auf das Erlernen bzw. das Verwalten der bedingten Wahrscheinlichkeiten zu konzentrieren (vgl. Abschnitt 2.6).

Beim Strukturlernen existieren mit den *testbasierten* und den *metrikbasierten* Verfahren zwei unterschiedliche prinzipielle Herangehensweisen. Eine umfassende vergleichende Diskussion der beiden Ansätze sowie der einzelnen Verfahren bieten Cheng, Greiner, Kelly, Bell und Liu (2002).

4.4.1 Testbasierte Verfahren

Testbasierte Verfahren zum Erlernen der Struktur Bayes'scher Netze (siehe beispielsweise Spirtes, Glymour & Scheines, 1990; Fung & Crawford, 1990; Spirtes, Glymour & Scheines, 1991; Steck, 2000; Cheng et al., 2002) versuchen anhand der verfügbaren Trainingsdaten unter Anwendung statistischer Tests sowie des d-Separationskriteriums, die lokalen bedingten Abhängigkeiten bzw. Unabhängigkeiten zwischen den Variablen einer Domäne zu identifizieren. Verschiedene Arbeiten (siehe z.B. Heckerman, Geiger & Chickering, 1995) zeigen, dass testbasierte Verfahren oft gegenüber metrikbasierten Ansätzen qualitativ zurückstehen. Dies ist insbesondere in solchen Situationen der Fall, die durch wenige und/oder verrauschte Trainingsdaten gekennzeichnet sind. Ein Vorteil testbasierter Algorithmen besteht in der höheren Effizienz bei einer großen Anzahl betrachteter Variablen. Aus den im nächsten Abschnitt erläuterten Gründen eignen sich metrikbasierte Verfahren besser für den Einsatz in benutzeradaptiven Systemen, weshalb für detailliertere Angaben zu testbasierten Verfahren auf die angeführte Literatur verwiesen wird.

4.4.2 Metrikbasierte Verfahren

Metrikbasierte Methoden (siehe z.B. Chow & Liu, 1968; Cooper & Herskovits, 1992; Lam & Bacchus, 1993; Suzuki, 1993) optimieren eine Bewertungsfunktion, die beschreibt, inwieweit die vorhandenen Daten durch das betrachtete Netz adäquat modelliert werden.

Wegen des hochdimensionalen Suchraums der Strukturen[7] muss beim Strukturlernproblem Bayes'scher Netze auf heuristische Suchverfahren wie Hillclimbing- oder Simulated-Annealing-Methoden zurückgegriffen werden. Bouckaert (1995) und Chickering, Geiger und Heckerman (1994) zeigen, dass das allgemeine Strukturlernproblem Bayes'scher Netze NP-hart ist. Dennoch hat sich gezeigt, dass entsprechende approximative Verfahren in realistischen Anwendungsszenarien durchaus brauchbare Resultate liefern (siehe z.B. Heckerman et al., 2000; Nicholson et al., 2001).

Die (Log-)Likelihood der Daten (Gleichungen 4.1 und 4.2) kann in diesem Fall nicht als Bewertungsfunktion verwendet werden, denn sie wird durch die *vollverbundene Struktur*, d.h., die Struktur, die alle möglichen Kanten beinhaltet,[8] maximiert: Eine vollverbundene Struktur widerspricht dem Effizienzgedanken. Sie besitzt die größtmögliche Anzahl freier Parameter (die bedingten Wahrscheinlichkeiten θ_{ijk} der CPTs $\boldsymbol{\theta}$) und ist somit zwar prinzipiell in der Lage, die Daten $\boldsymbol{D}$ optimal zu modellieren, allerdings ist dies einerseits in den meisten praktisch relevanten Anwendungsszenarien mit Overfitting verbunden und andererseits sind vollverbundene Strukturen sowohl im Rahmen der Lernverfahren als auch hinsichtlich der Inferenzverfahren in realistischen Domänen nicht praktikabel handhabbar.

Aus diesen Gründen werden im frequentistischen Ansatz die Bewertungsfunktionen üblicherweise durch Erweiterung der (Log-)Likelihood um einen zusätzlichen Term konstruiert, der in irgendeiner Form die Komplexität der erlernten Strukturen berücksichtigt. Modelle, die mehr Kanten enthalten, werden somit im Suchprozess schlechter bewertet als weniger komplexe. Dies entspricht der in *Occam's Razor* (siehe z.B. Mitchell, 1997, S. 65) formulierten Idee, dass einfachere Modelle bevorzugt werden sollten. Eine der Begründungen dieser These ist die Tatsache, dass beim Erlernen komplexer Modelle das Overfitting-Problem eine erhöhte Bedeutung erhält und damit die Generalisierungsfähigkeit vermindert wird.

In dieser Arbeit wird das *Bayesian Information Criterion (BIC)*[9] (Schwarz, 1978; Heckerman, 1995) als eine solche Bewertungsfunktion, die einfachere Modelle gegenüber komplexeren bevorzugt, verwendet:

$$\begin{aligned} BIC(G, \boldsymbol{D}) &= \ln P(\boldsymbol{D} \mid G, \widehat{\boldsymbol{\theta}}) - \frac{d}{2} \ln \mid \boldsymbol{D} \mid \\ &\approx \ln P(\boldsymbol{D} \mid G), \end{aligned} \tag{4.16}$$

wobei $\widehat{\boldsymbol{\theta}}$ die Maximum-Likelihood-Schätzungen der $\boldsymbol{\theta}$ repräsentieren und d ein Maß der Komplexität von B ist (im Wesentlichen die Anzahl der notwendigen bedingten Wahrscheinlichkeiten θ_{ijk}, d.h. die Anzahl der bedingten Wahrscheinlichkeiten, die unter Berücksichtigung des Constraints $\sum_j \theta_{ijk} = 1$ zur vollständigen Spezifikation der CPTs benötigt wird.). Der erste Term

[7] Die Anzahl der möglichen Strukturen $S(n)$ eines Netzes mit n Variablen kann nach Robinson (1977) gemäß der folgenden rekursiven Formel berechnet werden: $S(0) = 1, S(n) = \sum_{i=1}^{n} (-1)^{i+1} \frac{n!}{(n-i)!i!} 2^{i(n-i)} S(n-i)$.

[8] Eine vollverbundene Struktur besitzt eine Kante zwischen allen möglichen Variablenpaaren, derart, dass kein Zyklus existiert.

[9] Das BIC ist äquivalent zur Bewertung nach dem *Minimum-Description-Length-Prinzip* der Informationstheorie (siehe z.B. (Lam & Bacchus, 1993)).

des BIC ist die Log-Likelihood der Daten bei gegebener Struktur G in Kombination mit den Maximum-Likelihood-Schätzungen $\widehat{\boldsymbol{\theta}}$.

Ein weiterer Vorteil neben der einfachen Berechnung des BICs ist die Feststellung, dass es eine Approximation des Logarithmuses der *marginalen Likelihood* $P(\boldsymbol{D} \mid G)$ darstellt—ohne die Notwendigkeit der Spezifikation einer A-priori-Wahrscheinlichkeitsverteilung der bedingten Wahrscheinlichkeiten θ_{ijk} wie sie eigentlich im Zusammenhang mit einer Bayes'schen Bewertungsfunktion notwendig ist. Dies gilt insbesondere auch im Falle unvollständiger Trainingsdaten $\boldsymbol{D}$, wo konzeptionell eine nicht in geschlossener Form zu lösende Integration über alle möglichen Werte der freien Parameter $\boldsymbol{\theta}$ im Zusammenhang mit der betrachteten Struktur G zur Berechnung von $P(\boldsymbol{D} \mid G)$ notwendig ist:

$$P(\boldsymbol{D} \mid G) = \int P(\boldsymbol{D} \mid G, \boldsymbol{\theta}) P(\boldsymbol{\theta} \mid G) d\boldsymbol{\theta}. \tag{4.17}$$

Diese Integration wird durch die Approximation mit dem BIC durch die Verwendung der Maximum-Likelihood-Schätzungen $\widehat{\boldsymbol{\theta}}$ umgangen.

Damit kann das BIC auch im Bayes'schen Ansatz zusammen mit einer A-priori-Wahrscheinlichkeitsverteilung über den Strukturen zur Bestimmung der MAP-Lösung des Strukturlernproblems im Falle unvollständiger Trainingsdaten verwendet werden,[10] da gemäß dem Satz von Bayes (Satz 4.1) gilt:

$$P(G \mid \boldsymbol{D}) = \frac{P(\boldsymbol{D} \mid G) P(G)}{P(\boldsymbol{D})}. \tag{4.18}$$

Im Gegensatz dazu kann bei vollständigen Trainingsdaten unter Verwendung einer A-priori-Wahrscheinlichkeitsverteilung in Form einer Dirichlet-Verteilung das Integral der marginalen Likelihood $P(\boldsymbol{D} \mid G)$ und damit die A-posteriori-Wahrscheinlichkeit $P(G \mid \boldsymbol{D})$ der erlernten Netzstruktur in geschlossener Form bestimmt werden (Cooper & Herskovits, 1992):

$$P(G \mid \boldsymbol{D}) = \frac{P(G)}{P(\boldsymbol{D})} \prod_{i=1}^{n} \prod_{k=1}^{|\boldsymbol{pa}(X_i)|} \frac{\Gamma(s_{ik})}{\Gamma(s_{ik} + N_{ik})} \prod_{j=1}^{n_i} \frac{\Gamma(\alpha_j^{ik} + N_{ijk})}{\Gamma(\alpha_j^{ik})} \tag{4.19}$$

unter Ausnutzung der folgenden Eigenschaft der Gamma-Funktion: $\Gamma(x+1) = x\Gamma(x), \Gamma(1) = 1$. Diese Bewertungsfunktion wird als *BD-Metrik* (Bayes'sche Metrik mit Dirichlet-Prior) bezeichnet. Die Normalisierungskonstante $P(\boldsymbol{D})$ spielt für die Optimierungsaufgabe keine Rolle.

Nachteil der BD-Metrik ist die Notwendigkeit der Spezifikation der α_j^{ik}-Werte im Rahmen der Vorgabe der A-priori-Wahrscheinlichkeitsverteilung der bedingten Wahrscheinlichkeiten θ_{ijk}. Deshalb entwickelten Heckerman, Geiger und Chickering (1994) mit der *BDe-Metrik*[11] eine handhabbare Alternative. Die benötigten α_j^{ik}-Werte werden dabei anhand eines einzigen vorzugebenden Bayes'schen Netzes B_p, dass das A-priori-Wissen kodiert, und einer globalen ESS s folgendermaßen bestimmt:

$$\alpha_j^{ik} = s \cdot P(x_{ij}, pa_k(X_i) \mid B_p), \tag{4.20}$$

[10] Zu beachten ist, dass das BIC beim Auftreten verborgener Variablen keine adäquate Wahl zur Approximation der marginalen (Log-)Likelihood sein kann (Geiger, Heckerman, King & Meek, 1998).

[11] Das zusätzliche 'e' steht für 'equivalence' in der englischen Bezeichnung der *Likelihood-Äquivalenz*, welche die Eigenschaft der Bewertungsfunktion beschreibt, dass unterschiedliche Strukturen gleich bewertet werden. Beispiel hierfür ist das einfache Netz, das aus einer Kante zwischen zwei Variablen besteht. In dieser Situation spielt es keine Rolle, welche Richtung die Kante besitzt. Erst wenn die kausale Interpretation der Kanten berücksichtigt wird, kann diesbezüglich eine Entscheidung getroffen werden.

d.h., die lokalen von Gleichung 4.19 benötigten α_j^{ik}-Werte werden anteilig gemäß der Wahrscheinlichkeit des Eintretens der zugehörigen Zustandskombination durch Inferenz mittels B_p ermittelt.

Die A-priori-Wahrscheinlichkeitsverteilung der Strukturen wird durch folgende Funktion spezifiziert:

$$P(G) = c \cdot \kappa^{\delta}, \tag{4.21}$$

wobei $0 < \kappa \leq 1$ und δ die Summe der unterschiedlichen Elternanzahlen der Knoten im Vergleich von B und B_p angibt. c repräsentiert die benötigte Normalisierungskonstante. Mit dieser Festlegung werden komplexere Strukturen—ähnlich wie beim BIC—im Verlauf des Lernvorgangs schlechter bewertet als solche, die weniger Kanten besitzen.

Generell existieren beim Strukturlernproblem Bayes'scher Netze zwei Alternativen, die bei der Verwendung einer Bayes'schen Bewertungsfunktion in gewisser Weise dem MAP- bzw. dem vollen Bayes'schen Vorgehen entsprechen bzw. eine Approximation darstellen: (a) die *Modellselektion* bei der ein einziges (das wahrscheinlichste) Netz erlernt wird, das im Systembetrieb eingesetzt wird, und (b) das *Model-Averaging*, das auf einer Menge verschiedener erlernter Netze arbeitet, die im Rahmen der Inferenz gemäß ihrer A-posteriori-Wahrscheinlichkeiten gewichtet werden. Grundlage dabei ist die (nicht immer erfüllte) Annahme, dass diese Menge eine repräsentative Stichprobe der möglichen Bayes'schen Netze darstellen. Diese Arbeit beschränkt sich auf die Lösung der Problemstellungen im Zusammenhang mit Fall (a), viele Ergebnisse lassen sich allerdings direkt auf Situation (b) übertragen, indem die für einzelne Bayes'sche Netze entwickelten Verfahren auf jedes der in der Menge der betrachteten Netze separat angewendet wird.

4.4.3 Struktureller EM-Algorithmus

Im Folgenden wird der *strukturelle EM-Algorithmus* (*SEM*, Friedman, 1997, 1998) zur Behandlung der Strukturlernaufgabe Bayes'scher Netze benutzt. Er stellt eine Erweiterung des EM-Algorithmuses zum Erlernen der bedingten Wahrscheinlichkeiten auf den Strukturfall bei unvollständigen Daten dar. Auch hier werden die in den Trainingsdaten fehlenden Einträge durch den zugehörigen Erwartungswert der relativen Häufigkeiten ersetzt, um anschließend aufgrund dieses hypothetischen Trainingsdatensatzes das Erlernen der Struktur anzugehen. Die Grundidee des Verfahrens basiert—zusätzlich zum E- und M-Schritt—auf dem Alternieren zwischen Schritten zur Verbesserung (a) der Struktur und (b) der assoziierten bedingten Wahrscheinlichkeiten.

Es existieren für beide Lernansätze entsprechende Varianten des Algorithmuses: Im frequentistischen Ansatz wird als Bewertungsfunktion das BIC verwendet, wohingegen im Bayes'schen Ansatz (MAP) bei vollständigen Trainingsdaten die BDe-Bewertungsfunktion maximiert wird. Unvollständige Trainingsdaten können mit dem BIC in Kombination mit einer A-priori-Wahrscheinlichkeitsverteilung über den Strukturen, wie sie in Gleichung 4.21 spezifiziert wurde, bearbeitet werden.

Abbildung 4.4 zeigt das Gerüst des strukturellen EM-Algorithmuses, wobei Q die verwendete Bewertungsfunktion repräsentiert. Der Teil des Algorithmuses, der die Strukturuche implementiert, wird häufig—wie auch in dieser Arbeit—als Greedy-Hillclimbing-Prozedur realisiert, d.h., es wird immer eine potenzielle lokale Veränderung der Struktur (Einfügen einer zusätzlichen Kante, Entfernen einer Kante bzw. Ändern der Richtung einer Kante) untersucht und diejenige Änderung vorgenommen, die zur größten Verbesserung der aktuellen Bewertung führt. Dabei werden die neuen bedingten Wahrscheinlichkeiten θ'_{ijk} für die neue Struktur G' gegebenenfalls unter Anwendung der Inferenzverfahren anhand des „alten" Netzes bestimmt (in der Funktion *berechne_neue_CPTs()*).

STRUKTURELLE EM($B_s, \boldsymbol{D}$)
$B = (G, \boldsymbol{\theta}) \leftarrow B_s$
while ¬*Konvergenz* **do**
  while ¬*Konvergenz* **do**
    $\boldsymbol{\theta}' \leftarrow \arg\max_{\theta} Q(G, \boldsymbol{\theta}, \boldsymbol{D})$
  od
  $G' \leftarrow \arg\max_{G} Q(G, \boldsymbol{\theta}', \boldsymbol{D})$
  $\boldsymbol{\theta}' \leftarrow$ *berechne_neue_CPTs*(B, G')
  $B \leftarrow B' = (G', \boldsymbol{\theta}')$
od

***Abbildung 4.4**: Struktureller EM-Algorithmus*

Vorteile des SEM-Algorithmuses gegenüber anderen Strukturlernverfahren, die ihn für eine Verwendung in dieser Arbeit in besonderem Maße qualifizieren, bestehen darin, dass er sehr allgemein einsetzbar ist. Beispielsweise setzt er keine vorgegebene Ordnung über den Variablen voraus. Der SEM-Algorithmus bietet gute Möglichkeiten vorhandenes A-priori-Wissen entweder durch die Spezifikation einer Ausgangsstruktur für den Suchprozess oder entsprechender A-priori-Wahrscheinlichkeitsverteilungen im Bayes'schen Ansatz einzubringen. Der Grundalgorithmus kann gleichermaßen sowohl bei vollständigen als auch unvollständigen Trainingsdaten angewendet werden.

4.5 Adaption Bayes'scher Netze

Nachdem bisher die meist offline eingesetzen Batchlernverfahren diskutiert wurden, folgen in diesem Abschnitt Adaptionsverfahren für Bayes'sche Netze. Das Adaptionsproblem lässt sich als Variante des allgemeinen maschinellen Lernproblems Bayes'scher Netze (Definition 4.1) formulieren:

Definition 4.2 (Adaptionsproblem Bayes'scher Netze) *Gegeben sei ein Bayes'sches Netz $B = (G, \boldsymbol{\theta})$ sowie ein* Adaptionsfall *D^{adapt}, finde eine Modifikation $B' = (G', \boldsymbol{\theta}')$ von B, die das verwendete Performanzmaß Q bezüglich $\boldsymbol{D} \cup D^{adapt}$ optimiert.*

In der Praxis werden dabei oft einige aufeinander folgende Adaptionsfälle zu einer größeren Menge zusammengefasst und gemeinsam in einem Adaptionsschritt verarbeitet. Die verwendeten Performanzmaße berücksichtigen meist den zeitlichen Verlauf, d.h., ältere Daten werden üblicherweise im Adaptionsvorgang geringer gewichtet als aktuellere. Dazu muss eine temporale Ordnung $D_1^{adapt} <_t D_2^{adapt} <_t \ldots$ über den Adaptionsfällen $\boldsymbol{D}^{adapt}$ angewendet werden.

Auch beim Adaptionsvorgang unterscheidet man zwischen den beiden die Struktur bzw. die bedingten Wahrscheinlichkeiten betreffenden Teilaufgaben.

4.5.1 Adaption der bedingten Wahrscheinlichkeiten: AHUGIN

Ein Standardverfahren zur Adaption der bedingten Wahrscheinlichkeiten θ_{ijk} der CPTs $\boldsymbol{\theta}$ Bayes'scher Netze ist das AHUGIN-Verfahren von Spiegelhalter und Lauritzen (1990) bzw. Olesen, Lau-

ritzen und Jensen (1992). Es ist in der Lage sowohl mit vollständigen als auch mit unvollständigen Adaptionsfällen $\boldsymbol{D}^{adapt}$ umzugehen.

Bei vollständigen Adaptionsfällen entspricht das Verfahren dem sequentiellen Bayes'schen Lernen der bedingten Wahrscheinlichkeiten unter Verwendung von Dirichlet-Verteilungen wie in Abschnitt 4.3.1 anhand der Abbildung erläutert. Zusätzlich bietet das Verfahren die Möglichkeit einen freien Parameter, den so genannten *Fading Factor* f, zu spezifizieren, der bewirkt, dass ältere Daten nach und nach vom Netz „vergessen" werden. Technisch gesehen dient er u.a. zum Glätten der Verteilungskurven sowie zur Begrenzung der ESS-Werte auf maximale Werte, so dass vermieden wird, dass sich das Verfahren nach einer gewissen Zeit und einer entsprechend großen Anzahl an berücksichtigten Adaptionsfällen durch eine zu schmale Glockenform der Kurve bzw. zu große resultierende ESS-Werte auf ein Modell festlegt und keine Adaption mehr vorgenommen wird.

Bei unvollständigen Adaptionsdaten bietet die AHUGIN-Methode eine Approximation des Bayes'schen Ansatzes, der in diesem Fall nicht mehr praktisch handhabbar ist. Anstelle einer Linearkombination mehrerer Dirichlet-Verteilungen, wie die korrekte Lösung es verlangen würde (siehe Olesen et al., 1992), bestimmt AHUGIN eine Approximation der A-posteriori-Verteilung mit nur einer einzigen Dirichlet-Verteilung. Die relevante mathematische Formalisierung wird im Zusammenhang mit der Beschreibung des Verfahrens der differentiellen Adaption in Abschnitt 6 präsentiert.

4.5.2 Adaption der Struktur

> *"Unfortunately, no handy method for incremental adaptation of structure has been constructed."* (Jensen, 2001, S. 92)

Die Aussage von Jensen (2001) charakterisiert den aktuellen Stand der Forschung bezüglich der Adaption der Struktur Bayes'scher Netze anhand neuer Daten. Der Grund, den er für den Mangel an handhabbaren Strukturadaptionsverfahren anführt, ist die Beobachtung, dass strukturelle Veränderungen—im Gegensatz zu quantitativen Veränderungen der bedingten Wahrscheinlichkeiten—in diskreten Schritten erfolgen, die lediglich anhand einer akkumulierten Menge an Daten erkannt werden können.

Die existierenden Arbeiten (Buntine, 1991; Lam & Bacchus, 1994; Friedman & Goldszmidt, 1997), die sich mit der strukturellen Adaption Bayes'scher Netze befassen, lassen sich deshalb zwei Grundansätzen zuordnen: (a) das Sammeln von Fällen, auf deren gemeinsamer Basis Adaptionsentscheidungen getroffen werden und (b) das Verwalten mehrerer alternativer Modelle, die im Rahmen eines Model-Averaging-Konzepts gemäß ihrer A-posteriori-Wahrscheinlichkeiten im Inferenzprozess gewichtet werden. Unter Punkt (a) fällt beispielsweise der naive Ansatz, bei dem nach einer gewissen Zeitspanne wiederholt ein neues Modell anhand der neuen erhobenen Daten mittels Batchlernverfahren ermittelt wird. Eine im Vergleich zum nachfolgenden kompakten Überblick wesentlich ausführlichere vergleichende Diskussion der genannten Verfahren geben Roure und Sangüesa (1999).

Die existierenden Methoden besitzen den Nachteil, dass es sich meist um *lokale* Adaptionsverfahren handelt, das bedeutet, dass sie im Fall unvollständiger Adaptionsdaten, wenn die lokale Dekomposition der Bewertungsfunktionen wie sie in Abschnitt 4.3.1 beschrieben wurde nicht mehr erfüllt ist, nicht anwendbar sind (Buntine, 1991; Lam & Bacchus, 1994). Zusätzlich setzt beispielsweise Buntine (1991) eine feste Ordnung über den Variablen des Bayes'schen Netzes

voraus. Lam und Bacchus (1994) schlagen ein Verfahren vor, das im Wesentlichen die bekannte, in den alten Daten enthaltene Information, in einem Bayes'schen Netz zusammenfasst, welches daraufhin lokal anhand der neuen Adaptionsfälle modifiziert wird. Potenzieller Nachteil eines solchen Ansatzes ist es, dass durch die Komprimierung der Information in einem einzigen Modell der Adaptionsvorgang in einer Weise beeinflusst wird, so dass weitreichendere globale Veränderungen nicht mehr erfolgen können. Durch das vorhandene Modell wird der Suchprozess zu stark auf die Nachbarbereiche fokussiert. Das Verfahren der sequentiellen Anpassung von Friedman und Goldszmidt (1997), das sich mehr auf das sequentielle Erlernen einer Struktur als auf die Adaption an veränderte Einsatzkontexte konzentriert, versucht dieses Problem durch das Verwalten einer so genannten „Suchfront" (engl. „search frontier") zu verringern, die aus vielversprechenden alternativen Strukturen besteht. Nur solche Strukturen können im nächsten Schritt als neue aktuell „beste" Struktur ausgewählt werden.

Neben diesen aufwendigen Verfahren kommen in praktischen Anwendungen häufig das bereits erwähnte wiederholte Neulernen als eine einfache Ad-hoc-Lösung zur Aktualisierung der Struktur eines Bayes'schen Netzes zum Einsatz.

5 LERNEN INTERPRETIERBARER BEDINGTER WAHRSCHEINLICHKEITEN BAYES'SCHER NETZE

Inhalt dieses Kapitels ist die Vorstellung und Evaluation eines Verfahrens zum Erlernen *interpretierbarer* bedingter Wahrscheinlichkeiten der CPTs Bayes'scher Netze. Damit wird einer der zentralen Aspekte der in Kapitel 4 diskutierten Konzeption des maschinellen Lernens Bayes'scher Netze für benutzeradaptive Systeme behandelt (vgl. die markierten Teile in Abbildung 5.1). Nach der Formulierung und Diskussion der Problemstellung wird das *Verfahren des Lernens mit qualitativen Constraints* entwickelt und im Anschluss sowohl anhand synthetisch erzeugter als auch empirisch erhobener Daten in unterschiedlichen Lernsituationen evaluiert. Es handelt sich dabei um eine Modifikation der Standardlernverfahren, die es ermöglicht, vorhandenes qualitatives Wissen in den Lernvorgang einzubringen und zu berücksichtigen.

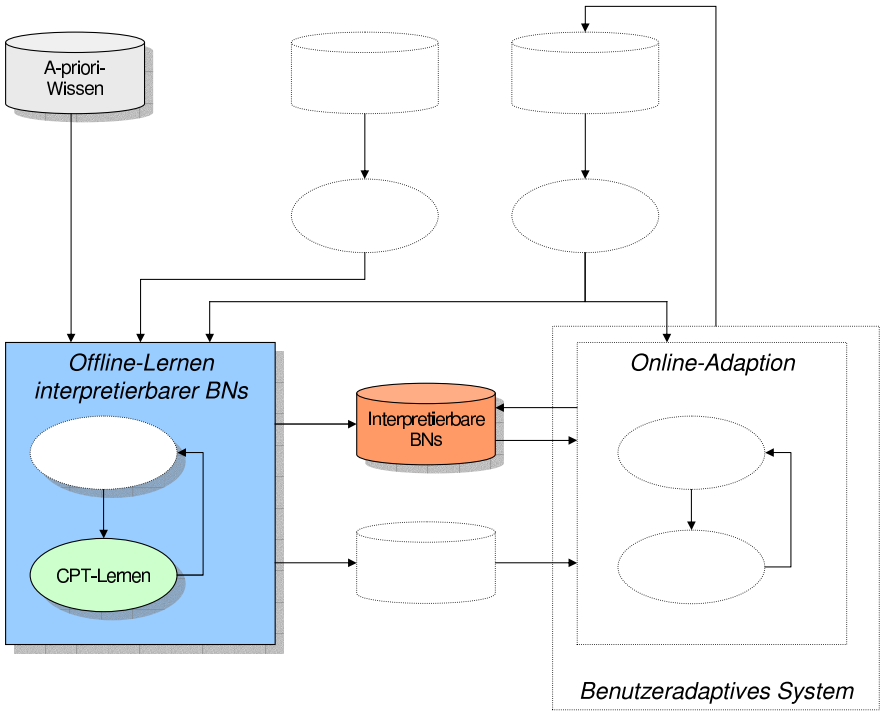


***Abbildung 5.1**: Einordnung des Lernens interpretierbarer bedingter Wahrscheinlichkeiten in die integrative Konzeption*
(Die in diesem Kapitel diskutierten Teile der Konzeption sind farbig gekennzeichnet.)

5.1 Motivation: Interpretierbarkeit der erlernten Modelle durch verborgene Variablen

Aus den in Abschnitt 4.2.5 diskutierten Gründen spielen verborgene Variablen in Benutzermodellen in Form Bayes'scher Netze eine große Rolle. Der Fokus liegt dabei wesentlich auf der resultierenden erhöhten Interpretierbarkeit der Modelle.

Allerdings ergibt sich aus der Verwendung verborgener Variablen—zusätzlich zur höheren Komplexität der benötigten Lernverfahren (vgl. Abschnitt 4.2.5)—ein weiteres Problem: Es ist nicht gesichert, dass die Lernverfahren CPTs ermitteln, die mit der intendierten Semantik korrespondieren, sondern dass statt dessen die angestrebte Eigenschaft der Interpretierbarkeit zerstört bzw. nicht erzielt wird. Durch die Vorgabe der Struktur des Bayes'schen Netzes kann zwar die potenzielle Existenz eines kausalen Zusammenhangs zwischen Variablen für die Lernprozedur spezifiziert werden, es wird damit allerdings nichts hinsichtlich der quantitativen Ausprägung des Zusammenhangs—kodiert in den CPTs—ausgesagt. Somit kann die Situation auftreten, dass zwar bedingte Wahrscheinlichkeiten erlernt werden, welche die empirischen Daten sehr gut modellieren, die aber in keiner Weise die zugrunde liegenden kausalen Beziehungen der verborgenen Variablen auch qualitativ adäquat repräsentieren. Ein Beispiel einer solchen Situation ist das Bayes'sche Netz zur Modellierung der Versuchspersonen des Anweisungsexperiments aus Abbildung 2.5 (b): Die Anwendung eines der in Abschnitt 4.3 beschriebenen Standardlernverfahren (EM) lieferte ein Netz, in dem eine Erhöhung der Anweisungsanzahl von zwei auf drei—entgegen der Erwartungshaltung—eine Verminderung des Erwartungswertes der kognitiven Belastung bewirkt, wohingegen sich die weitere Erhöhung von drei auf vier Anweisungen in einem—erwarteten—Anstieg der Wahrscheinlichkeit für eine höhere kognitive Belastung widerspiegelt. Solche Ergebnisse des Lernprozesses repräsentieren nicht die intendierte monotone Beziehung zwischen der Variablen ANZAHL DER ANWEISUNGEN und KOGNITIVE BELASTUNG. Betrachtet man die weiteren Elternvariablen von KOGNITIVE BELASTUNG, so beobachtet man in diesem erlernten Netz eine Vielzahl solcher unerwünschten Muster. Eine Kompensation dieser Effekte wird vom angewendeten Lernverfahren in den CPTs der Kinder der verborgenen Variablen, d.h., den Symptomvariablen, vorgenommen, so dass das gesamte Bayes'sche Netz dennoch eine sehr gute (numerische) Modellierung der beobachteten empirischen Daten bzw. der zugrunde liegenden gemeinsamen Wahrscheinlichkeitsverteilung darstellt. Im Beispiel bedeutet dies, dass jede Erhöhung der Anzahl der zu bearbeitenden Anweisungen die Wahrscheinlichkeiten für mehr Fehler und längere Ausführungszeiten erhöht. Damit ist allerdings die Eigenschaft der Interpretierbarkeit eines solchen Bayes'schen Netzes in keiner Weise gegeben.

Diese Problematik kann im Wesentlichen zwei potenzielle Gründe haben: (a) eine „falsche" Benennung der Zustände der verborgenen Variablen durch die Standardlernverfahren und (b) die typischerweise hohe Dimensionalität des Suchraums mit vielen lokalen Maxima, in denen der Lernvorgang—möglicherweise irrtümlich—bei hinsichtlich der Interpretierbarkeit unerwünschten Lösungen „hängen bleiben" kann. Punkt (a) könnte zur im angesprochenen Beispiel beschriebenen Situation führen, falls der verwendete Lernalgorithmus z.B. die beiden Zustände *niedrig* und *mittel* der Variablen KOGNITIVE BELASTUNG „vertauscht" hat. Dieser Effekt kann in diesem speziellen Fall in einfacher Weise durch eine manuelle Umbennung der Zustände nach dem Lernvorgang korrigiert werden, was aber wegen des komplexen Zusammenspiels der Variablenzustände im Zusammenhang mit einer verborgenen Variablen im Allgemeinen keine triviale Aufgabe darstellt. Punkt (b) ist eng verknüpft mit der Overfitting-Problematik: Je geringer die verfügbare Trainingsmenge,

desto größer ist üblicherweise die Anzahl der vorhandenen lokalen Maxima des Lösungsraums. Das in diesem Kapitel vorgestellte Verfahren versucht den Lernvorgang derart zu modifizieren, dass möglichst viele der lokal optimalen Lösungen, die keine interpretierbaren Modelle repräsentieren, als potenzielle Resultate ausgeschlossen werden. Gleichzeitig erwartet man, dass durch das Ausschließen „schlechter“, nicht mit der intendierten Semantik in Übereinstimmung zu bringender Lösungen, auch die prediktive Qualität der Lernergebnisse verbessert wird.

Die Entwicklung eines solchen Verfahrens wurde von Binder et al. (1997) in ihrem Papier zur Beschreibung der APN-Methode als wichtige offene, allgemeine Aufgabenstellung—nicht nur für den speziellen Kontext benutzeradaptiver Systeme—erkannt. Dieses Kapitel bzw. die zugehörige Veröffentlichung (Wittig & Jameson, 2000) kann als eine detaillierte Ausarbeitung einer in beliebigen Szenarien einsetzbaren Lösung inklusive einer ausführlichen empirischen Evaluierung angesehen werden. Dabei wird an dieser Stelle das Verfahren erstmals mit empirisch erhobenen Daten getestet—zusätzlich zu den bereits in der genannten Veröffentlichung (in ähnlicher Form) durchgeführten Analysen mit synthetisch erzeugten Datensätzen. Weiterhin werden Ergebnisse ergänzender Untersuchungen vorgestellt, die die Eigenschaften des Verfahrens tiefergehend beleuchten.

Das Grundprinzip des im Rahmen dieser Arbeit entwickelten Verfahrens basiert auf der Vorgabe von Information zur Qualität der Beziehung zwischen den verborgenen Variablen und ihren Nachbarn in der Struktur des Bayes’schen Netzes. Die Bewertungsfunktionen der Standardverfahren werden um einen zusätzlichen Term erweitert, der gerade in den Fällen zu schlechteren Bewertungen führt, in denen das untersuchte Netz den vorgegebenen qualitativen Informationen widerspricht. Dadurch wird der Suchprozess weitestgehend nur durch solche Bereiche des Lösungsraums „geführt“, die (in großem Umfang) mit den qualitativen Informationen konsistent sind.

5.2 Methode des Lernens mit qualitativen Constraints

Das Verfahren baut auf Ideen der von Wellman (1990) eingeführten *qualitativen probabilistischen Netze* und verwandter Arbeiten von Druzdzel und van der Gaag (1995) auf. Bei qualitativen probabilistischen Netzen handelt es sich im Wesentlichen um einen Spezialfall Bayes’scher Netze, deren Kanten statt mit CPTs (lediglich) mit qualitativen Informationen annotiert werden. Formen dieser qualitativen Informationen sind u.a. monotone Beziehungen, die mit dem Begriff der *qualitativen Einflüsse* bezeichnet werden, und qualitative Synergien.[1] Beispielsweise würde man in einem qualitativen probabilistischen Netz zur Modellierung des Anweisungsexperiments der entsprechenden Kante eine positive (+) monotone Beziehung zwischen den Variablen ANZAHL DER ANWEISUNGEN und KOGNITIVE BELASTUNG zuordnen. Damit kann die Annahme modelliert werden, dass mehr Anweisungen zu einer erhöhten kognitiven Belastung der Versuchsperson führen. Abbildung 5.2 zeigt die zur Modellierung der beiden Experimente verwendeten Netze, annotiert mit den nahe liegenden qualitativen Einflüssen zwischen den Variablen. Es hat sich gezeigt, dass diese eingeschränkte Variante Bayes’scher Netze in einer Vielzahl von Anwendungsszenari-

[1] Anschließend wird das allgemeine Verfahren anhand des—nach Ansicht des Autors—wichtigsten Falls der monotonen Beziehungen, d.h., der qualitativen Einflüsse eingeführt. Eine Erweiterung auf die anderen von Druzdzel und van der Gaag (1995) angeführten Arten von qualitativen Informationen lässt sich analog durchführen.

en erfolgreich eingesetzt werden kann. Insbesondere der Wegfall des aufwendigen Prozesses der Spezifikation der bedingten Wahrscheinlichkeiten durch Experten sowie die sehr effizienten Inferenzmechanismen qualitativer probabilistischer Netze führen in entsprechenden Domänen dazu, dass sie „normalen“ Bayes'schen Netzen vorgezogen werden.

Das im Folgenden beschriebene *Verfahren des Lernens mit qualitativen Constraints* erweitert die Standardlernverfahren der CPTs Bayes'scher Netze um die Möglichkeit, vor Beginn des Lernvorgangs die verfügbaren qualitativen Informationen in Form so genannter *qualitativer Constraints*[2] für die Lernaufgabe vorzugeben, mit dem Ziel, sowohl (a) *die Interpretierbarkeit der Resultate zu verbessern* als auch (b) *die Performanz der erlernten Netze durch das Vermeiden von Overfitting zu erhöhen*.

5.2.1 Qualitative Constraints für den Lernprozess

Angenommen, ein Domänenexperte wurde gefragt, ob die bedingten Wahrscheinlichkeiten θ_{ijk} der CPTs $\boldsymbol{\theta}$ des zu lernenden Bayes'schen Netzes eine bestimmte Menge $\boldsymbol{C}$ qualitativer Constraints erfüllen, und der Experte hat mit „Ja“ geantwortet. Wie kann diese Tatsache im Zusammenhang mit Gleichung 4.2 berücksichtigt werden?

Eine mögliche Konzeptualisierung besteht in der Interpretation der Aussage des Experten als eine von ihm anhand einer geschätzten Wahrscheinlichkeit getroffenen Entscheidung, die in Bezug gesetzt wird zur tatsächlichen Situation, d.h., in welchem Ausmaß die Constraints $\boldsymbol{C}$ tatsächlich vom betrachteten Netz erfüllt werden.

Formal lässt sich dies folgendermaßen beschreiben: Angenommen, es wird eine Funktion $\mathit{violation}(\boldsymbol{\theta}, \boldsymbol{C})$ definiert, die das tatsächliche Ausmaß repräsentiert, inwieweit die CPTs $\boldsymbol{\theta}$ die qualitativen Constraints $\boldsymbol{C}$ verletzen: *violation* nimmt den Wert 0 an, wenn kein Verletzung vorliegt, andernfalls nimmt sie einen positiven Wert an, der mit zunehmendem Grad der Verletzung ansteigt.

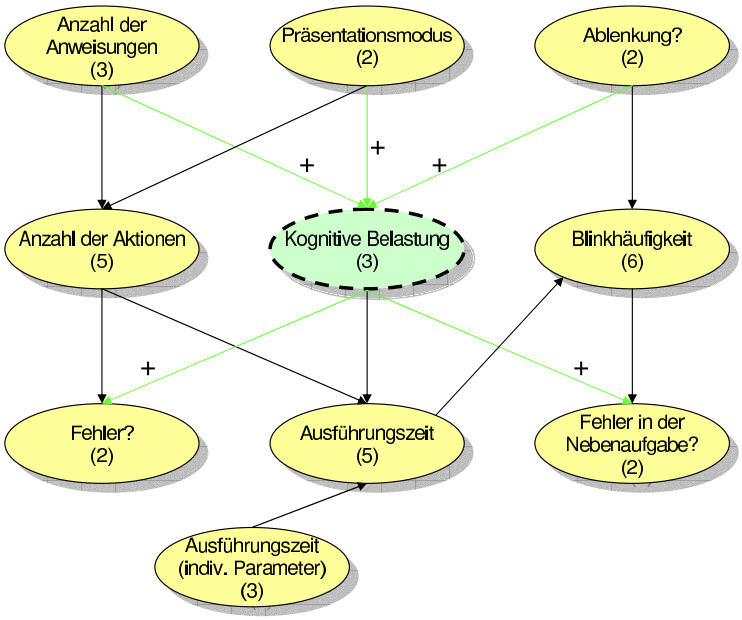
Betrachtet man die Wahrscheinlichkeit dafür, dass der Experte die Antwort „Ja“ gibt, als eine Funktion von $\mathit{violation}(\boldsymbol{\theta}, \boldsymbol{C})$, dann sollte diese Wahrscheinlichkeit gegen 0 tendieren, wenn $\mathit{violation}(\boldsymbol{\theta}, \boldsymbol{C})$ sich von ihrem Minimum 0 entfernt (wie in Abbildung 5.3 schematisch dargestellt).

Eine Funktion, die diese Anforderungen erfüllt, ist die folgende:

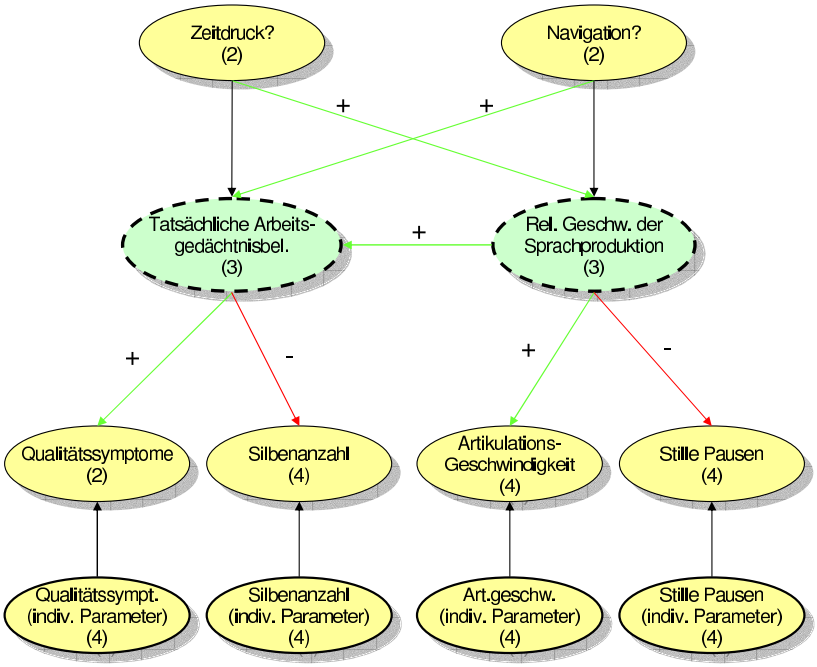
$$P(\mathit{Antwort} = \mathit{ja} \mid \boldsymbol{\theta}, \boldsymbol{C}) = e^{-w \cdot \mathit{violation}(\boldsymbol{\theta}, \boldsymbol{C})}. \tag{5.1}$$

Das positive Gewicht w—im Folgenden auch als *Constraint-Gewicht* bezeichnet—bestimmt wie schnell die Wahrscheinlichkeit von ihrem Maximum 1 abnimmt, wenn im gleichen Zug das Ausmaß der Constraint-Verletzungen $\mathit{violation}(\boldsymbol{\theta}, \boldsymbol{C})$ ausgehend von deren Minimum 0 zunimmt.

[2] In diesem Zusammenhang steht der Begriff ‘Constraint’ für zusätzliche Informationen, die den Suchprozess einschränken. Er steht in keinem engeren Zusammenhang mit dem formalen Begriff aus dem Forschungsgebiet der logischen Constraint-Programmierung.



(a) Anweisungsexperiment



(b) Flughafenexperiment

***Abbildung 5.2**: Qualitative Zusammenhänge zwischen den Variablen der beiden Experimente (+ bzw. eine grüne Kante markiert einen positiven qualitativen Zusammenhang, - bzw. eine rote Kante einen negativen)*

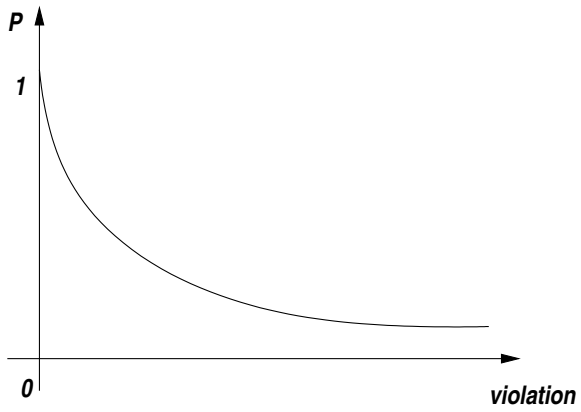


***Abbildung 5.3**: Schematische Darstellung der* violation*-Funktion*

Die Aussage des Experten kann als eine einzige—aber besonders wichtige—„Beobachtung“ angesehen werden, die zusammen mit den „normalen“ Beobachtungen in Form der Trainingsfälle beim Lernen berücksichtigt werden kann. Dementsprechend kann die Log-Likelihood dieser „Experten-Beobachtung“ zur rechten Seite der Gleichung 4.2 addiert werden—in analoger Vorgehensweise zum Einbringen der Terme zur Bestrafung zu komplexer Strukturen im frequentistischen Ansatz des Strukturlernens (vgl. Abschnitt 4.4.2)—, um eine modifizierte, *erweiterte Log-Likelihood* aller Beobachtungen bzw. Daten zu erhalten:

$$\ln P(\boldsymbol{D} \mid \boldsymbol{\theta}) - w \cdot \mathit{violation}(\boldsymbol{\theta}, \boldsymbol{C}). \tag{5.2}$$

Eine alternative Sichtweise ist die Interpretation der Expertenaussage als subjektive Einschätzung des Experten im Rahmen des MAP-Lernansatzes. Die A-posteriori-Wahrscheinlichkeit $P(\boldsymbol{\theta} \mid \boldsymbol{D}, \boldsymbol{C})$ wird nach dem Satz von Bayes aus der Likelihood der Daten $P(\boldsymbol{D} \mid \boldsymbol{\theta})$ und der A-priori-Einschätzung des Experten $P(\boldsymbol{\theta} \mid \boldsymbol{C})$ berechnet als:

$$P(\boldsymbol{\theta} \mid \boldsymbol{C}) = \beta \cdot e^{-w \cdot \mathit{violation}(\boldsymbol{\theta}, \boldsymbol{C})}, \tag{5.3}$$

$$P(\boldsymbol{\theta} \mid \boldsymbol{D}, \boldsymbol{C}) = \gamma \cdot P(\boldsymbol{D} \mid \boldsymbol{\theta}) \cdot e^{-w \cdot \mathit{violation}(\boldsymbol{\theta}, \boldsymbol{C})}. \tag{5.4}$$

β und γ sind die notwendigen Normalisierungskonstanten, die für die Maximierungsaufgabe keine Rolle spielen. Wird der Logarithmus auf diese Gleichung angewendet, ergibt sich Formel 5.2 (bis auf eine für die Maximierung nicht relevante Konstante).

Die Aufgabe besteht nun darin, diese erweiterte Log-Likelihood zu maximieren. Der Term $\mathit{violation}(\boldsymbol{\theta}, \boldsymbol{C})$ kann als „Strafterm“ angesehen werden, der den Suchalgorithmus veranlasst, Regionen des Suchraumes zu meiden, die qualitative Constraints verletzen. w stellt dabei das Gewicht des Strafterms in Relation zur Likelihood der Daten dar.

Die empirischen Ergebnisse, die in diesem Kapitel vorgestellt werden, legen den Schluss nahe, dass in Situationen, in denen die qualitativen Constraints $\boldsymbol{C}$ tatsächlich vom Modell, das zur Erzeugung der Daten genutzt wurde, erfüllt werden, Lösungen produziert werden, die sich durch einen *violation*-Wert (nahe) 0 auszeichnen.

Es wäre denkbar, alternative Funktionen in Gleichung 5.1 anstelle der Exponentialfunktion zu verwenden. Die diesbezüglich zu erfüllende Mindestanforderung besteht neben den bereits

in Abschnitt 5.2.1 genannten in der Differenzierbarkeit. Eine Bestimmung des tatsächlichen Zusammenhangs zwischen den Verletzungen der qualitativen Constraints und den Expertenansichten würde empirischen Untersuchungen erfordern, wobei es wahrscheinlich unmöglich ist, eine domänenunabhängige Formulierung zu bestimmen. Die obigen Ausführungen können deshalb als eine Spezifikation eines prototypischen Szenarios angesehen werden, in welchem dem Strafterm eine probabilistische Interpretation zugewiesen werden kann.

Um Formel 5.2 zum Erlernen interpretierbarer CPTs Bayes'scher Netze einsetzen zu können, müssen die folgenden beiden Fragen geklärt werden:

1. Wie kann die *violation*-Funktion für sinnvolle Klassen qualitativer Zusammenhänge definiert und motiviert werden?

2. Welche Algorithmen können zur Maximierung der neuen Bewertungsfunktion 5.2 verwendet werden?

Diese Fragen werden in den folgenden beiden Abschnitten geklärt.

5.2.2 Formalisierung qualitativer Constraints

Im Zusammenhang mit den qualitativen probabilistischen Netzen von Wellman (1990) geben Druzdzel und van der Gaag (1995) formale probabilistische Definitionen verschiedener Arten qualitativer Beziehungen, die zwischen Variablen Bayes'scher Netze existieren können. Druzdzel und van der Gaag nutzen diese Definitionen im Rahmen einer Methode zur Kombination verschiedener Wissensarten zur Spezifikation der CPTs Bayes'scher Netze. Allerdings setzen sie keinerlei Standardlernverfahren, wie den EM-Algorithmus oder gradienten-basierte Methoden ein. Das vorgestellte Verfahren des Lernens mit qualitativen Constraints kann als eine Integration von Teilen der Methode von Druzdzel und van der Gaag (1995) und Standardlernverfahren angesehen werden.

5.2.2.1 Qualitative Einflüsse zwischen Variablen

Eine der von Druzdzel und van der Gaag (1995) betrachteten Klassen von Informationen stellen *qualitative Einflüsse* dar, die monotone Beziehungen zwischen den Werten zweier benachbarter Variablen eines Bayes'schen Netzes repräsentieren.

Das Konzept eines qualitativen Einflusses ist nur dann anwendbar, wenn eine Ordnung über den Zuständen der beteiligten Variablen definiert wurde. Beispielsweise bietet sich folgende Ordnung der Zustände der Variablen KOGNITIVE BELASTUNG an: $niedrig < mittel < hoch$. Ohne Beschränkung der Allgemeinheit kann für alle relevanten Variablen X_i als Ordnung ihrer diskreten Zustände $x_{i1} < x_{i2} < \ldots < x_{in_i}$ angenommen werden. Ein qualitativer Einfluss wird mit $S^?(X_w, X_z)$ bezeichnet, wobei $? \in \{+, -\}$ die Qualität des monotonen Zusammenhangs zwischen einer Variablen X_w und eines ihrer Kinder X_z angibt. Es existieren zwei Arten qualitativer Einflüsse: Gilt ein *positiver* (+), dann bedingt eine Erhöhung des Zustandes von X_w ebenfalls eine Erhöhung (zumindest keine Verminderung) des Zustandes von X_z. Ist die Beziehung *negativen* Charakters (-), bedingt die Erhöhung bezüglich X_w eine Verminderung (zumindest keine Erhöhung) hinsichtlich X_z. Formal definieren Druzdzel und van der Gaag (1995) einen qualitativen Einfluss folgendermaßen:

Definition 5.1 (Positiver qualitativer Einfluss) *Ein* positiver qualitativer Einfluss $S^+(X_w, X_z)$ *zwischen einer Variablen X_w und einem ihrer Kinder X_z in einem Bayes'schen Netz existiert genau dann wenn gilt: Für alle Zustände x_{zm} von X_z mit $m > 1$ und allen Paaren unterschiedlicher Zustände x_{wi}, x_{wj} von X_w derart, dass $i > j$, sowie allen möglichen Zustandskombinationen $\boldsymbol{y}$ der Eltern der Variablen X_z ausgenommen X_w, $\boldsymbol{Pa}(X_z) \setminus X_w$, folgende Ungleichung gilt:*

$$P(X_z \geq x_{zm} \mid x_{wi}, \boldsymbol{y}) \geq P(X_z \geq x_{zm} \mid x_{wj}, \boldsymbol{y}). \tag{5.5}$$

Diese Ungleichung kann mit Hilfe der bedingten Wahrscheinlichkeiten der verschiedenen Zustände von X_z formuliert werden, was in einer Menge von Ungleichungen der nachfolgenden Art resultiert:

$$\sum_{l=m}^{n_z} P(x_{zl} \mid x_{wi}, \boldsymbol{y}) \geq \sum_{l=m}^{n_z} P(x_{zl} \mid x_{wj}, \boldsymbol{y}). \tag{5.6}$$

Für jede der Kombinationen aus einem x_{zm} mit $m > 1$, einem Paar x_{wi} und x_{wj} mit $i > j$, und einer Zustandskonfiguration $\boldsymbol{y}$ der von X_w verschiedenen Eltern von X_z existiert eine solche Ungleichung, deren Gesamtheit den aus diesem qualitativen Einfluss $S^+(X_w, X_z)$ resultierenden qualitativen Constraint für die Lernprozedur repräsentiert.[3]

Negative qualitative Einflüsse werden analog definiert:

Definition 5.2 (Negativer qualitativer Einfluss) *Ein* negativer qualitativer Einfluss $S^-(X_w, X_z)$ *zwischen einer Variablen X_w und einem ihrer Kinder X_z in einem Bayes'schen Netz existiert genau dann wenn gilt: Für alle Zustände x_{zm} von X_z mit $m > 1$ und allen Paaren unterschiedlicher Zustände x_{wi}, x_{wj} von X_w derart, dass $i > j$, sowie allen möglichen Zustandskombinationen $\boldsymbol{y}$ der Eltern der Variablen X_z ausgenommen X_w, $\boldsymbol{Pa}(X_z) \setminus X_w$, folgende Ungleichung gilt:*

$$P(X_z \geq x_{zm} \mid x_{wi}, \boldsymbol{y}) \leq P(X_z \geq x_{zm} \mid x_{wj}, \boldsymbol{y}). \tag{5.7}$$

5.2.2.2 Konstruktion einer Bewertungsfunktion zum Lernen mit qualitativen Constraints

Mit der Menge der Ungleichungen aus 5.6 ist man in der Lage, ein Maß der Verletzungen der qualitativen Constraints $\boldsymbol{C}$ bezüglich der CPTs $\boldsymbol{\theta}$ zu definieren, d.h., eine Definition der *violation*-Funktion zu geben. Ungleichung 5.6, ein Teil der mathematischen Beschreibung eines positiven qualitativen Einflusses $S^+(X_w, X_z)$ von X_w auf X_z, kann man umformulieren zu:

$$\underbrace{\sum_{l=m}^{n_z} P(x_{zl} \mid x_{wi}, \boldsymbol{y}) - \sum_{l=m}^{n_z} P(x_{zl} \mid x_{wj}, \boldsymbol{y})}_{=:c'^{?wz}_{mij\boldsymbol{y}}} \geq 0. \tag{5.8}$$

Zu jedem verletzten positiven qualitativen Constraint muss mindestens eine solche Ungleichung existieren, die nicht erfüllt ist, d.h., bei der die Differenz der linken Seite der Ungleichung negativ wird. Analog führen Verletzungen negativer Constraints zu Werten *größer* 0.

[3] Für eine eindeutige Definition eines qualitativen Einflusses werden lediglich die zu benachbarten Zuständen gehörenden Ungleichungen benötigt, d.h, wenn gilt $i = j + 1$, da die verbleibenden Ungleichungen durch die Transitivitätseigenschaft der Ordnungsrelation $<$ impliziert werden. Allerdings ermöglichen die redundanten Ungleichungen in Fällen, in denen ein qualitativer Constraint verletzt wird, die Identifikation aller an der Verletzung beteiligten Werte. Damit wird es ermöglicht, alle beteiligten Werte gleichzeitig zu verändern, um die Verletzung schneller zu beheben. Dieser Sachverhalt wird im weiteren Verlauf der Beschreibung des Verfahrens deutlich werden.

Ein einer einzigen Ungleichung zugeordneter *partieller Verletzungsterm* $c^{?wz}_{mij\boldsymbol{y}}$ kann wie folgt definiert werden:

$$c^{?wz}_{mij\boldsymbol{y}} := \begin{cases} -c'^{?wz}_{mij\boldsymbol{y}} & \text{, falls } ? = + \text{ und } c'^{?wz}_{mij\boldsymbol{y}} < 0, \\ c'^{?wz}_{mij\boldsymbol{y}} & \text{, falls } ? = - \text{ und } c'^{?wz}_{mij\boldsymbol{y}} > 0, \\ 0 & \text{, sonst.} \end{cases} \tag{5.9}$$

Der gesamte Verletzungsterm *violation*$(\boldsymbol{\theta}, \boldsymbol{C})$ wird als die Summe aller relevanten partiellen Verletzungsterme definiert:

$$\textit{violation}(\boldsymbol{\theta}, \boldsymbol{C}) := \sum_{m,i,j,\boldsymbol{y},w,z} c^{?wz}_{mij\boldsymbol{y}}, \tag{5.10}$$

wobei ? die Qualität (+ oder -) des zu den Indizes w und z gehörigen qualitativen Einflusses $S^?(X_w, X_z)$ bezeichnet. Es ist zu beachten, dass für jede der Kombinationen der zu diesen Indizes zugeordneten Variablen nur eine einzige Qualität ? existieren kann, da es keinen Sinn macht, sowohl einen positiven als auch einen negativen Einfluss zwischen denselben Variablen zu deklarieren.

5.2.3 Integration der qualitativen Constraints in die Standardlernverfahren

Nachdem erläutert wurde wie der von Gleichung 5.2 benötigte *violation*-Term definiert werden kann, bleibt die Frage, wie dieser Ausdruck zu maximieren ist. Eine mögliche Lösung besteht in Modifikationen der in Abschnitt 4.3 vorgestellten iterativen Standardlernverfahren der CPTs Bayes'scher Netze, die im Folgenden detailliert besprochen werden.

5.2.3.1 Adaptive-Probabilistic-Networks mit qualitativen Constraints

Um den APN-Algorithmus zur Maximierung der erweiterten Log-Likelihood aus Gleichung 5.2 einzusetzen, muss ein erweiterter Gradient berechnet werden:

$$\boldsymbol{\nabla} \ln P(\boldsymbol{D} \mid \boldsymbol{\theta}) - \boldsymbol{\nabla} w \cdot \textit{violation}(\boldsymbol{\theta}, \boldsymbol{C}). \tag{5.11}$$

Die partiellen Ableitungen des ersten Terms wurden bereits in Gleichung 4.15 angegeben. Bezüglich des zweiten Terms geht man wie folgt vor:

$$\nabla^u_{ijk} w \cdot \textit{violation}(\boldsymbol{\theta}, \boldsymbol{C}) = w \cdot v_{ijk}(\boldsymbol{\theta}, \boldsymbol{C}). \tag{5.12}$$

Die $v_{ijk}(\boldsymbol{\theta}, \boldsymbol{C})$ sind die partiellen Ableitungen der *violation*-Funktion nach den bedingten Wahrscheinlichkeiten θ_{ijk}. Diese partiellen Ableitung sind nach Ungleichung 5.8 sehr einfach zu bestimmen: Jeder partielle Verletzungsterm ist eine lineare Funktion der CPT-Einträge θ_{ijk}, wobei jeder Eintrag höchstens einmal mit dem Koeffizient +1 oder -1 auftritt. Nur die partiellen Verletzungsterme $c^{?wz}_{mij\boldsymbol{y}}$, die am aktuellen Punkt $\boldsymbol{\theta}$ im Suchraum eine nicht erfüllte Ungleichung repräsentieren, tragen zum gesamten Verletzungsterm *violation*$(\boldsymbol{\theta}, \boldsymbol{C})$ bei.

Die $v_{ijk}(\boldsymbol{\theta}, \boldsymbol{C})$ können wie folgt berechnet werden:

$$v_{ijk}(\boldsymbol{\theta}, \boldsymbol{C}) = v^-_{ijk}(\boldsymbol{\theta}, \boldsymbol{C}) - v^+_{ijk}(\boldsymbol{\theta}, \boldsymbol{C}), \tag{5.13}$$

wobei $v^-_{ijk}(\boldsymbol{\theta}, \boldsymbol{C})$ die Anzahl der verletzten Ungleichungen ist, die auf kleinere θ_{ijk}-Werte hindeuten und $v^+_{ijk}(\boldsymbol{\theta}, \boldsymbol{C})$, diejenige, die auf größere hindeuten.

Damit ergibt sich als Gradient der erweiterten Log-Likelihood aus Gleichung 5.2:

$$\nabla^u_{ijk} = \sum_{l=1}^{s} \frac{P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta})}{\theta_{ijk}} - w \cdot v_{ijk}(\boldsymbol{\theta}, \boldsymbol{C}). \tag{5.14}$$

Wie beim Standard-APN-Algorithmus muss dieser unprojizierte Gradient noch auf die Constraintoberfläche projiziert werden, die durch $\sum_j \theta'_{ijk} = 1$ und $\theta'_{ijk} \in [0, 1]$ definiert ist.

Dieser erweiterte Gradient wird analog zum Standardvorgehen genutzt, um (kleine) Schritte im Suchraum durchzuführen, bis ein lokales Maximum erreicht ist.

5.2.3.2 Expectation-Maximization mit qualitativen Constraints

Es stellt sich die Frage, wie der EM-Algorithmus verwendet werden kann, um die erweiterte Bewertungsfunktion aus Gleichung 5.2 anstelle der Log-Likelihood der Daten zu maximieren?

Die eleganteste Lösung würde eine Anwendung von bezüglich der erweiterten Log-Likelihood modifizierten E- und M-Schritten umfassen. Dazu müsste eine Vorschrift zur Berechnungen der neuen $\boldsymbol{\theta}'$-Werte hergeleitet werden, die anstelle von Gleichung 4.12 bzw. 4.13 angewendet werden kann. Unglücklicherweise ist der allgemeine, speziell für die Maximierung der Likelihood entwickelte EM-Ansatz nicht in gleichem Maße auf alle Bewertungsfunktionen übertragbar (siehe z.B. Dempster et al., 1977). Insbesondere die Anwendung auf die an dieser Stelle betrachtete erweiterte Log-Likelihood führt zu einer Menge abhängiger, nicht-linearer Gleichungen, für die keine analytische Lösung gefunden werden konnte. Diese Problematik wird in Anhang A detailliert beschrieben. Möglicherweise kann die Betrachtung leicht veränderter Bewertungsfunktionen diesbezüglich zum Erfolg führen.[4]

Dennoch erscheint es vielversprechend, das große Potential der Grundidee des EM-Algorithmuses in dieser Anwendungssituation auszunutzen, beispielsweise um vorhandene Implementationen des EM-Algorithmuses um die Fähigkeit des Erlernens interpretierbarer CPTs zu erweitern. Es existiert diesbezüglich bereits eine Vielzahl von erfolgreich eingesetzen, hybriden Ansätzen, die das EM-Verfahren mit gradienten-basierten Methoden kombinieren (vgl. z.B. Ortiz & Kaelbling, 1999; Bauer, Koller & Singer, 1997). Die zugrunde liegende Vorgehensweise aller dieser Methoden ist es, anstelle der Maximierung im M-Schritt lediglich Schritte in Richtung des Anstiegs der Bewertungsfunktion durchzuführen—wie allgemein in gradienten-basierten Verfahren üblich. Der E-Schritt bleibt gegenüber der Originalversion des Algorithmuses unverändert. Solche Verfahren werden als *verallgemeinerte EM-Verfahren* bezeichnet (engl. *generalized EM*, *GEM*).

Das Gerüst des entwickelten, hybriden EM-Algorithmuses zur Maximierung der erweiterten Log-Likelihood bilden zwei alternierend durchgeführte Schritte zur Modifikation der aktuellen bedingten Wahrscheinlichkeiten der CPTs $\boldsymbol{\theta}$:

1. der Standard-M-Schritt aus Gleichung 4.12, der zu einer Zwischenlösung führt, die durch eine höhere Log-Likelihood der Daten charakterisiert ist;

2. ein gradienten-basierter Hillclimbing-Schritt, der den in Gleichung 4.15 spezifizierten Gradienten nutzt, um das Ausmaß der aktuell—nach obigem Schritt—verletzten Constraints zu

[4] Auf eine entsprechende weitergehende Untersuchung dieses Problems wurde an dieser Stelle verzichtet, da sie höchst wahrscheinlich den Rahmen dieser Arbeit gesprengt hätte—insbesondere hinsichtlich der notwendigen Verfahren aus der Statistik.

verringern. Dabei ist eine Schrittweite zu wählen, die in Relation zum Ausmaß des vorgehenden M-Schritts steht—beispielsweise wie in den folgenden Analysen durch eine Normalisierung des *violation*-Gradienten, so dass der betragsmäßig größte Gradienten-Eintrag dem betragsmäßig größten Schritt in einer der möglichen Dimensionen des vorangehenden M-Schritts entspricht. Das danach erhaltene Resultat muss wie beim APN-Verfahren beschrieben ebenfalls einer Projektion auf die Constraint-Oberfläche unterzogen werden. Sind keine der vorgegebenen Constraints $\boldsymbol{C}$ verletzt, erübrigt sich dieser Schritt, da alle partiellen Ableitungen im Gradienten den Wert 0 besitzen.

Dieses Verfahren stellt eine Variante des GEM-Ansatzes dar, indem es den Gradienten des zweiten Schritts auf der Basis der CPTs $\boldsymbol{\theta}'$ *nach* der Durchführung des Standard-M-Schritts ermittelt. Beim üblichen GEM-Vorgehen würde dies im Zuge der Verbesserung der Gesamtbewertung (Log-Likelihood und *violation*-Term) anhand der CPTs $\boldsymbol{\theta}$ vor der aktuellen Iteration geschehen. Die Motivation für diese Verfahrensweise ist es, die aktuellen Verletzungen nach dem M-Schritt zu verringern, nicht die vorhergehenden Verletzungen, die möglicherweise im Zuge der Durchführung des M-Schritts stark verändert—gegebenenfalls sogar gänzlich eliminiert—werden. Außerdem können in dieser Weise bestehende Implementierungen des EM-Algorithmus in einfacher Art um das Lernen mit qualitativen Constraints erweitert werden. Die Implementation des E- und M-Schritts muss dazu nicht verändert werden.

Die Ergebnisse dieser Methode sind theoretisch weniger vorhersagbar als die des Standard-EM-Algorithmuses, da nicht garantiert werden kann, dass die Bewertung mit jeder Iteration verbessert wird: Prinzipiell kann ein M-Schritt zur leichten Verbesserung der Log-Likelihood in einer starken Verschlechterung der Erfüllung der Constraints resultieren. Umgekehrt gilt die analoge Argumentation für die gradienten-basierten Schritte zur Verringerung der Verletzungen. Andererseits sind—unter der Annahme, dass die spezifizierten Constraints tatsächlich gelten—die beiden Teilziele des hybriden Algorithmuses im Allgemeinen kompatibel, und man erwartet deshalb nicht, dass sich die beiden Schritte gegenseitig in ihrer Wirkung neutralisieren. Dies zeigt sich in der Tat in den im Anschluss vorgestellten Ergebnissen der empirischen Analysen.

5.2.4 Diskussion

Das Einbringen qualitativer Constraints in die Standardlernverfahren resultiert nicht in einer erhöhten asymptotischen Komplexität der resultierenden Verfahren. Die Behandlung der Constraints besteht aus geschachtelten Schleifen über der Menge der qualitativen Constraints $\boldsymbol{C}$ und den CPT-Einträgen θ_{ijk} bzw. den zugehörigen Zustandskombinationen zur Berechnung der Summen aus Ungleichung 5.8. Details zu einer entsprechenden Implementierung unter Angabe der Lernverfahren im Pseudocode finden sich bei Decker (2001). Die Komplexität der Lernverfahren wird weiterhin durch die Komplexität der eingesetzten Inferenzverfahren dominiert. Die im Rahmen der in den nachfolgenden Abschnitten vorgestellten Analysen beobachteten Zeiten zur Behandlung der qualitativen Constraints fielen in der Praxis in Relation zu den Gesamtlaufzeiten der Standardvarianten nicht ins Gewicht.

In der vorgestellten Variante der Lernverfahren mit qualitativen Constraints müssen die Schritteweite α der Hillclimbing-Schritte bei der APN-Variante und das Constraint-Gewicht w als Parameter der Verfahren vorgegeben werden. Hinsichtlich der Schrittweite können im APN-Ansatz aufwendigere, konjugierte Gradientenverfahren (Press, 1992) zum Einsatz kommen, die die optimale Schrittweite während des Verfahrens selbst bestimmen. Es ist denkbar, einen Wert für das

Constraint-Gewicht w selbst mit Techniken des maschinellen Lernens zu bestimmen—beispielsweise unter Verwendung von Kreuzvalidierungstechniken. Ebenso erscheint es vielversprechend, mit adaptiven w-Werten arbeiten, die z.B. in einer frühen Phase mit einem größeren Constraint-Gewicht lernen, das, wenn die Constraints zum Großteil erfüllt werden, sukzessive verringert wird, um der Log-Likelihood eine höhere Bedeutung im Lernprozess zukommen zu lassen.

Das vorgestellte Verfahren der qualitativen Constraints erzwingt keine Erfüllung der vorgegebenen Constraints durch die Lernresultate. Dies ist im Kontext benutzeradaptiver Systeme kein Problem, da die modellierten qualitativen Einflüsse im Normalfall gültig sind und den empirischen Trainingsdaten nicht widersprechen. Existieren im Lernergebnis wider Erwarten dennoch verletzte Constraints, so ist dies ein Hinweis darauf, dass die bezüglich der Konstruktion des Netzes gemachten Annahmen überdacht werden sollten. In dieser Weise kann das Verfahren der qualitativen Constraints den Designprozess benutzeradaptiver Systeme durch Vermeidung von Fehlannahmen unterstützen.

Die vorgestellte prinzipielle Vorgehensweise der Erweiterung der Standardbewertungsfunktion kann in vielen Fällen mit alternativen Bewertungsfunktionen ohne größere Veränderungen übernommen werden. Damit können auch Verfahren, die den Lernprozess hinsichtlich bestimmter Inferenzaufgaben optimieren, wie z.B. ELQ (Greiner et al., 1997) oder die Arbeit von Friedman et al. (1997), um den Ansatz der qualitativen Constraints erweitert werden.

5.3 Empirische Evaluation des Verfahrens

In diesem Abschnitt wird das vorgestellte Verfahren der Spezifikation qualitativer Constraints zum Erlernen interpretierbarer CPTs Bayes'scher Netze mit verborgenen Variablen—sowohl die APN- als auch die EM-Variante—anhand der Ergebnisse ausführlicher empirischer Analysen evaluiert. Dazu wurden Versuchsreihen mit synthetisch erzeugten Datensätzen und den im Rahmen der in Abschnitt 2.2 vorgestellten Experimente erhobenen empirischen Daten durchgeführt. Weitere Studien zur Evaluation des Verfahrens, die qualitativ konsistente Ergebnisse lieferten, finden sich bei Decker (2001) und Wittig und Jameson (2000).

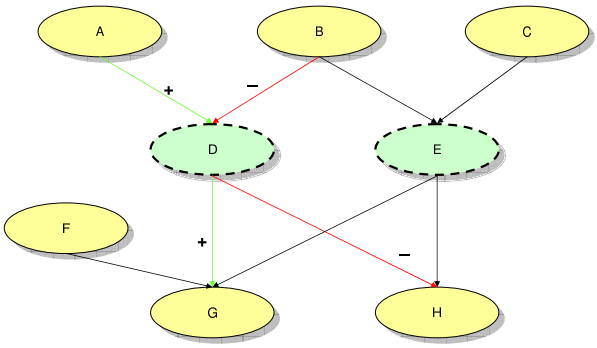
5.3.1 Evaluation mit synthetischen Daten

In einem ersten Schritt wird das Verfahren unter Verwendung synthetisch erzeugter Daten untersucht. Dies hat den Vorteil, dass jeweils das zur Erzeugung der Daten verwendete Bayes'sche Netz—im Folgenden als *Originalnetz* bezeichnet—bekannt ist und somit als Vergleichsmaßstab zur Bewertung herangezogen werden kann. Weiterhin ist es hier möglich, Datensätze beliebiger Größe zu erzeugen und beliebige Variablen als verborgene Variablen zu deklarieren, um das Verfahren mit interessanten Lernaufgaben zu konfrontieren.

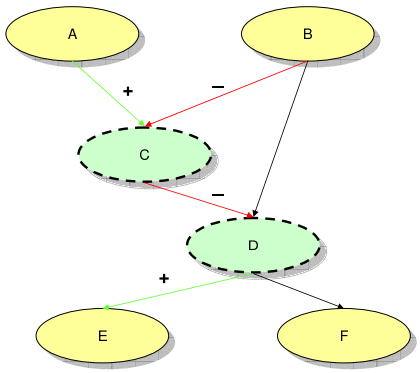
5.3.1.1 Methode

Die Strukturen der beiden zur Evaluation verwendeten Originalnetze sind in Abbildung 5.4 dargestellt. Sie beinhalten jeweils *zwei* verborgene Variablen. Auf eine Präsentation von Ergebnissen des Falls einer verborgenen Variablen wird an dieser Stelle verzichtet, da es sich dabei um eine wesentlich einfachere Lernaufgabe handelt als die vorliegende. Die zu den beiden Strukturen gehörigen CPTs wurden manuell derart spezifiziert, dass die in der Abbildung annotierten qualitativen Einflüsse erfüllt werden. Mit dieser Wahl der Originalnetze werden zwei verschiedene

Strukturfälle betrachtet: das Netz aus Abbildung 5.4 (a) beinhaltet zwei strukturelle *parallel* angeordnete verborgene Variablen, wohingegen Abbildung 5.4 (b) eine Netzstruktur mit *sequentiell* angeordneten verborgenen Variablen enthält. In Fall (b) spielen (direkte) qualitative Einflüsse zwischen den beiden verborgenen Variablen eine Rolle, in (a) liegen solche direkten Einflüsse nicht vor.



(a) Parallele verborgene Variablen



(b) Sequentielle verborgene Variablen

***Abbildung 5.4**: Zur Evaluation des Lernens mit qualitativen Constraints anhand synthetischer Daten verwendete Bayes'sche Netze*

Zur Evaluation des Lernverfahrens der qualitativen Constraints genügt es, im Fall von verborgenen in Kombination mit vollständig beobachteten Variablen, mit solchen relativ einfachen Netzstrukturen zu arbeiten. Für das Lernen der bedingten Wahrscheinlichkeiten, die im Zusammenhang mit den verborgenen Variablen stehen, ist nach dem d-Separationskriterium lediglich die *Markov-Nachbarschaft* der Variablen relevant. Die Markov-Nachbarschaft einer Variablen umfasst neben ihren Eltern, ihre Kinder sowie deren Eltern. Sind die Zustände aller Variablen der Markov-Nachbarschaft bekannt, ist die Variable von den restlichen Variablen des Bayes'schen Netzes unabhängig, d.h., Änderungen in anderen Teilen des Netzes haben keine Auswirkungen auf die Variable. Das bedeutet für den Lernvorgang, dass in Situationen, in denen es nur vollbeobachtbare und verborgene Variablen zu behandeln gilt, die bedingten Wahrscheinlichkeiten zu einem gewissen Grad lokal erlernt werden können. Sind alle Variablen der Markov-Nachbarschaft einer verborgenen Variablen beobachtet, dann können ihre bedingten Wahrscheinlichkeiten lokal nur unter Berücksichtigung der Markov-Nachbarschaft erlernt werden. Gänzlich beobachtete Teile der Netze können lokal mit den in Abschnitt 4.3.1 beschriebenen Methoden bestimmt werden.

Anhand der beiden Originalnetze wurden jeweils vier Datensätze generiert, die nur Werte zu denjenigen Variablen enthielten, die als nicht verborgen für die Lernaufgabe festgelegt wurden. Drei Datensätze bestehend aus 100, 500 bzw. 1000 Fällen wurden als Trainingsmengen benutzt, ein Datensatz mit 10000 Fällen als Testmenge zur Bewertung der erlernten Bayes'schen Netze. Die Testmenge wurde in dieser Größenordnung gewählt, um die Effekte möglicher Zufallsschwankungen, wie sie in kleinen Datensätzen auftreten können, weitestgehend auszuschließen.

Als Bewertungsfunktion wird, wie in der einschlägigen Literatur üblich (und zum Teil aus didaktischen Gründen), die auf der Log-Likelihood basierende *durchschnittliche negative Log-Likelihood pro Testfall* verwendet. Damit repräsentieren geringere Werte gemäß dieser Bewer-

tungsfunktion bessere Resultate und es muss anstelle einer Maximierungs- eine Minimierungsaufgabe gelöst werden —was konzeptionell kein Problem darstellt. Die Durchschnittsbildung über alle Fälle macht einen Vergleich von Bewertungsergebnissen mit unterschiedlich großen Datenmengen möglich.

Zusätzlich werden die Ergebnisse hinsichtlich des Ausmaßes der Verletzungen anhand der in Abschnitt 5.2.2.2 definierten *violation*-Funktion bewertet. Dabei muss beachtet werden, dass die Verletzungen (zum Teil) auf vom Lernverfahren fälschlicherweise vorgenommene Zustandspermutationen zurückzuführen ist (vgl. Abschnitt 5.1). Um dies zu berücksichtigen, wurden die *violation*-Werte der ohne die Spezifikation qualitativer Constraints erlernten Netze zusätzlich als Minimum über alle entsprechenden Zustandspermutationen ermittelt.

In jedem der beiden Strukturfälle wurden zehn Bayes'sche Netze der entsprechenden Struktur mit zufällig gewählten bedingten Wahrscheinlichkeiten θ_{ijk} generiert, die als unterschiedliche Startpunkte des Lernprozesses dienten, um die Effekte zufälliger (un-)günstiger vom Startnetz abhängiger Konstellationen für den Suchprozess bei der Interpretation der Lernergebnisse ausschließen zu können. In beiden Fällen wurde der Lernvorgang der Vergleichbarkeit der Ergebnisse wegen auf 200 bzw. 50 Iterationen des (erweiterten) APN- bzw. (erweiterten) EM-Verfahrens beschränkt. Als feste Schrittweite des APN-Verfahrens wurde $\alpha = 0.03$ gewählt.

Neben der Anzahl der dem Lernverfahren zur Verfügung stehenden Trainingsfälle wurde das Gewicht w variiert (APN-Variante: $w = 2, 4$; EM-Variante: $w = 0.05, 0.25$).[5] Ein Gewicht von 0 repräsentiert das Lernen mit den Standardverfahren ohne die Berücksichtigung qualitativer Einflüsse bzw. Constraints.

5.3.1.2 Ergebnisse nach Beendigung des Lernvorgangs

Ein Überblick der Hauptergebnisse der durchgeführten Analysen wird der Übersichtlichkeit wegen am Beispiel der APN-Variante des Verfahrens der qualitativen Constraints im Fall mit parallel angeordneten verborgenen Variablen gegeben. Dabei liegt der Schwerpunkt auf dem für benutzeradaptive Systeme interessanten Fall weniger verfügbarer Trainingsdaten. Die entsprechende Untersuchung bzw. ein Vergleich mit der EM-Variante sowie eine Betrachtung des zweiten Strukturfalls mit sequentiellen verborgenen Variablen folgen im Anschluss.

Abbildung 5.5 zeigt die Bewertungen der erlernten zehn Netze nach 200 Iterationen bei Einsatz der APN-Variante mit 100 verwendeten Trainingsfällen. Zu jedem der je 10 Startnetze wird das Ergebnis des Standard-APN-Verfahrens und der Variante mit qualitativen Constraints mit den unterschiedlichen Gewichten $w = 2$ und $w = 4$ aufgeführt. Die horizontale Achse des Diagramms stellt die Bewertung des zugehörigen Originalnetzes mit einer durchschnittlichen negativen Log-Likelihood von 6.499 bezüglich der 10000 Testfälle als Vergleichsmaßstab—einer Approximation der bestmöglichen Bewertung—dar.

Erwartungsgemäß werden alle erlernten Netze schlechter (mit größeren durchschnittlichen negativen Log-Likelihood-Werten) als das Originalnetz bewertet, was sich an den nach oben ausgerichteten Balken ablesen lässt, die die entsprechende Abweichung bemessen. Zunehmendes Gewicht w der qualitativen Constraints wirkt sich in zunehmend besseren Bewertungen aus: von einer durchschnittlichen Bewertung über aller zehn Startnetze von 8.020 ohne Verwendung qualitativer

[5] Zu beachten ist dabei die unterschiedliche Rolle des Gewichts bei APN- bzw. EM-Variante als einerseits relatives Gewicht zwischen Likelihood- und *violation*-Gradient bzw. andererseits als maximale Schrittweite zur Verringerung der Constraint-Verletzungen beim hybriden EM-Ansatz.

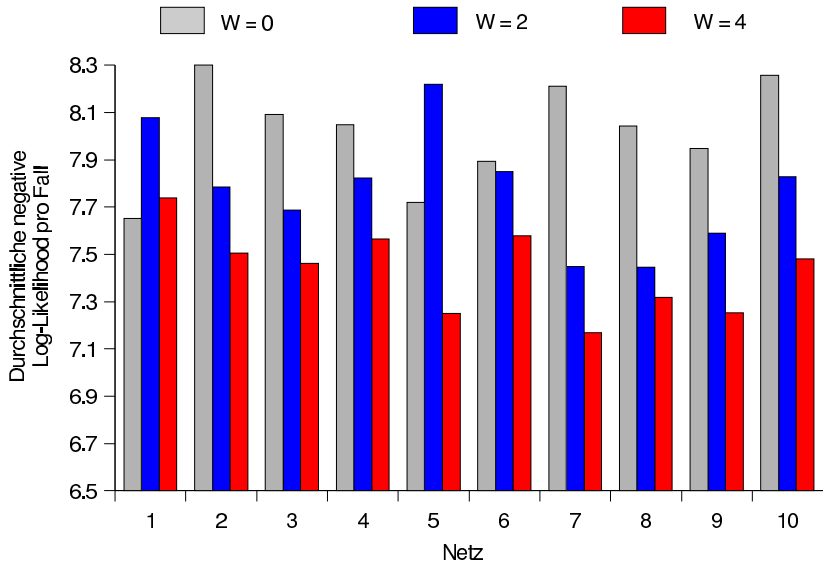


***Abbildung 5.5**: Erzielte Bewertungen des (erweiterten) APN-Verfahrens bei zwei parallel angeordneten verborgenen Variablen mit 100 Trainingsfällen*

Constraints über 7.776 bei einem Constraint-Gewicht von $w = 2$ bis auf einen Durchschnittswert von 7.432 bei $w = 4$. Es gilt zu beachten, dass sich, wenn das Constraint-Gewicht zu hoch gewählt wird und die vorhandenen Trainingsdaten die spezifizierten qualitativen Einflüsse nicht in vollem Umfang unterstützen, dieser Effekt umkehren kann und dadurch (deutlich) schlechtere Bewertungen erzielt werden. Wie bereits in Abschnitt 5.2.4 angesprochen wurde, können Methoden zur automatischen Anpassung des Gewichts in die Verfahren aufgenommen werden.

	$n = 100$	$n = 500$	$n = 1000$
ohne qualitative Constraints	8.020	6.759	6.592
$w = 2$	7.776	6.682	6.576
$w = 4$	7.432	6.675	6.565

***Tabelle 5.1**: Durchschnittlich erzielte Bewertungen des (erweiterten) APN-Verfahrens bei zwei parallel angeordneten verborgenen Variablen mit 100, 500 und 1000 Trainingsfällen*

Tabelle 5.1 zeigt alle durchschnittlich erzielten Bewertungen bei unterschiedlicher Anzahl verwendeter Trainingsfälle. Es zeigt sich, dass sowohl eine Erhöhung des Gewichts w als auch die Verwendung einer größeren Menge an Trainingsdaten—separat als auch in Kombination betrachtet—in einer quantitativen *Verbesserung* der Bewertung resultiert. Die absoluten Veränderungen nehmen dabei mit wachsender Anzahl von Trainingsdaten ab.

Abbildung 5.6 zeigt die Bewertungen der zehn erlernten Netze *bewertet anhand der Trainingsfälle*. In diesem Fall werden die Lernergebnisse besser (d.h., mit kleiner durchschnittlichen negativen Log-Likelihood-Werten) als das Originalnetz auf den Testfällen bewertet—durch die nach unten ausgerichteten Balken erkennbar, die wiederum den absoluten Wert der Abweichung repräsentieren. Diese Bewertungen verschlechtern sich bei zunehmendem Gewicht von durchschnittlich 5.911 über 5.997 auf 6.036. Tabelle 5.2 umfasst die Bewertungen in allen untersuchten Lernsituationen.

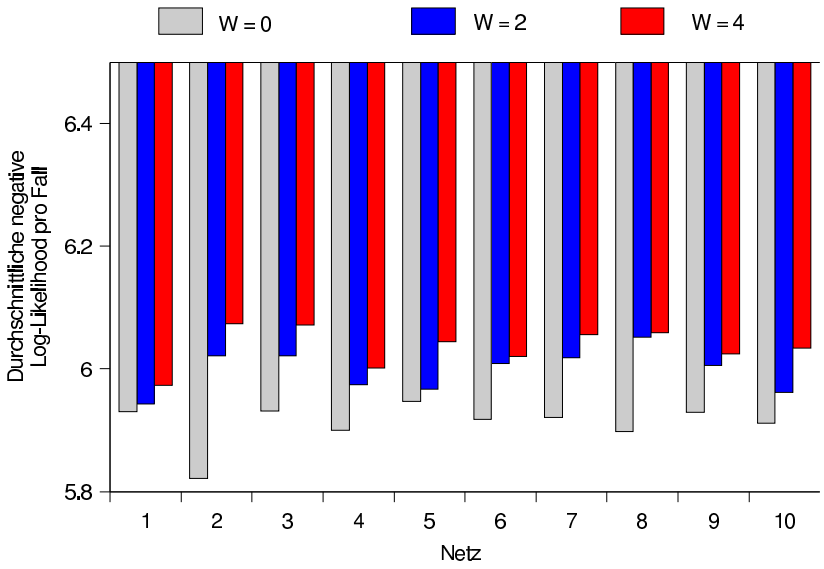


***Abbildung 5.6**: Erzielte Bewertungen des (erweiterten) APN-Verfahrens bei zwei parallel angeordneten verborgenen Variablen mit 100 Trainingsfällen—bewertet anhand der Trainingsdaten*

	$n = 100$	$n = 500$	$n = 1000$
ohne qualitative Constraints	5.911	6.360	6.437
$w = 2$	5.997	6.377	6.442
$w = 4$	6.036	6.387	6.445

***Tabelle 5.2**: Durchschnittlich erzielte Bewertungen des (erweiterten) APN-Verfahrens bei zwei parallel angeordneten verborgenen Variablen mit 100, 500 und 1000 Trainingsfälle—bewertet anhand der Trainingsdaten*

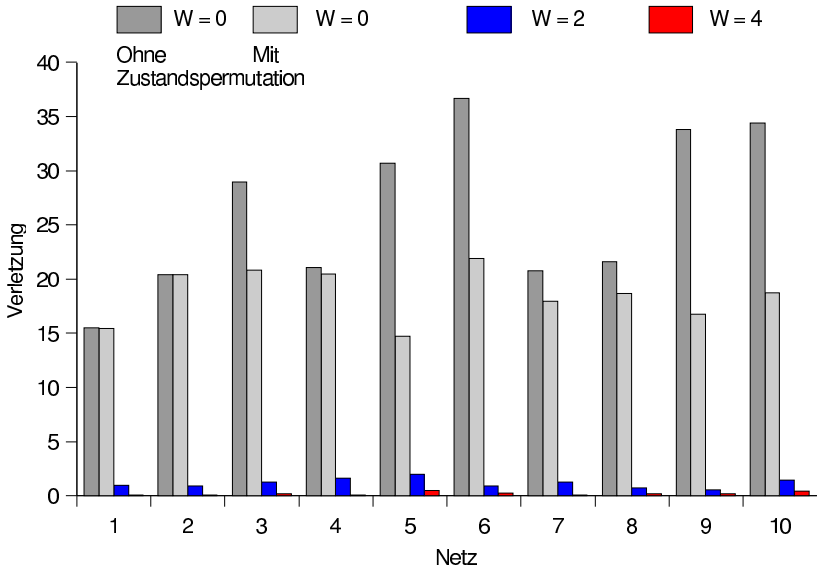


***Abbildung 5.7**: Aufgetretene Verletzungen beim (erweiterten) APN-Verfahren bei zwei parallel angeordneten verborgenen Variablen mit 100 Trainingsfällen*

Sowohl mit zunehmendem Gewicht als mit zunehmender Anzahl an verfügbaren Trainingsfällen *verschlechtern* sich die Bewertungen. Ähnlich wie bei der Bewertung mit den 10 000 Testdaten nimmt das Ausmaß des Effekts bei größerer Anzahl verfügbarer Trainingsfälle ab.

Betrachtet man die Verbesserung bei Bewertung der erlernten Netze unter Verwendung der Testfälle und die Verschlechterung bei Bewertung mit den Trainingsfällen zusammen, so stellt man fest, dass das Lernen mit qualitativen Constraints zu einer *Verringerung des Overfittings* beiträgt. Wie allgemein bekannt und auch hier erwartet, nehmen die Auswirkungen des Overfitting-Effekts bei Verwendung größerer Trainingsmengen ab.

Betrachtet man die in Abbildungen 5.7 dargestellten Verletzungen der durch die spezifizierten qualitativen Einflüsse induzierten Constraints, so lässt sich eine deutliche Verbesserung mit steigendem Gewicht w feststellen. Bei entsprechender Wahl des Gewichts w können Netze erlernt werden, die die vorgegebenen qualitativen Constraints mit einem *violation*-Wert nahe 0 weitestgehend erfüllen. An den beiden links angeordneten Balken dieser Abbildungen wird weiterhin deutlich, dass in der Tat ein gewisser Anteil der Verletzungen auf Permutationen der Zustandskombinationen beruht. Einen Überblick über alle Werte gibt Tabelle 5.3, die die Verbesserung der Ergebnisse durch die qualitativen Constraints des Falls mit 100 Trainingsdaten in den anderen Situationen bestätigen.

	$n = 100$	$n = 500$	$n = 1000$
ohne qualitative Constraints, ohne Zustandspermutation	26.386	24.669	20.023
ohne qualitative Constraints, mit Zustandspermutation	18.592	16.524	11.622
$w = 2$	1.163	3.040	2.821
$w = 4$	0.201	0.772	1.007

***Tabelle 5.3**: Durchschnittlich aufgetretene Verletzungen beim (erweiterten) APN-Verfahren bei zwei parallel angeordneten verborgenen Variablen mit 100, 500 und 1000 Trainingsfällen*

5.3.1.3 Der Verlauf der Lernvorgangs

Um ein besseres Verständnis des Verfahrens des Lernens mit qualitativen Constraints zu erlangen, wird der Verlauf des Lernprozesses näher betrachtet. Dazu werden in den folgenden Abbildungen an den Beispielen der prototypischen Resultate einzelner Startnetze interessante Aspekte der Entwicklung des Lernprozesses beleuchtet. Um dies zu ermöglichen wurden die als Zwischenresultate des Lernens erhaltenen Bayes'schen Netze jeweils mit den (kompletten) Trainings- bzw. Testfällen bewertet.

Betrachtet man die drei oberen Kurven in Abbildung 5.8 bzw. dem Ausschnitt in Abbildung 5.9 so können zwei Phasen des Lernprozesses unterschieden werden. In einer ersten Phase, die in diesem Fall ungefähr die ersten fünf Iterationen des Lernprozesses umfasst, ergeben sich bessere Bewertungen für die (Zwischen-)Netze, die mit den Standardverfahren (ohne die Berücksichtigung qualitativer Constraints) erlernt wurden. Danach wird das entgegengesetzte Verhalten beobachtet: Die Ergebnisse verbessern sich mit zunehmendem Constraint-Gewicht. Diese Zweiteilung des Lernprozesses lässt sich erklären, wenn man berücksichtigt, dass sich die Suchprozedur in der initialen Phase typischerweise noch in einer Region befindet, die ein großes Ausmaß an Constraint-Verletzungen aufweist. Deshalb versuchen die Verfahren der qualitativen Constraints in den ersten Schritten diese Verletzungen zu eliminieren oder zumindest zu vermindern. Die Standardverfahren fokussieren sich hingegen bereits zu Beginn auf die Verbesserung der Log-Likelihood und erlan-

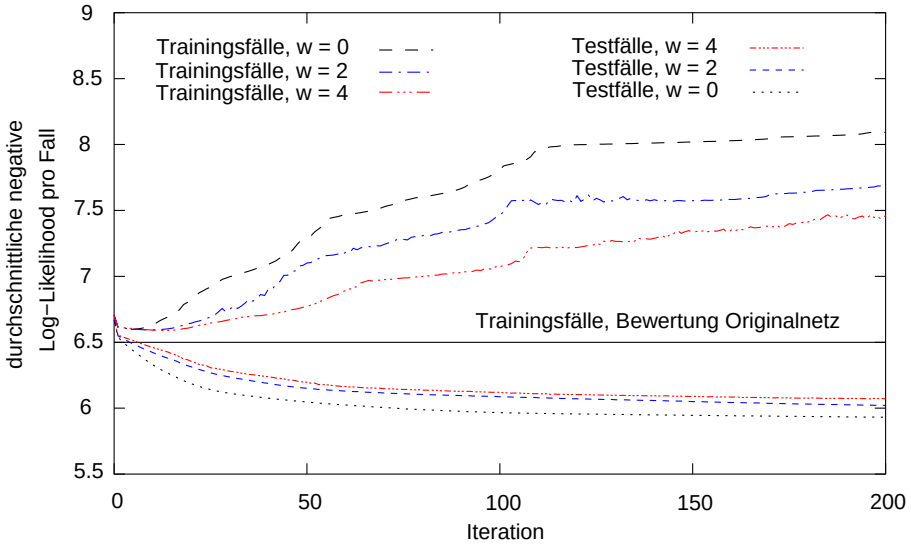


***Abbildung 5.8**: Prototypischer Verlauf des Lernprozesses des (erweiterten) APN-Verfahrens bei zwei parallel angeordneten verborgenen Variablen mit 100 Trainingsfällen*
(Die in dieser und den folgenden Abbildungen präsentierten Ergebnisse wurden mit dem dritten Startnetz erzielt.)

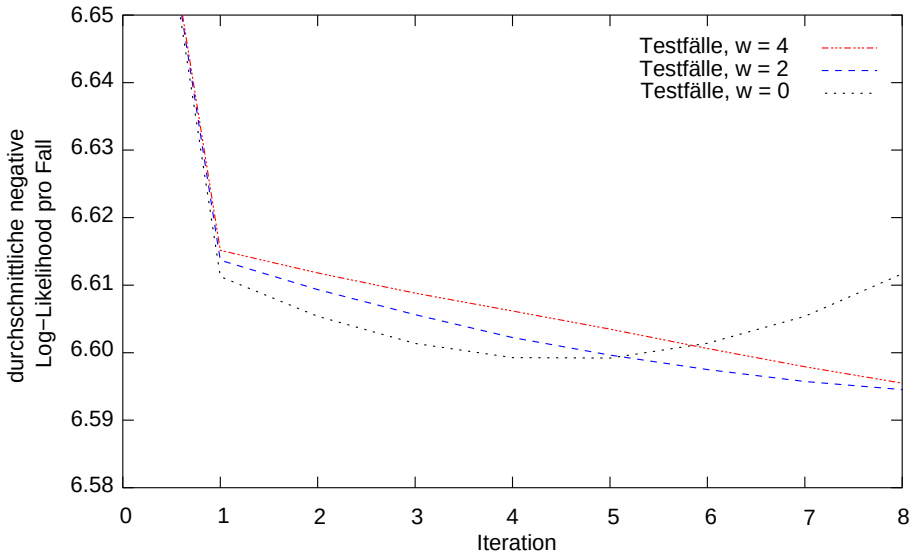


***Abbildung 5.9**: Die ersten acht Iterationen aus Abbildung 5.8*

gen somit diesbezüglich einen Vorteil. Sind Regionen im Suchraum erreicht, wo die Constraints weitestgehend erfüllt sind, so arbeiten auch die erweiterten Verfahren verstärkt—oder im Idealfall ausschließlich—an der Verbesserung der Log-Likelihood der Daten.

Die beste Bewertung eines (Zwischen-)Ergebnisses wird zu einem recht frühen Zeitpunkt im Lernprozess erreicht (hier je nach Wahl des Constraint-Gewichts zwischen zehn und 15 Iterationen). Um das bestmögliche Ergebnis bezüglich der numerischen Genauigkeit des erlernten Bayes'schen Netzes zu erreichen, müsste man den Lernvorgang zu diesem Zeitpunkt terminieren. Da es in der Praxis nicht möglich ist diesen optimalen Zeitpunkt exakt vorherzusagen, liegt einer der Hauptbeiträge des Lernens mit qualitativen Constraints in der Eigenschaft, die Lernkurven „flach" zu halten, d.h., keine Verschlechterungen der numerischen Genauigkeit zuzulassen, so dass ein etwas zu spät gewählter Terminationszeitpunkt keine (deutlich) schlechteren Ergebnisse liefert.

Die drei unteren Kurven in Abbildung 5.8 stellen die entsprechenden Bewertungen mit den Trainingsfällen dar. An diesen Kurven wird das zunehmende Overfitting, das in schlechteren Bewertungen der Testfälle resultiert und zum Teil durch das Verfahren der qualitativen Constraints vermindert werden kann, an den zunehmend kleineren negativen Log-Likelihood-Werten deutlich.

Abbildung 5.10 und 5.11 zeigen die unterschiedlich starken Auswirkungen des Einbringens der qualitativen Einflüsse in den Lernprozess in Abhängigkeit von der verfügbaren Menge an Trainingsdaten. Durch das weniger stark ausgeprägte Overfitting bei größeren Anzahlen an Trainingsdaten, spielen die qualitativen Constraints für eine potenzielle Verbesserung der Resultate hinsichtlich der Log-Likelihood eine geringere Rolle. Das Verfahren der qualitativen Constraints ist insbesondere im Fall weniger verfügbarer Trainingsdaten von Bedeutung, welcher in benutzeradaptiven Systemen von großem Interesse ist.

Neben der Verringerung des Overfitting-Effekts spielt die Verbesserung der Interpretierbarkeit der Lernresultate im Kontext benutzeradaptiver Systeme eine wichtige Rolle. Abbildungen 5.12 und 5.13 zeigen den prototypischen Verlauf der *violation*-Werte im Lernprozess. Man beobachtet, dass sich die Reduktion der Constraint-Verletzungen auf die initiale Phase des Lernvorgangs konzentriert. Später werden durch das Verfahren weitestgehend keine neuen, zusätzlichen Verletzungen mehr „erlaubt".

Ein höheres Gewicht spiegelt sich in einer beschleunigten Elimination der Verletzungen wider: Bei einem Gewicht von 4.0 wird das Minimum der *violation*-Funktion bereits nach ca. 23 Iterationen erreicht, wohingegen dies bei einem Gewicht von 2.0 erst nach ca. 40 Iterationen der Fall ist. Der leichte Anstieg der *violation*-Werte im weiteren Verlauf des Lernens im ersten Fall resultiert aus einer in Relation betrachtet stärkeren Verbesserung des Log-Likelihood-Anteils der erweiterten Bewertungsfunktion, so dass die Gesamtbewertung sich dennoch verbessert (vgl. hierzu auch Decker, 2001).

An Abbildung 5.13 wird deutlich, dass das Verfahren der qualitativen Constraints gerade in Lernsituationen mit wenigen Trainingsdaten in der Lage ist, Bayes'sche Netze zu erlernen, die weitestgehend die postulierten qualitativen Zusammenhänge erfüllen, d.h., die interpretierbare CPTs besitzen. Dies hängt mit der Beobachtung zusammen, dass sich der Suchraum in solchen Situationen durch eine Vielzahl lokaler Optima auszeichnet und somit eine größere Wahrscheinlichkeit vorliegt, dass der Lernprozess ohne die Vorgabe qualitativer Constraints in einem „nicht interpretierbaren" Optimum terminiert.

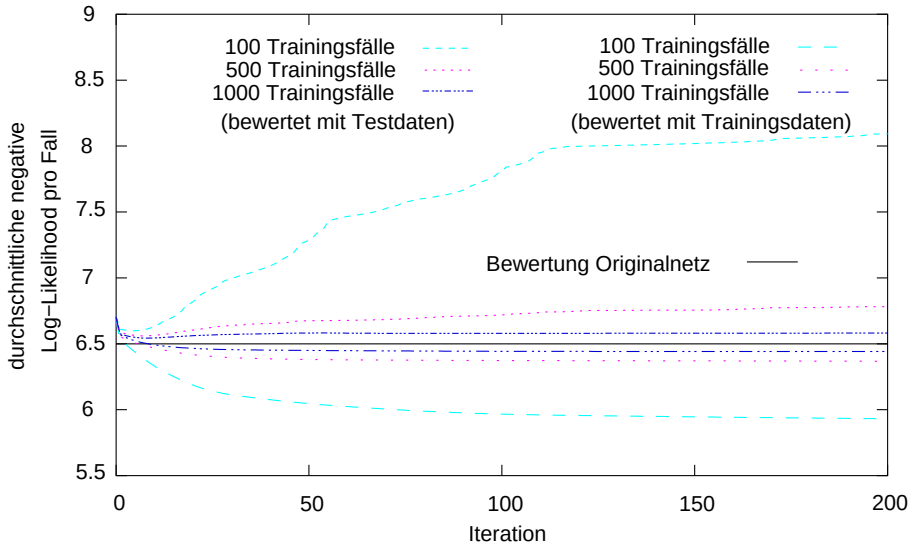


***Abbildung 5.10**: Prototypischer Verlauf des Lernprozesses des Standard-APN-Verfahrens bei zwei parallel angeordneten verborgenen Variablen mit unterschiedlichen Trainingsmengen (ohne qualitative Constraints)*

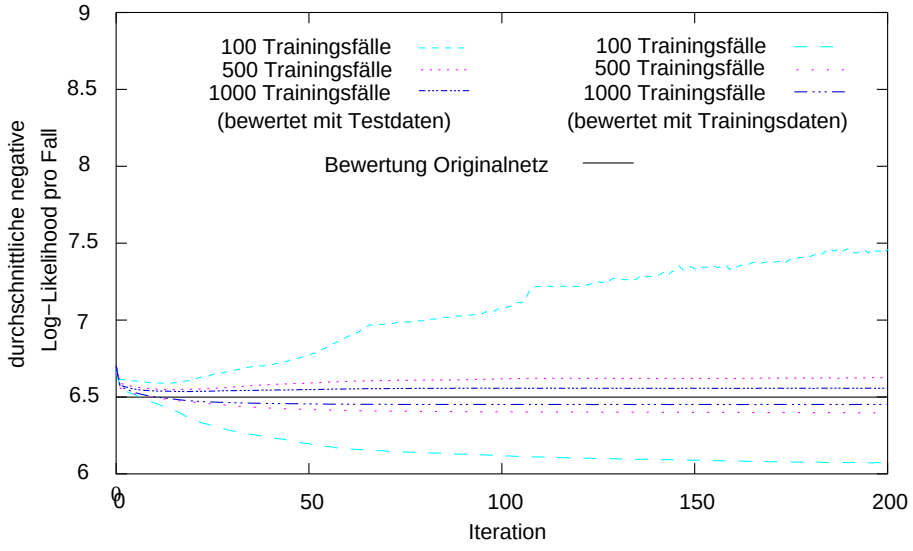


***Abbildung 5.11**: Prototypischer Verlauf des Lernprozesses des erweiterten APN-Verfahrens mit qualitativen Constraints ($w = 4$) bei zwei parallel angeordneten verborgenen Variablen mit unterschiedlichen Trainingsmengen*

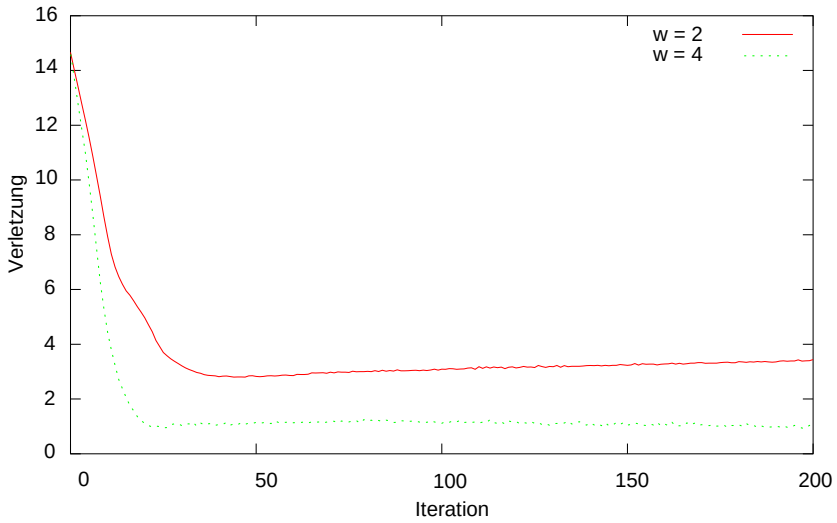


***Abbildung 5.12**: Prototypische Entwicklung der Verletzungen des (erweiterten) APN-Verfahrens bei zwei parallel angeordneten verborgenen Variablen mit 1000 Trainingsfällen mit unterschiedlichen Constraint-Gewichten*

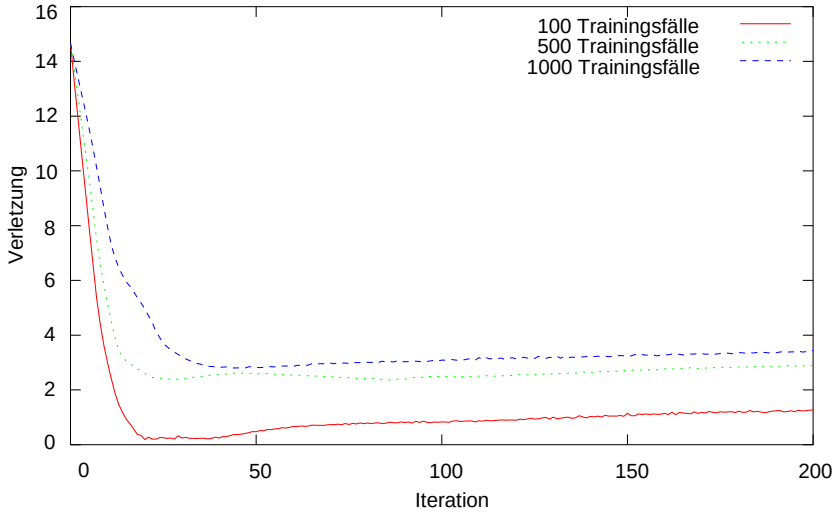


***Abbildung 5.13**: Prototypische Entwicklung der Verletzungen des (erweiterten) APN-Verfahrens bei zwei parallel angeordneten verborgenen Variablen bei einem Constraint-Gewicht von $w = 2$ mit unterschiedlichen Trainingsmengen*

5.3.1.4 Überblick der Ergebnisse verschiedener Lernaufgaben

Die Tabellen 5.4, 5.5, 5.6 und 5.7 fassen die Ergebnisse weiterer Analysereihen bei Variation der untersuchten Struktur und der eingesetzen Variante des Verfahrens der qualitativen Constraints im für die vorliegende Arbeit interessantesten Fall weniger verfügbarer Trainingsdaten ($n = 100$) zusammen.

Bewertung Originalnetz: 6.499

	Testfälle	Lernfälle	Verletzungen (ohne Permutation)
ohne qualitative Constraints	8.020	5.911	18.592 (26.386)
$w = 2$	7.776	5.997	1.163
$w = 4$	7.432	6.036	0.201

***Tabelle 5.4**: Übersicht: Durchschnittlich erzielte Ergebnisse der APN-Variante bei zwei parallel angeordneten verborgenen Variablen*

Bewertung Originalnetz: 6.499

	Testfälle	Lernfälle	Verletzungen (ohne Permutation)
ohne qualitative Constraints	8.616	5.873	21.748 (28.142)
$w = 0.05$	8.001	5.897	22.479
$w = 0.25$	7.825	5.963	12.472

***Tabelle 5.5**: Übersicht: Durchschnittlich erzielte Ergebnisse der EM-Variante bei zwei parallel angeordneten verborgenen Variablen*

Bewertung Originalnetz: 4.056

	Testfälle	Lernfälle	Verletzungen (ohne Permutation)
ohne qualitative Constraints	4.407	3.884	3.617 (7.265)
$w = 2$	4.398	3.898	0.869
$w = 4$	4.385	3.904	0.106

***Tabelle 5.6**: Übersicht: Durchschnittlich erzielte Ergebnisse der APN-Variante bei zwei sequentiell angeordneten verborgenen Variablen*

Diese Ergebnisse bestätigen die im vorigen Abschnitt präsentierten Resultate der APN-Variante angewendet auf das Bayes'sche Netze mit zwei strukturell parallel angeordneten verborgenen Variablen in den anderen Lernsituationen.[6] Allerdings hat die EM-Variante im Fall der parallel angeordneten verborgenen Variablen Schwierigkeiten, die Constraint-Verletzungen möglichst vollständig zu eliminieren—dennoch kann sie eine deutliche Reduktion erzielen, die aber im Vergleich zu den anderen Fällen wesentlich geringer ausfällt. Dies deutet darauf hin, dass hinsichtlich der EM-Variante Lernsituationen auftreten können, in denen sich der hybride Charakter des

[6] Es existiert mit der Verschlechterung der Bewertung beim Übergang vom Gewicht 0.05 auf 0.25 bei der EM-Variante angewendet auf das Netz mit strukturell sequentiell angeordneten verborgenen Variable nur eine einzige Ausnahme.

Bewertung Originalnetz: 4.056

	Testfälle	Lernfälle	Verletzungen (ohne Permutation)
ohne qualitative Constraints	4.550	3.856	4.979 (8.283)
$w = 0.05$	4.421	3.867	2.878
$w = 0.25$	4.468	3.885	0.038

***Tabelle 5.7**: Übersicht: Durchschnittlich erzielte Ergebnisse der EM-Variante bei zwei sequentiell angeordneten verborgenen Variablen*

Verfahrens in einer verminderten Performanz auswirkt.[7] Die beiden Teilziele—Verbesserung der Likelihood der Daten und der Elimination der Constraints—können dann nicht ohne sich negativ auswirkende Interaktionen bearbeitet werden. Dies erscheint insbesondere bei parallel angeordneten verborgenen Variablen problematisch, da hier keine direkten (qualitativen) Beziehungen zwischen den verborgenen Variablen existieren, die die gegenseitige „Abstimmung“ der den beiden Variablen zugeordneten Teillernprobleme erleichtert.

Ein Vergleich der beiden Verfahrensvarianten—APN und EM—macht anhand der absoluten Werte der Tabellen aufgrund des unterschiedlichen Charakters der Verfahren und den daraus resultierenden Verhaltensweisen sowie den verschiedenen Parametern wie der Schrittweite des APN-Algorithmus und den beiden Interpretationen des Constraint-Gewichts keinen Sinn. Qualitativ betrachtet verhalten sich die beiden Alternativen gleich, d.h., die qualitativen Ergebnisse der Diskussion des Verlaufs des Lernprozesses (Existenz zweier Phasen, Verminderung des Overfittings, Elimination der Verletzungen) sind auf die EM-Variante übertragbar.

5.3.2 Evaluation mit empirischen Daten

Nach der ausführlichen Untersuchung der Eigenschaften und Wirkungsweise des Verfahrens der qualitativen Constraints anhand synthetisch erzeugter Daten soll in diesem Abschnitt überprüft werden, ob die angestrebten und in der Tat identifizierten Vorzüge der Methode auch unter „realen“ Bedingungen mit „echten“ empirischen Daten zu beobachten sind. Dazu werden die im Rahmen der in Abschnitt 2.2 beschriebenen Experimente erhobenen empirischen Daten verwendet.

Es gilt zu berücksichtigen, dass es sich hierbei im Vergleich zur Situation mit synthetisch erzeugten Daten um eine anspruchsvollere Aufgabenstellung handelt. Das Originalnetz ist nicht bekannt, es kann lediglich ein Modell in Form der Struktur des Bayes’schen Netzes zugrunde gelegt werden, dass auf der Basis psychologischer Erkenntnisse und Annahmen fußt. Es gibt keine Garantie, dass sich dabei um das „Originalnetz“ handelt, dass die erhobenen Daten „erzeugt“ hat. Eng verwandt damit ist die Unsicherheit, ob postulierte und für den Lernprozess vorgegebene qualitative Einflüsse bzw. Constraints in der Tat Gültigkeit besitzen und somit auch in den gesammelten Daten repräsentiert werden, wie es bei synthetischen Daten der Fall ist. Dies sind Schwierigkeiten mit denen das Verfahren in der Mehrzahl der potenziellen Anwendungssituationen typischerweise konfrontiert wird, und deren adäquate Behandlung die Voraussetzung eines erfolgreichen Einsatzes in der Praxis darstellt.

[7] Entsprechende Beobachtungen wurden auch im Rahmen der Untersuchungen von Decker (2001) sowie Wittig und Jameson (2000) gemacht.

5.3.2.1 Methode

Abbildung 5.2 zeigt die für die beiden Experimente verwendeten Netzstrukturen inklusive der angenommenen qualitativen Einflüsse zwischen den Variablen. Dabei ist zu beachten, dass man erwartet, dass aufgrund der komplexeren Strukturen, die über die Markov-Nachbarschaft der verborgenen Variablen hinausgehen, die Ergebnisse nicht so deutlich zu Tage treten werden wie bei den einfacheren, im vorigen Abschnitt verwendeten Strukturen. Dies ist in der Tatsache begründet, dass es sich bei der Log-Likelihood um eine *globale* Bewertungsfunktion handelt, die das komplette Netz berücksichtigt und sich nicht auf einzelne Teile fokussiert, die mit den verborgenen Variablen und den zugehörigen qualitativen Einflüssen in Zusammenhang stehen.

Um die Situation, in der nur eine begrenzte Menge an Trainingsdaten zur Verfügung steht, zu simulieren, wurden zwei Varianten der Untersuchung durchgeführt, einerseits unter Verwendung der *kompletten* Datenmenge und andererseits der Verwendung einer *begrenzten* Menge an Daten:

- *Komplette Trainingsdaten:* Für jede Versuchsperson wurde mit allen Daten der anderen Versuchspersonen ein Bayes'sches Netz erlernt.

- *Begrenzte Trainingsdaten:* Für jede Versuchsperson wurde nur eine zufällige Teilmenge bestehend aus 30% der Daten der jeweils verbleibenden Versuchspersonen zum Lernen verwendet. Dieser Fall ist, wie die Ergebnisse mit synthetisch erzeugten Daten bereits andeuten, von besonderem Interesse—gerade auch in benutzeradaptiven Systemen—, da hier das Einbringen der qualitativen Information in den Lernprozess von besonderem Nutzen erscheint.

Beim Berechnen der Bewertung der erzielten Lernergebnisse wurde jeweils der komplette von einer Versuchsperson produzierte Datensatz verwendet.

Ansonsten handelte es sich um die gleiche Vorgehensweise wie im vorhergehenden Abschnitt, die in eine 24- bzw. 32fache Kreuzvalidierung eingebettet war. Die im Folgenden vorgestellten Ergebnisse für die je zehn zufälligen Startnetze stellen die entsprechenden Durchschnittswerte der 24 bzw. 32 Kombinationen von Trainings- und Testdatenmengen dar.

Es werden die detaillierten Ergebnisse der EM-Variante des Verfahrens präsentiert, da einerseits die Analysen mit synthetischen Daten darauf hindeuten, dass diese Methode hinsichtlich der Log-Likelihood und den *violation*-Werten wegen ihres hybriden Charakters den kritischen der beiden Fälle darstellt und andererseits dort bereits ausführlich Ergebniskurven der APN-Variante verwendet wurden. Es wurde aufgrund der Erfahrungen mit mehreren Studien in diesem Szenario ein Constraint-Gewicht von $w = 0.3$ gewählt.

5.3.2.2 Wenige Lerndaten

Abbildung 5.14 zeigt die Ergebnisse der Lernaufgabe mit der begrenzten Anzahl an Trainingsdaten. Man sieht, dass die gleichen qualitativen Effekte wie in den Analysen mit synthetischen Daten erzielt werden—wenn auch aus den angesprochenen Gründen in weniger ausgeprägter Form:[8] Verletzungen der vorgegebenen qualitativen Constraints werden deutlich reduziert (von durchschnittlich 10.004 auf 1.612 im Flughafenexperiment und von 7.560 auf 2.249 im Fall des Anweisungsexperiments) und das Ausmaß des Overfittings wird verringert.

[8] Da das Originalnetz in dieser Studie nicht bekannt ist, kann auch kein entsprechender Vergleichsmaßstab in den Graphen aufgetragen werden.

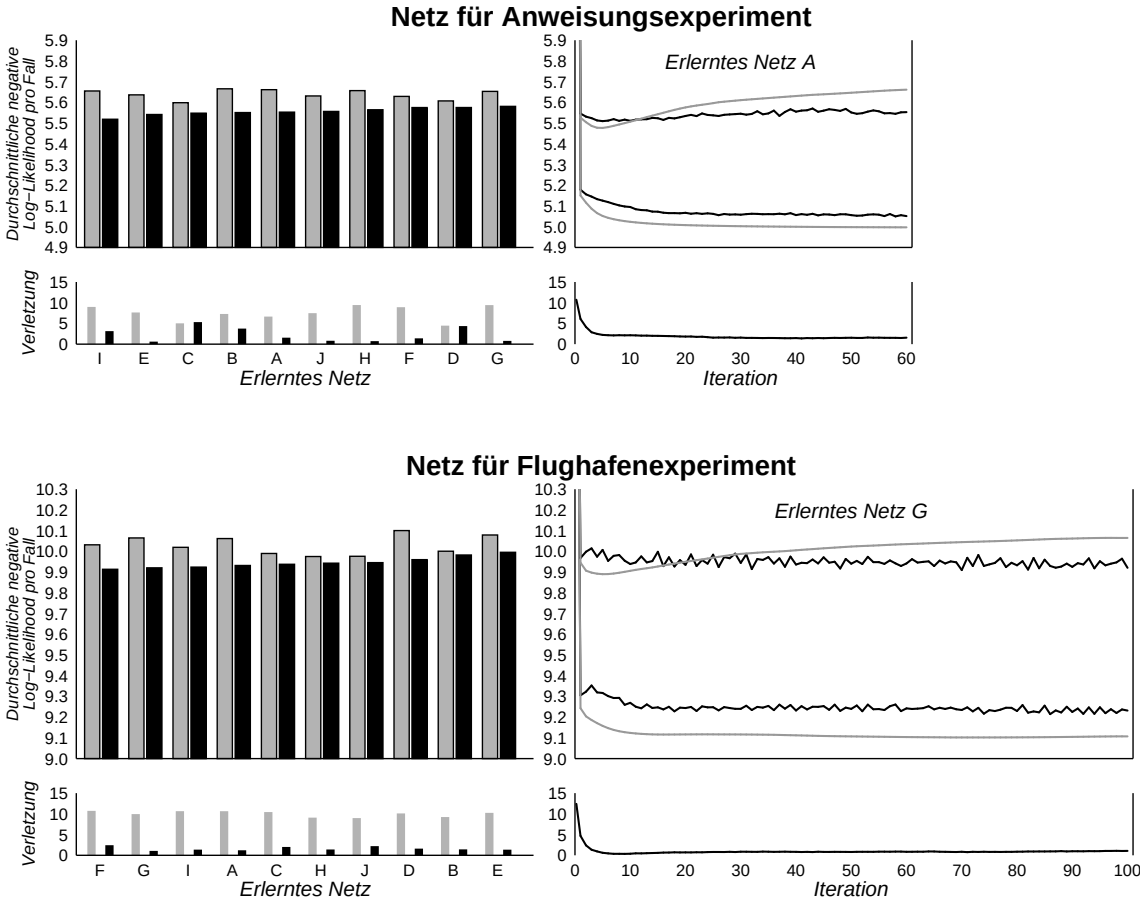


Abbildung 5.14*: Ergebnisse des Lernens interpretierbarer CPTs mit qualitativen Constraints anhand empirischer experimenteller Daten*

5.3.2.3 Zusammenfassung der Ergebnisse bei mehr Lerndaten

Wird die gleiche Studie mit dem kompletten (Trainings-)Datensatz durchgeführt, so beobachtet man hinsichtlich der Elimination der Verletzungen ein ähnliches Ergebnis: Die durchschnittliche Verletzung wird von 6.714 beim Anweisungsexperiment bzw. 9.937 beim Flughafenexperiment auf 1.253 bzw. 0.689 reduziert.

Im Gegensatz dazu verschwindet der Vorteil des Lernens mit Constraints bezüglich der Log-Likelihood fast vollständig: Die Durchschnittswerte sind 5.365 bzw. 9,841 beim Lernen ohne Constraints und 5.368 bzw. 9.823 mit Constraints.

Diese Ergebnisse deuten wie bei der Analyse mit synthetischen Daten darauf hin, dass wenn genügend[9] Trainingsdaten vorhanden sind, um ein Overfitting weitestgehend zu vermeiden, sich der Mehrwert des Lernens mit qualitativen Constraints auf die (Verbesserung der) Interpretierbarkeit der erlernten Bayes'schen Netze beschränkt. Außerdem wird auch hier deutlich, dass dieser Vorteil nicht auf Kosten einer verminderten Genauigkeit erkauft wird.

[9] Die Anzahl der benötigten Trainingsfälle ist u.a. abhängig von der Anzahl der verborgenen Variablen, der Anzahlen der Zustände der Variablen und der Struktur des Netzes.

5.3.3 Lernen ohne Daten

Eine weitere, etwas weniger typische Anwendung des Verfahrens des Lernens mit qualitativen Constraints kann in in Situationen sinnvoll sein, wenn überhaupt keine empirischen Daten für den Lernprozess verfügbar sind: Es ist dann immer noch möglich die qualitativen Einflüsse zu spezifizieren und das Lernverfahren ohne Daten arbeiten zu lassen. Das Verfahren arbeitet dann solange an einer Verringerung der Constraint-Verletzungen, bis ein Punkt im Suchraum erreicht wird, an dem alle Constraints erfüllt sind. Dort terminiert das Verfahren mit einem Bayes'schen Netz, das die spezifizierten qualitativen Einflüsse beachtet.

Dieses Netz kann dann beispielsweise der Ausgangspunkt einer weiteren Modelladaption sein—möglicherweise anhand von Daten, die nur im Laufzeitbetrieb eines Systems erhebbar sind.

Diese Vorgehensweise stellt eine Alternative zur manuellen Konstruktion eines die qualitativen Einflüsse beachtenden Bayes'schen Netzes dar, einer nicht trivialen Aufgabe, die durch die potenziell große Anzahl an zu spezifizierenden bedingten Wahrscheinlichkeiten in vielen Fällen sehr bzw. zu komplex ist.

5.4 Zusammenfassung

In diesem Kapitel wurde eine Konzeptualisierung des Einbringens qualitativer Informationen in den Lernprozess der bedingten Wahrscheinlichkeiten eines Bayes'schen Netzes anhand vorhandener Trainingsdaten vorgestellt. Dazu wurde—basierend auf der Arbeit von Druzdzel und van der Gaag (1995)—eine Definition eines quantitativen Index des Ausmaßes von Verletzungen postulierter qualitativer Einflüsse gegeben. Es wurde gezeigt, wie existierende Standardlernverfahren (wie APN und EM) in entsprechender Weise modifiziert werden können, um interpretierbare CPTs zu erlernen. Anhand unterschiedlicher Strukturfälle mit synthetisch erzeugten sowie empirisch erhobenen Daten wurden das entwickelte Verfahren des Lernens mit qualitativen Constraints evaluiert.

Betrachtet man die Gesamtheit der vorgestellten Ergebnisse, so lässt sich zusammenfassen, dass die beiden mit der Entwicklung des Verfahrens des Lernens mit qualitativen Constraints verfolgten Ziele erreicht wurden:

- *Verbesserung der Modellqualität durch Elimination bzw. Verringerung des Overfittings:* Durch das Einbringen von zusätzlichem Wissen in Form der qualitativen Einflüsse in den Lernvorgang können einige der schlechteren lokalen Optima des hochdimensionalen Suchraums vermieden werden. Dieser Effekt ist besonders ausgeprägt im Fall weniger verfügbarer Trainingsdaten. Es ist bekannt, dass ein Teil des Overfittings im MAP-Ansatz bereits durch die Spezifikation einer uninformierten A-Priori-Wahrscheinlichkeitsverteilung vermindert werden kann. Das zweite verfolgte Ziel kann damit jedoch nicht gelöst werden.

- *Erhöhung der Interpretierbarkeit der erlernten Bayes'schen Netze:* Die präsentierten Ergebnisse zeigen, dass es bei entsprechender Parameterwahl in vielen Fällen möglich ist, Bayes'sche Netze zu erlernen, die die postulierten qualitativen Einflüsse zwischen den Variablen modellieren. Dieses Resultat ist nicht auf den Fall weniger für den Lernprozess verfügbarer Trainingsdaten beschränkt.

Die Kombination der beiden Hauptergebnisse erhöhen das Potenzial maschineller Lernverfahren für Bayes'sche Netze für einen Einsatz in benutzeradaptiven Systemen, da sie in diesem Kontext wichtige Problemstellungen behandeln. Die Erhöhung der Interpretierbarkeit der erlernten Bayes'schen Netze alleine betrachtet stellt in diesem Zusammenhang bereits einen Fortschritt dar.

6 Alternative nicht-strukturelle Adaptionsmethoden Bayes'scher Netze

Thema dieses Kapitels sind alternative Methoden der Laufzeit-Adaption von Benutzermodellen in Form Bayes'scher Netze an den individuellen Benutzer. Damit wird ein weiterer wichtiger Bestandteil der Konzeption des maschinellen Lernens Bayes'scher Netze für benutzeradaptive Systeme behandelt (siehe Abbildung 6.1).

Insbesondere wird das im Rahmen dieser Arbeit entwickelte Verfahren der *differentiellen Adaption* (Jameson & Wittig, 2001) vorgestellt und anhand empirischer Daten im Vergleich mit alternativen Ansätzen evaluiert. Die Methode der differentiellen Adaption ist aus der Benutzermodellierung heraus motiviert und kann die entsprechenden Probleme in vielen Szenarien benutzeradaptiver Systeme besser als die existierenden allgemeinen Adaptionsmethoden behandeln.

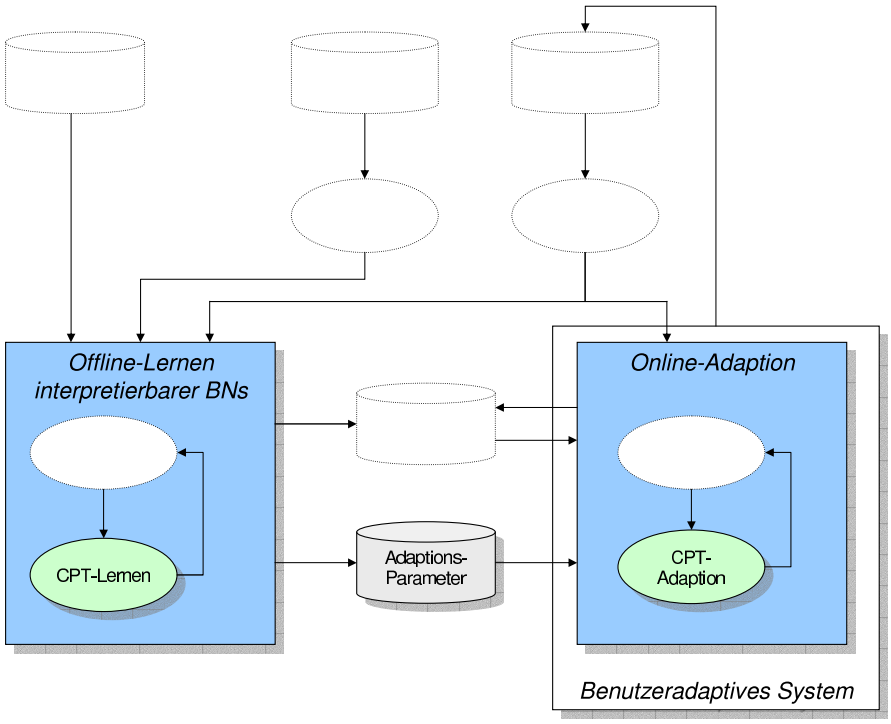


***Abbildung 6.1**: Einordnung der nicht-strukturellen Adaption in die integrative Konzeption*

6.1 Motivation: Inter-individuelle Unterschiede zwischen Benutzern

Dieses Kapitel beschäftigt sich mit dem essentiellen Problem des maschinellen Lernens in benutzeradaptiven Systemen: der Erkennung und adäquaten Behandlung individueller Unterschiede zwischen den einzelnen Benutzern—wie in Abschnitt 3.1.3.2 in allgemeiner Weise diskutiert.

In der Praxis werden bisher im Wesentlichen zwei Alternativen betrachtet: (a) der Einsatz eines anhand einer Trainingsmenge von Daten zu einer Vielzahl von Benutzern erlernten *allgemeinen* Benutzermodells und (b) der Einsatz eines *individuellen* Benutzermodells, das nur auf Daten eines einzelnen—des aktuell mit dem System interagierenden—Benutzers basiert. Dies spiegelt sich auch in der Auswahl der benutzeradaptiven Systeme in Abschnitt 2.6, die maschinelle Lernverfahren Bayes'scher Netze verwenden, wider. Die nahe liegende Kombination der beiden Varianten—das Erlernen eines allgemeinen Modells, das anschließend anhand von Interaktionsdaten des aktuellen Benutzers an diesen individuell angepasst wird—spielt in den dem Autor bekannten, auf Bayes'schen Netzen basierenden Systemen, meist keine Rolle. Es existieren zwar Methoden (teilweise mit Ad-hoc-Charakter), die eine Adaption eines Bayes'schen Netzes ermöglichen, die aber nicht spezifisch auf den Benutzermodellierungskontext zugeschnitten sind, und dementsprechend nicht in der Lage sind, die komplette, potenziell verfügbare Informationsmenge auszunutzen. Im Folgenden wird im Rahmen eines Vergleichs alternativer Adaptionsansätze ein Verfahren vorgestellt, dessen Ziel es ist, automatisch individuelle Unterschiede zwischen den Benutzer zu erkennen und davon im Rahmen des Adaptionsprozesses zu profitieren, um eine möglichst optimale und schnelle Anpassung des Modells an den individuellen Benutzer zu gewährleisten.

Die entscheidende Frage, die es diesbezüglich zu klären gilt, lautet:

> *Wie soll das erlernte allgemeine Bayes'sche Netz an den neuen Benutzer angepasst werden?*

Oder alternativ formuliert:

> *Wie soll das allgemeine Netz gegenüber den neu gesammelten Interaktionsdaten gewichtet werden bzw. wie schnell soll das allgemeine Bayes'sche Netz anhand der neuen Daten an den neuen Benutzer angepasst werden, um dessen individuelle Eigenschaften im Modell zu erfassen?*

Die dem Verfahren zugrunde liegende Idee ist es, anhand der verfügbaren Daten, die Unterschiede zwischen den einzelnen Benutzern innerhalb eines Offline-Arbeitsschritts, der auch zum Erlernen des allgemeinen Bayes'schen Netzes dient, zu ermitteln. Des Weiteren werden Parameter für den Online-Part der Adaption festgelegt, die spezifizieren, wie schnell die unterschiedlichen Aspekte des Benutzermodells adaptiert werden sollen. Der Vorteil dieses Verfahrens im Vergleich zu existierenden CPT-Adaptionsalgorithmen Bayes'scher Netze besteht darin, dass im Rahmen des Offline-Teils automatisch unterschiedliche Adaptionsgeschwindigkeiten für unterschiedliche Teile des Bayes'schen Netzes gelernt werden.

Im Rahmen eines Vergleichs mit alternativen Möglichkeiten wird das Verfahren mit den empirischen Daten des Anweisungs- und Flughafenexperiments evaluiert. Daneben gilt es, Anforderungen der Domäne bzw. Einschränkungen des Einsatzes der betrachteten Verfahren im Konstruktionsprozess eines Systems zu analysieren und zu bewerten, um eine für die vorliegende Situation geeignete Methode auszuwählen. Jedes der Verfahren besitzt Vor- und Nachteile, die keines als unbestrittene Standardlösung für alle potenziellen Einsatzszenarien erscheinen lassen.

6.2 Alternative Verfahren der Adaption

Bevor das neu entwickelte Verfahren beschrieben wird, werden im Folgenden Alternativen der Adaption von Benutzermodellen erläutert, die ohne Modifikationen der Standardlern- bzw. Adaptionsverfahren für Bayes'sche Netze auskommen. Diese Methoden können weitestgehend als direkte Anwendung der vorhandenen Algorithmen im Kontext benutzeradaptiver Systeme angesehen werden. Das im Anschluss erläuterte, im Rahmen dieser Arbeit neu entwickelte Verfahren versucht, einige der Schwächen der existierende Ansätze im Benutzermodellierungskontext zu beheben.

Beginnend mit den beiden Endpunkten des Spektrums der Adaptionsmöglichkeiten—die individuellen bzw. allgemeinen Benutzermodelle—werden danach zwei Methoden vorgestellt, die versuchen die Vorteile der beiden extremen Verfahren—soweit durch die empirischen Daten unterstützt—zu kombinieren.

Individuelles Benutzermodell Das rein *individuelle* als Bayes'sches Netz modellierte Benutzermodell wird nur anhand der Daten des einzelnen, aktuellen Benutzers erstellt. Dabei wird angenommen, dass zu Beginn keine bereits erhobenen Daten zu diesem Benutzer vorliegen. Entsprechend wird das Modell mit gleichverteilten bedingten Wahrscheinlichkeiten θ_{ijk} in den CPTs initialisiert. Die Adaption erfolgt unter Verwendung des in Abschnitt 4.5.1 beschrieben AHUGIN-Algorithmuses. Nach jeder neuen Interaktion des Benutzers mit dem System wird die beobachtete Information durch Anwendung dieser Methode in die relevanten CPT-Einträge eingebracht. Die von AHUGIN benötigten ESS-Werte s_{ik} werden mit Werten nahe Null initialisiert.[1] Dies bewirkt, dass sobald ein für eine Zustandskombination von Eltern relevanter Adaptionsfall einer Variable bearbeitet wurde, die initiale, gleichverteilte bedingte Wahrscheinlichkeit keinen Einfluss mehr auf den Adaptionsprozess hinsichtlich dieses Teils der CPTs besitzt (vgl. dazu die formale Beschreibung des Verfahrens in Abschnitt 6.3). Es wird nur noch die gesehene Adaptionsinformation im Modell repräsentiert. Dieser Ansatz repräsentiert den Extrempunkt des Spektrums der Adaptionsoptionen, der durch das Fehlen des Offline-Lernens eines allgemeinen Ausgangsmodells charakterisiert ist.

Allgemeines Benutzermodell Den entgegengesetzten Extrempunkt bildet das *allgemeine Modell*. Es repräsentiert lediglich die allgemeinen Zusammenhänge, die anhand der kompletten Trainingsmenge aller verfügbaren Interaktionsdaten einer großen Menge von (früheren) Benutzern des Systems bzw. Versuchspersonen einer Studie erlernt werden. Dabei werden individuelle Unterschiede nicht explizit während des Lernvorgangs bzw. im Modell berücksichtigt. Zur Laufzeit des Systems findet keinerlei Adaption des Benutzermodells an den aktuellen Benutzer statt. Diese Methode lässt sich als Verfahren interpretieren, dass nur die (statischen) allgemein gültigen Zusammenhänge der betrachteten Domäne identifiziert und ausnutzt.

Parametrisiertes Benutzermodell Zwischen den beiden Extremen befindet sich das *parametrisierte Modell*, das durch die explizite Modellierung individueller Unterschiede durch die Aufnahme entsprechender Parametervariablen in das Bayes'sche Netz charakterisiert ist. Dieser Ansatz wurde bereits in Abschnitt 2.4.2 vorgestellt und im Beispiel zur Erkennung der kognitiven Ressourcenbeschränkungen im Flughafenexperiment verfolgt. In einem Offline-Arbeitsschritt wird

[1] Der Wert Null verhindert den Adaptionsprozess (siehe HUGIN Expert A/S, 2000).

ein allgemeines Netz erlernt, wobei allerdings im Gegensatz zum allgemeinen Modell explizit die Werte der individuellen Parametervariablen im Lernprozess berücksichtigt werden. In vielen Domänen ist es sehr einfach, die benutzerspezifischen Eigenschaften zu erheben oder anhand der gesammelten Daten zu ermitteln (z.B. im Flughafenexperiment die durchschnittliche Artikulationsgeschwindigkeit einer Versuchsperson als arithmetisches Mittel der Artikulationsgeschwindigkeiten aller Äußerungen). Dieses parametrisierte allgemeine Modell kann als Zeitscheibe eines dynamischen Bayes'schen Netzes genutzt werden, um die Adaption an den individuellen Benutzer mit Hilfe der Parametervariablen zu realisieren (vgl. Abschnitt 2.4.2). Dabei werden nach jeder Interaktion zwischen Benutzer und System neue Zeitscheiben an das Netz angehängt und anhand der beobachteten Daten die Werte der individuellen Parametervariablen sukzessiv mit zunehmender Genauigkeit eingeschätzt. Die Parametervariablen werden dazu als statisch deklariert, da sie (statische) Eigenschaften der Benutzer repräsentieren. Die zunehmend genauer eingeschätzten Werte dieser Variablen sollten zugleich zunehmend genauere Vorhersagen über das Interaktionsverhalten des aktuellen Benutzers ermöglichen, da seine individuellen Eigenschaften immer besser vom Modell erfasst werden.[2]

Adaptives Benutzermodell Das *adaptive Modell*[3] benutzt das AHUGIN-Verfahren ähnlich wie das individuelle Modell, lernt aber zusätzlich während einer Offline-Phase ein allgemeines Netz, das als Ausgangspunkt des Adaptionsprozesses dient. Im Gegensatz zum parametrisierten Modell werden keine individuellen Parametervariablen betrachtet. Die Adaption an die individuellen Eigenschaften des aktuellen Benutzers geschieht ausschließlich über die AHUGIN-Adaption der bedingten Wahrscheinlichkeiten. Vorteil dieser Vorgehensweise ist ihr Potential, unterschiedlichste Dimensionen der individuellen Unterschiede ohne vorherige Antizipation durch den Systementwickler automatisch erkennen und ins Modell einbringen zu können, wie es beispielsweise zur Festlegung der Parametervariablen beim parametrisierten Ansatz notwendig ist. Die entscheidende Frage ist hier—wie bereits angesprochen—die Festlegung der Geschwindigkeit, mit der das allgemeine Ausgangsmodell zur Laufzeit an den neuen Benutzer angepasst wird. Eine einfache Lösung dieses Problems, die häufig auch in anderen Kontexten angewendet wird, besteht in der manuellen Spezifikation eines globalen ESS-Parameters der AHUGIN-Methode.

6.3 Methode der differentiellen Adaption

Die manuelle, globale Spezifikation der ESS-Werte ist aus zwei Gründen problematisch: (a) der absolute ESS-Wert stellt lediglich eine Einschätzung des Experten dar und stimmt im Allgemeinen nicht mit dem optimalen „echten" Wert überein, (b) alle Teile der CPTs des Bayes'schen Netzes werden mit der gleichen Geschwindigkeit angepasst, obwohl es typischerweise Regionen des allgemeinen Modells gibt, die bezüglich aller potenziellen Benutzer weitestgehend übereinstimmen, im Gegensatz dazu aber auch Teile des Benutzermodells existieren, die sich stark individuell unterscheiden. So zeigten beispielsweise die Versuchspersonen im Anweisungsexperiment zwar eine ähnliche Fehlertendenz, unterschieden sich aber deutlich hinsichtlich der benötigten Ausführungszeiten zur Bearbeitung der Anweisungen.

[2] Die in Abschnitt 2.4.2 angewendete Prozedur entspricht nicht exakt dem parametrisierten Modell in der hier vorgestellten Form. Die Werte der Parametervariablen waren in der Studie als bekannt vorgegeben.

[3] Auch das individuelle und das parametrisierte Modell stellen adaptive Ansätze dar. Der Begriff 'adaptives Modell' dient in diesem Zusammenhang zur Unterscheidung der verschiedenen Verfahren.

Das im Folgenden vorgestellte Verfahren der *differentiellen Adaption* löst dieses Problem. Es ermittelt automatisch anhand der verfügbaren empirischen Daten *lokale* ESS-Werte: Für jede der Elternzustandskombinationen der Variablen des betrachteten Bayes'schen Netzes wird ein separater ESS-Wert errechnet, der eine für den entsprechenden Aspekt des Benutzermodells spezifische Konfidenz angibt. Mit der Gesamtheit der offline ermittelten unterschiedlichen lokalen ESS-Werte werden im Rahmen des AHUGIN-Adaptionsprozesses zur Laufzeit unterschiedliche Anpassungsgeschwindigkeiten der verschiedenen Teile des Benutzermodells erreicht. Das Verfahren kann als eine Erweiterung der AHUGIN-Methode um eine automatische Ermittlung lokaler ESS-Werte angesehen werden. Es wird bei diesem Verfahren keine (fehleranfällige) manuelle Spezifikation dieser Werte (durch Experten) benötigt.

Ziel ist es, Teile des allgemeinen Bayes'schen-Netz-Benutzermodells, die große individuelle Unterschiede aufweisen, mit höherer Geschwindigkeit an den aktuellen Benutzer anzupassen, als solche, die bei allen bekannten Benutzern weitestgehend übereinstimmen und damit eine hohe allgemeine Gültigkeit besitzen. Die zugrunde liegende Annahme ist dabei, dass es sich lohnt, sich bei der Anpassung auf die Teile der Modelle zu konzentrieren, die die individuell ausgeprägten Eigenschaften der Benutzer repräsentieren. Bei den restlichen, allgemein gültigen Teilen wird durch eine „schwerfälligere" Adaptionsstrategie versucht, Zufallsschwankungen in den beobachteten Interaktionsdaten nicht ins Modell einfließen zu lassen. Es macht bei solchen Aspekten eines Benutzermodells beispielsweise keinen Sinn anhand einer einzigen (widersprüchlichen) Beobachtung radikale Modifikationen vorzunehmen. Es handelt sich im Normalfall hierbei um einen „Ausreißer".

6.3.1 Algorithmus

Abbildung 6.2 beinhaltet das (informelle) Grundgerüst der Vorgehensweise zur differentiellen Adaption von Benutzermodellen in Form Bayes'scher Netze. Im Folgenden wird das Verfahren formal vorgestellt:

DIFFERENTIELLE ADAPTION$(G, \boldsymbol{D})$

1. Lerne ein separates Bayes'sches Netz für jeden Benutzer anhand der Trainingsdaten $\boldsymbol{D}$
2. Bestimme mit Hilfe dieser Netze ein „Durchschnittsmodell" als allgemeines Ausgangsmodell des Adaptionsprozesses
3. Bestimme lokale ESS-Werte anhand der Varianzen der bedingten Wahrscheinlichkeiten in der Menge der separat erlernten Benutzermodelle
4. Wende zur Adaption das AHUGIN-Verfahren mit den ermittelten lokalen ESS-Werten an, um damit unterschiedliche Teile des allgemeinen Benutzermodells mit unterschiedlichen Geschwindigkeiten an den aktuellen Benutzer anzupassen

---

***Abbildung 6.2**: Grundgerüst der Methode der differentiellen Adaption*

Zur Vereinfachung der Notation wird die Methode für eine bestimmte Zustandskombination $pa_k(X_i)$ der Eltern einer Variablen X_i vorgestellt. Zusätzlich wird angenommen, dass zu jedem Benutzer die gleiche Menge an Daten verfügbar ist. Für die Zustandskombination existieren n_i bedingte Wahrscheinlichkeiten $\theta_{ijk}, j = 1, \ldots, n_i$, in der mit X_i assoziierten CPT $\boldsymbol{\theta}_i$. Zusätz-

lich zu diesen n_i bedingten Wahrscheinlichkeiten verwaltet das AHUGIN-Verfahren für jede der Zustandskombinationen $pa_k(X_i)$ eine Dirichlet-Verteilung, um die Wahrscheinlichkeitsverteilung der θ_{ijk} zu modellieren (vgl. Abschnitt 4.3.1). Die Parameter jeder dieser Dirichlet-Verteilungen sind:[4]

- ein Vektor von n_i Mittelwerten m_j,
- eine ESS s_{ik}.

Die Mittelwerte m_j entsprechen den aktuellen Einschätzungen der bedingten Wahrscheinlichkeiten θ_{ijk}. Aus didaktischen Gründen ist es aber sinnvoll, sie mit m_j zu bezeichnen, wenn sie als Parameter der Dirichlet-Verteilung interpretiert werden.

Ein Adaptionsfall, der die Elternzustandskombination $pa_k(X_i)$ instanziiert, d.h., in dem die entsprechenden Zustände der Elternvariablen beobachtet wurden, wird bearbeitet, indem die ESS s_{ik} um Eins erhöht wird und die m_j entsprechend der beim Bayes'schen Lernen üblichen Methode zur Modifikation der Dirichlet-Verteilungen (siehe Abschnitt 4.3.1) behandelt werden.

Die lokalen ESS-Werte s_{ik} können anhand vollständiger Trainingsdaten von N anderen Benutzern folgendermaßen bestimmt werden:

1. Lerne N separate Bayes'sche Netze $B^n = (G, \boldsymbol{\theta}^n)$ anhand der Trainingsdaten—je eines pro Benutzer—unter Verwendung der Standardlernverfahren für die bedingten Wahrscheinlichkeiten im Falle vollständiger Trainingsdaten.

2. Für jede der Zustandskombinationen $pa_k(X_i)$ der Eltern einer Variablen X_i liefern die separat erlernten Netze einen Vektor empirisch ermittelter bedingter Wahrscheinlichkeiten θ^n_{ijk}.

Diese N Vektoren können als eine Stichprobe von Vektoren angesehen werden, auf deren Basis eine Einschätzung des Vektors gemacht werden kann, den man für einen neuen Benutzer erhalten würde, wenn man genügend Daten zu ihm zur Verfügung hätte.

Es bleibt die Frage zu beantworten, wie diese Einschätzung in Form einer initialen n_i-dimensionalen Dirichlet-Verteilung modelliert werden kann. Das Vorgehen ist ähnlich der von Olesen et al. (1992) vorgestellten Methode, eine gegebene empirische Verteilung mit Hilfe einer einzigen Dirichlet-Verteilung zu approximieren.

Die n_i Mittelwerte der Dirichlet-Verteilung sollen den Mittelwerten der zu approximierenden Verteilung exakt entsprechen. Das bedeutet im vorliegenden Fall, dass jedes der m_j wie folgt definiert wird:

$$m_j = \frac{\sum_{n=1}^{N} \theta^n_{ijk}}{N}, \tag{6.1}$$

d.h., jedes m_j entspricht dem arithmetischen Mittel der N CPT-Einträge θ^n_{ijk} der separat erlernten Bayes'schen Netze B^n.

Im Idealfall sollte jede der n_i Varianzen der Dirichlet-Verteilungen der Varianz der entsprechenden N CPT-Einträge θ^n_{ijk} der separat erlernten Netze entsprechen. Im Allgemeinen ist dies nicht möglich, da mit der ESS s_{ik} lediglich ein Freiheitsgrad zur Festlegung der Varianz der Dirichlet-Verteilung vorliegt. Olesen et al. (1992) schlagen vor, die ESS s_{ik} so zu wählen, dass

[4] Die Hyperparameter α^{ik}_j der Dirichlet-Verteilung können anhand der ESS s_{ik} und der Mittelwerte m_j bestimmt werden (siehe Olesen et al., 1992), so dass alternativ beide Parametermengen zur eindeutigen Spezifikation derselben Verteilung verwendet werden können.

der (gewichtete) Durchschnitt der Varianzen der Dirichlet-Verteilungen (in Formeln mit v bezeichnet) dem gewichteten Durchschnitt der Varianzen der zu approximierenden Verteilung entspricht. Gegeben die Formel der Varianz einer Dimension einer Dirichlet-Verteilung,

$$v_j = \frac{m_j(1 - m_j)}{s_{ik} + 1}, \tag{6.2}$$

ergibt sich für die gewichtete durchschnittliche Varianz:

$$v = \frac{\sum_{j=1}^{n_i} m_j^2(1 - m_j)}{s_{ik} + 1}. \tag{6.3}$$

Aufgelöst nach s_{ik} erhält man:

$$s_{ik} = \frac{\sum_{j=1}^{n_i} m_j^2(1 - m_j)}{v} - 1. \tag{6.4}$$

Um die gewünschte Schätzung für die ESS s_{ik} zu berechnen, muss lediglich v durch den berechneten Durchschnitt der n_i Varianzen der empirisch ermittelten CPT-Einträge v' ersetzt werden. Jede dieser n_i Varianzen v'_j ist gegeben durch

$$v'_j = \frac{\sum_{n=1}^{N}(\theta_{ijk}^n - m_j)^2}{N}, \tag{6.5}$$

da m_j bereits als arithmetisches Mittel der entsprechenden θ_{ijk}^n bestimmt wurde.

Um das gewichtete Mittel der Varianzen zu bestimmen, werden die m_j als Gewichte genutzt:

$$v' = \sum_{j=1}^{n_i} m_j v'_j. \tag{6.6}$$

Zusammengefasst ergibt sich als Schätzwert der ESS s_{ik} unter Verwendung der separat gelernten bedingten Wahrscheinlichkeiten und der entsprechenden arithmetischen Mittel m_j:

$$s_{ik} = \frac{N \sum_{j=1}^{n_i} m_j^2(1 - m_j)}{\sum_{j=1}^{n_i} m_j \sum_{n=1}^{N}(\theta_{ijk}^n - m_j)^2} - 1. \tag{6.7}$$

6.3.2 Beispiel

Als erläuterndes Beispiel der Methode soll ihre Anwendung anhand der CPT der binären Variablen FEHLER? des Anweisungsexperiments aus Abbildung 2.5 (a) dienen. Dabei wird die experimentelle Bedingung bestehend aus vier Anweisungen, gebündelter Präsentation und dem Vorhandensein der Nebenaufgabe betrachtet—womit gleichzeitig die Zustandskombination der Eltern von FEHLER? vollständig spezifiziert ist.

Angenommen, man hat durch Anwenden der beschriebenen Methode des separaten Lernens der Benutzermodelle und Berechnen des Mittelwerts einen Wert von 0.6 für die Fehlerhäufigkeit m der Versuchspersonen in dieser Bedingung ermittelt. Außerdem beobachtet man große Unterschiede in den individuellen bedingten Wahrscheinlichkeiten der separat erlernten Benutzermodelle. Diese Situation wird durch die relativ niedrigen, ähnlich hohen, grauen Balken des oberen linken Graphen in Abbildung 6.3 widergegeben. Diese Balken repräsentieren eine solche

hypothetische Stichprobe der potenziellen Wahrscheinlichkeitswerte in der Menge der separat erlernten Bayes'schen Netze. Die zugrunde liegende empirische Verteilung kann mit Hilfe einer Dirichlet-Verteilung approximiert werden—in diesem binären Fall mit der 2-dimensionalen Variante einer Dirichlet-Verteilung, der Beta-Verteilung. In diesem Beispiel wird angenommen, dass sich $Beta(3, 2)$ mit einer berechneten ESS von 5 ergibt.

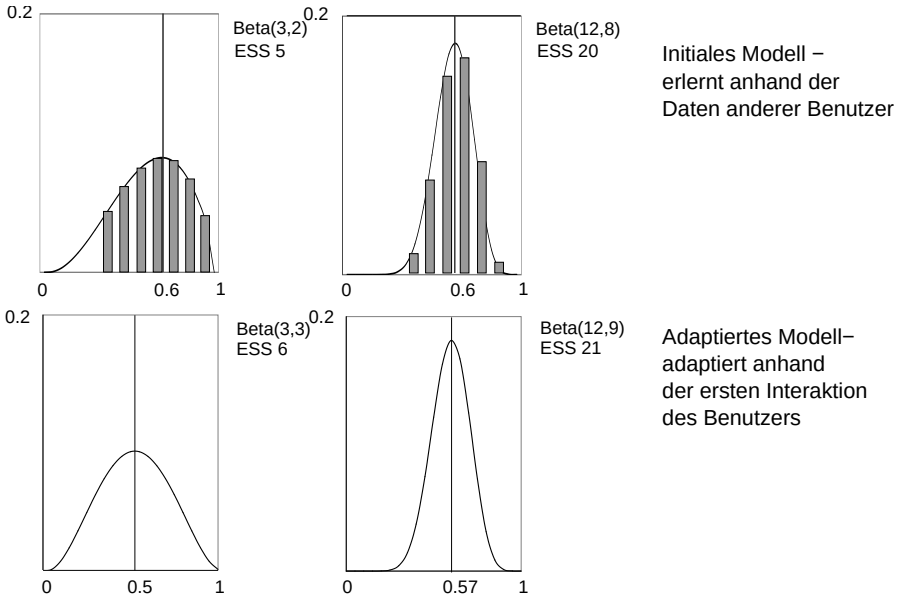


Abbildung 6.3*: Erläuterndes Beispiel zum Verfahren der differentiellen Adaption der bedingten Wahrscheinlichkeiten eines Bayes'schen Netzes*

Angenommen, man beobachtet in einer anderen Situation ebenfalls eine Fehlerhäufigkeit von 0.6, aber weniger große Unterschiede zwischen den separat erlernten bedingten Wahrscheinlichkeiten. Dies resultiert in einer schmaleren Form der Beta-Verteilung $Beta(12, 8)$ mit einer ESS von 20 wie in Abbildung 6.3 rechts oben dargestellt. Man beachte, dass der höhere ESS-Wert von 20 gegenüber 5 die höhere Konfidenz hinsichtlich der initialen Wahl der Wahrscheinlichkeit von 0.6 widerspiegelt, da in dieser Situation weniger Variationen in den personenspezifischen Fehlerhäufigkeiten auftreten.

Bislang wurden die initialen Werte spezifiziert, es bleibt den dem Bayes'schen Lernansatz entsprechenden Vorgang der Adaption an einen neuen Benutzer zu beschreiben: Angenommen, der neue Benutzer interagiert mit dem System und führt dabei Anweisungen fehlerfrei aus. Dann kann diese Beobachtung als Adaptionsfall genutzt werden, um das Bayes'sche Netz folgendermaßen an den Benutzer anzupassen: In der ersten Situation, in der das Modell größere individuelle Unterschiede repräsentiert, liefert die Anwendung des AHUGIN-Adaptionsprozesses $Beta(3, 3)$ mit Mittelwert 0.5 und ESS 6 (siehe unterer linker Teil der Abbildung). In der Situation mit geringerer Varianz der Fehlerhäufigkeiten erhält man als Mittelwert 0.57 und eine ESS von 21. Man sieht, dass wenn dem Modell eine höhere Konfidenz zugeordnet ist (durch einen hohen ESS-Wert), eine langsamere Adaption der bedingten Wahrscheinlichkeit vorgenommen wird—obwohl beide Modelle auf der gleichen Menge an empirischen Daten basieren.

6.3.3 Diskussion

Eine Bewertung des Verfahrens der differentiellen Adaption Bayes'scher Netze unter dem Komplexitätsgesichtspunkt in Bezug auf einen Einsatz in benutzeradaptiven Systemen attestiert diesbezüglich eine gute Eignung. Zusätzlicher Berechnungsaufwand—im Vergleich zum Standard-Adaptionsverfahren AHUGIN—findet nur im Offline-Teil des Verfahrens durch die Bestimmung der lokalen ESS s_{ik} statt. Die asymptotische Komplexität dieser Teilaufgabe wird durch die eingesetzten CPT-Lernverfahren determiniert. Das online eingesetzte AHUGIN-Verfahren zur Adaption der bedingten Wahrscheinlichkeiten erfolgt unmodifiziert.

Unterschiedlichen Trainingsdatensatzgrößen der einzelnen Benutzer sollte durch eine entsprechend gewichtete Bestimmung der Mittelwerte m_j Rechnung getragen werden.

Die Approximation der empirischen Verteilung mit lediglich einer Dirichlet-Verteilung (anstelle einer Linearkombination von Dirichlet-Verteilungen, vgl. Abschnitt 4.5.1) erscheint bei der Existenz von mehr als einem Häufungspunkt der Stichprobe der bedingten Wahrscheinlichkeiten problematisch. In einer den Graphen in Abbildung 6.3 entsprechender Darstellung würde dies durch eine sehr flache Glockenform mit einem Mittelwert zwischen zwei Häufungspunkten, die zwei in sich homogene, aber gegenseitig sehr unterschiedliche Benutzergruppen repräsentieren, resultieren. In der Praxis stellt dies aber nur ein geringes Problem dar, da bedingt durch die hohe (Gesamt-)Varianz ein kleiner ESS-Wert errechnet wird, was wiederum in einer sehr schnellen Adaptionsgeschwindigkeit mündet. Deshalb wird im Normalfall das System bereits nach wenigen Adaptionsfällen den aktuellen Benutzer der richtigen Gruppe zugeordnet haben.

Aus Sicht der Statistik kann das (differentiell) adaptive Modell als eine Approximation des *hierarchischen Bayes'schen Ansatzes* (siehe z.B. Berger, 1985) interpretiert werden. Im hierarchischen Bayes'schen Ansatz wird ein zweistufiges Modell aufgebaut, das auf der oberen Ebene, Abhängigkeiten zwischen den freien Parametern der unteren Ebene in Form von Wahrscheinlichkeitsverteilungen (zweiter Stufe) modelliert. Übertragen auf den Benutzermodellierungskontext könnte man auf der oberen Ebene eine Wahrscheinlichkeitsverteilung über verschiedene Gruppenmodelle verwalten, wohingegen auf der unteren Ebene die mit Hilfe eines einzelnen Modells repräsentierte Wahrscheinlichkeitsverteilung einer Gruppe betrachtet wird.

Tabelle 6.1 fasst die vorgestellten Alternativen der Adaption von Benutzermodellen in Form Bayes'scher Netze kompakt zusammen und stellt sie vergleichend gegenüber.

6.4 Analysen

Es folgt ein Vergleich der Performanz der vorgestellten alternativen Adaptionsverfahren anhand der empirischen Daten des Anweisungs- und Flughafenexperiments. Der Schwerpunkt der Diskussion liegt dabei auf der Evaluation des entwickelten Verfahrens der differentiellen Adaption in Bezugnahme auf die existierenden Ansätze der Adaption allgemeiner Benutzermodelle an individuelle Benutzer.

6.4.1 Methode

Tabelle 6.2 fasst die in eine 24- bzw. 32fache Leave-one-out-Kreuzvalidierung eingebettete Evaluationsprozedur sowie Informationen zur Präsentation der Ergebnisse zusammen. Für jeden der fünf in Tabelle 6.1 angeführten Ansätze wurde diese Prozedur durchgeführt.

Art des Modells	**Lernen anhand anderer Benutzer**	**Adaption zur Laufzeit an den aktuellen Benutzer**
Individuell	Kein Offline-Lernen: Im initialen Benutzermodell werden die CPTs gleichverteilt spezifiziert und eine ESS nahe 0 vorgegeben	Nach jedem Adaptionsfall werden die relevanten Zustandskombinationen der Eltern-Kind-Paare der Variablen gemäß des AHUGIN-Verfahrens adaptiert
Allgemein	Offline-Lernen auf der Basis aller verfügbaren Daten (anderer Benutzer), ohne die Verwendung individueller Parametervariablen	Keine Adaption
Parametrisiert	Offline-Lernen einer Zeitscheibe anhand aller verfügbaren Daten (anderer Benutzer) mit individuellen Parametervariablen	Dynamisches Bayes'sches Netz mit neuer Zeitscheibe für jeden Adaptionsfall. Sukzessives Einschätzen der statischen individuellen Parametervariablen
Adaptiv	Offline-Lernen eines allgemeinen Modells mit zusätzlicher manuellen Spezifikation einer globalen ESS	Nach jedem Adaptionsfall werden die relevanten Zustandskombinationen der Eltern-Kind-Paare der Variablen gemäß des AHUGIN-Verfahrens unter Verwendung der spezifizierten globalen ESS adaptiert
Differentiell adaptiv	*Offline-Lernen eines allgemeinen Modells und automatische Spezifikation lokaler ESS-Werte nach dem Verfahren der differentiellen Adaption*	*Nach jedem Adaptionsfall werden die relevanten Zustandskombinationen der Eltern-Kind-Paare der Variablen gemäß des* AHUGIN*-Verfahrens unter Verwendung der im Rahmen der differentiellen Adaption bestimmten lokalen ESS adaptiert*

Tabelle 6.1*: Zusammenfassung der Alternativen zur Adaption der CPTs Bayes'scher Netze*

Da an dieser Stelle die Performanz der erlernten bzw. adaptierten Bayes'schen Netze hinsichtlich der Inferenz der Werte einzelner Variablen in den gleichzeitig als Adaptionsfälle dienenden Daten untersucht werden soll, wurde als Qualitätsmaß das Standardmaß des quadratische Fehlers zwischen der inferierten Wahrscheinlichkeitsverteilung und den tatsächlich beobachteten Werten gewählt. Damit ist es möglich, die Qualität der Verfahren separat für Teilaspekte—insbesondere für einzelne Variablen—der Bayes'schen Netze zu diskutieren. Dies ist gerade im Zusammenhang mit der differentiellen Adaption von besonderem Interesse.

Die in den Graphen dargestellten Ergebnisse wurden in Blöcken von je acht Adaptionsschritten als Durchschnitt der acht Einzelbewertungen zusammengefasst, da sonst die Hauptergebnisse durch zufallsbedingte Schwankungen nur sehr schwer zu identifizieren sind.

- *Initiales Modell*
  Ein Bayes'sches Netz, das dem betrachteten Adaptionsansatz entsprechend (siehe Tabelle 6.1) spezifiziert bzw. anhand der Daten der anderen 23 bzw. 31 Versuchspersonen erlernt wurde
- *Aufbereitung der Testdaten*
  Festlegung einer zufälligen Reihenfolge der im entsprechenden Experiment zur betrachteten Versuchsperson erhobenen Daten (je 72 bzw. 80 Fälle)
- *Testen des Modells*
  Abarbeitung der Daten eines Benutzers in der festgelegten Reihenfolge wie folgt:
  1. Ermitteln der Wahrscheinlichkeitsverteilung der Zustände der untersuchten Variable durch Instantiierung aller verbleibenden Variablen des Bayes'schen Netzes mit anschließender Anwendung der Inferenzverfahren
  2. Bestimmen des quadratischen Fehlers bezüglich des tatsächlichen Wertes
  3. Adaption des Modells gemäß des eingesetzten Adaptionsmechanismus (siehe Tabelle 6.1) anhand des kompletten Adaptionsfalls
- *Präsentation der Ergebnisse*
  Jede Kurve eines Graphen repräsentiert die im Rahmen der Kreuzvalidierung ermittelten durchschnittlichen quadratischen Fehler

  Zur Verdeutlichung der allgemeinen Trends wurden die Ergebnisse über mehrere (8) Adaptionsfälle aggregiert, um die Effekte von Zufallsschwankungen in der Darstellung zu vermindern.

***Tabelle 6.2**: Evaluationsprozedur zum Vergleich der alternativen Adaptionsverfahren*

6.4.2 Ergebnisse

Die zu den beiden Datensätzen der Experimente gehörigen Ergebnisse werden separat präsentiert, um anhand der Experimentalsituation die beobachteten Effekte ausführlich diskutieren zu können.

6.4.2.1 Anweisungsexperiment

Abbildung 6.4 zeigt das in der vorliegenden Studie eingesetzte Bayes'sche Netz. Die individuelle Parametervariable war dabei nur in den Zeitscheiben des parametrisierten Ansatzes vorhanden; alle verbleibenden Adaptionsansätze wurden mit der Netzstruktur evaluiert, die diese Variable und die zugehörige Kante nicht enthielt. Prinzipiell könnte eine individuelle Parametervariable für FEHLER? in das Netz aufgenommen werden. Darauf wurde in dieser Analyse verzichtet, da aufgrund der geringen Häufigkeit von Fehler (durchschnittlich ca. 6 Stück bei 72 Anweisungsfolgen) bei begrenztem Datenmaterial (72 Fälle pro Versuchsperson) es nicht möglich ist, systematische Unterschiede in den Fehlertendenzen zwischen den einzelnen Personen zu erkennen. Dies wird auch durch die Ergebnisse des entsprechenden Abschnitts bestätigt, da das adaptive Modell nicht in der Lage ist besser als die Alternativen zu arbeiten. Eine Aufnahme der Parametervariable hätte im Gegenteil schlechtere Ergebnisse zur Folge, da auf der Basis der gleichen Datenmenge eine höhere Anzahl an—wie diskutiert—wenig aussagekräftigen bedingten Wahrscheinlichkeiten erlernt werden müsste.

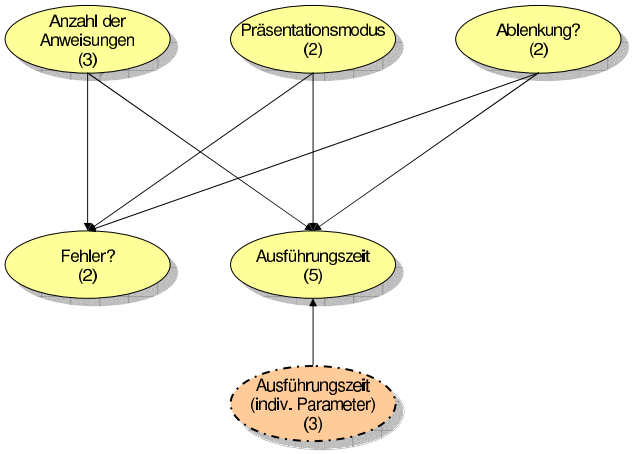


***Abbildung 6.4**: Zur Evaluation der differentiellen Adaption verwendetes Bayes'sches Netz für das Anweisungsexperiment*

Zuerst werden die Ergebnisse einer Vorhersage der Werte der beiden Variablen AUSFÜHRUNGSZEIT und FEHLER? behandelt. Anschließend werden die die Variablen PRÄSENTATIONSMODUS, ANZAHL DER ANWEISUNGEN und ABLENKUNG? betreffenden Klassifikationsaufgaben, d.h., die Ermittlung der experimentellen Bedingung anhand der beobachteten Fehler bzw. Ausführungszeiten, besprochen.

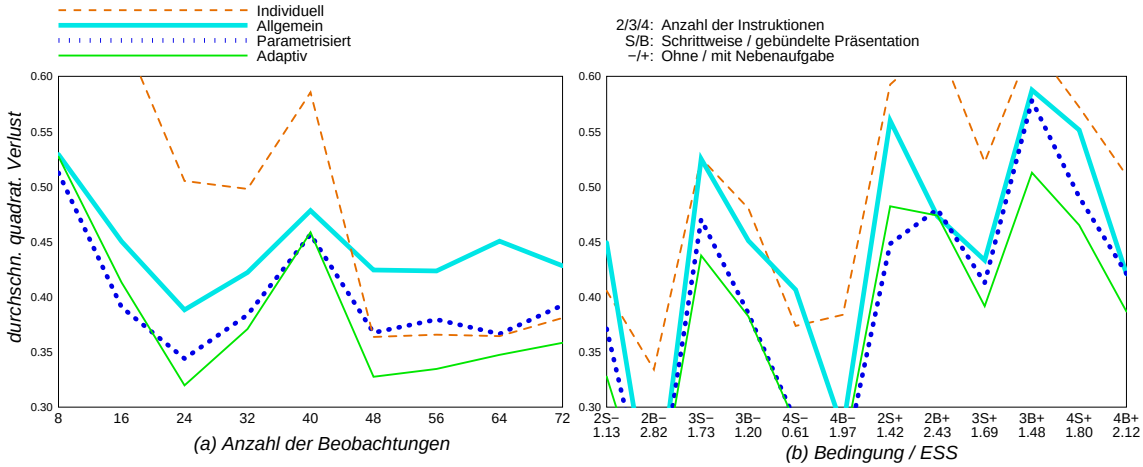
Vorhersage einer mehrwertigen Variablen mit individuellen Unterschieden Die Ergebnisse des allgemeinen Benutzermodells für die Variable AUSFÜHRUNGSZEIT werden in Abbildung 6.5 durch die durchgezogene Kurve dargestellt. Es ist zu beachten, dass der einzige Grund, weshalb es sich dabei nicht um eine horizontale Linie handelt, in den zufallsbedingten Fluktuationen der Daten begründet liegt, die durch die Anordnung in zufälliger Reihenfolge sowie der Durchschnittsbildung im Rahmen der Kreuzvalidierung nicht komplett eliminiert werden konnten. Deshalb macht es an dieser Stelle keinen Sinn, die „Zick-Zack"-Form der Kurve des allgemeinen Benutzermodells einer Interpretation zu unterziehen. Insbesondere die scheinbare Verbesserung der Vorhersagequa-

lität zwischen dem ersten und zweiten Block der Adaptionsfälle ist aus den angeführten Gründen rein zufallsbedingt.

Im Gegensatz dazu ist es durchaus sinnvoll, die Performanz des allgemeinen Modells mit derjenigen des parametrisierten zu vergleichen. Im Folgenden wird die statistische Signifikanz der Ergebnisse mittels des einfachen Vorzeichentests angegeben, der auf den Ergebnissen der letzten 24 Adaptionsfälle (3 Blöcke in den Abbildungen) basiert. Das parametrisierte Modell liefert konsistent bessere Vorhersagen zur Variablen AUSFÜHRUNGSZEIT in diesen letzten 24 Fällen ($p < 0.001$). Diese Beobachtung ist verständlich, wenn man die großen individuellen Unterschiede, die hinsichtlich den Ausführungszeiten existieren, beachtet (vgl. Abschnitt 2.2.1). Beispielsweise konnten einige Versuchspersonen die Aufgabe schneller bearbeiten, da sie u.a. geübter im Umgang mit der Computermaus waren als andere.

Das differentiell adaptive Benutzermodell ist in der Lage, diese individuellen Unterschiede ungefähr genauso gut zu behandeln, wie das parametrisierte; zusätzlich scheint es in den letzten drei Blöcken etwas besser zu arbeiten. Dieser Unterschied ist in diesem Fall allerdings nicht statistisch signifikant ($p = 0.15$); in der Diskussion der Resultate des Flughafenexperiments wird eine Situation beschrieben werden, in der der entsprechende Unterschied zwischen den beiden Ansätzen in der Tat signifikant ist.

Die Kurve des individuellen Benutzermodells weist eine Form auf, die sich als typisch herausstellen wird: Zuerst liefert das Modell sehr schlechte Vorhersagen—wie man es auch aufgrund der Initialisierung mit den gleichverteilten Wahrscheinlichkeiten erwartet. Zu Beginn der letzten 24 Adaptionsfälle hat das individuelle Benutzermodell weitestgehende zu den anderen Ansätzen aufgeschlossen. Es liefert dann signifikant bessere Ergebnisse als das allgemeine Modell ($p < 0.05$).



(a) Anzahl der Beobachtungen

(b) Bedingung / ESS

***Abbildung 6.5**: Vorhersagegenauigkeit für die Variable* AUSFÜHRUNGSZEIT

Abbildung 6.5 (b) gibt eine andere Perspektive auf die Resultate: Sie zeigt den durchschnittlichen quadratischen Fehler jeder Modellalternative separat für jede der 12 experimentellen Bedingungen des Anweisungsexperiments, die eindeutig durch die Zustandskombinationen der unabhängigen Variablen spezifiziert sind. Da diese Bedingungen in zufälliger Reihenfolge in den Adaptionsfällen auftraten, sagt dieser Graph nichts über die zeitliche Entwicklung der Adaptionsfähigkeit der alternativen Ansätze aus. Er gibt Hinweise auf den Erfolg der unterschiedlichen

Methoden in den einzelnen experimentellen Bedingungen: Die relative Performanz der Modelle kann auch in dieser Darstellung in allen zwölf Bedingungen beobachtet werden. Insgesamt ist die Vorhersage der Ausführungszeit in den Bedingungen schwieriger, in denen die ablenkende Nebenaufgabe vorhanden war (rechte Hälfte des Graphen). Diese Tatsache ist vermutlich darauf zurückzuführen, dass die Nebenaufgabe eine teilweise unvorhersagbare Zusatzbelastung für die Versuchspersonen darstellte (vgl. Abschnitt 2.2.1). Die sehr schlechte Performanz des individuellen Modells in der initialen Phase wirkt sich hier in den absolut schlechtesten Durchschnittswerten aller Methoden aus.

Abbildung 6.5 (b) beinhaltet zusätzlich die ESS-Werte für jede der durch die experimentelle Bedingung gegebenen Elternzustandskombination von AUSFÜHRUNGSZEIT. Es ist zu beobachten, dass in der Tat unterschiedliche Werte auftreten. Die Unterschiede werden im Zusammenhang mit anderen Variablen diskutiert, wo sie stärker ausgeprägt sind.

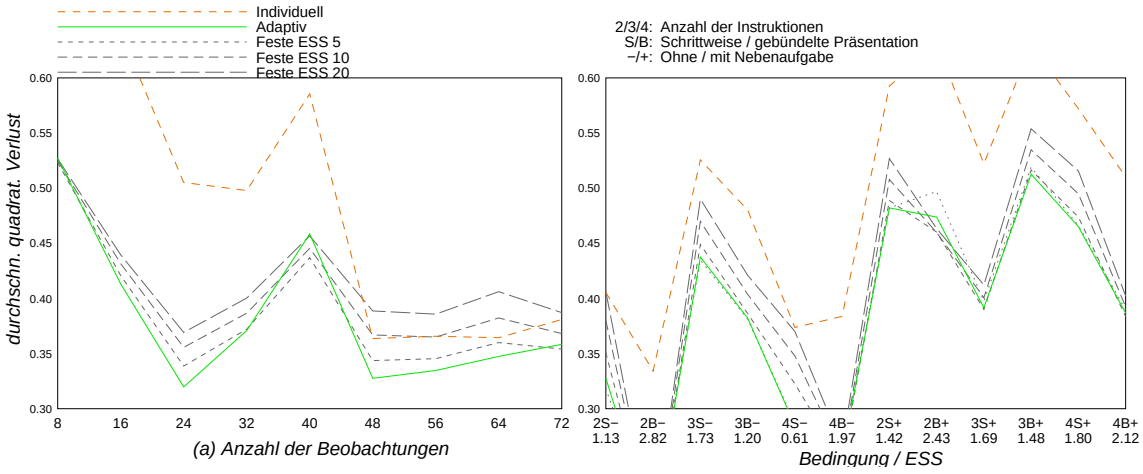


***Abbildung 6.6**: Vorhersagegegenauigkeit für die Variable* AUSFÜHRUNGSZEIT *- Vergleich mit manuell spezifizierter, globaler ESS*

Abbildung 6.6 zeigt—der Übersichtlichkeit wegen in einer separaten Darstellung—die Ergebnisse des adaptiven Benutzermodells mit unterschiedlichen manuell spezifizierten, globalen ESS-Werten im direkten Vergleich mit der differentiellen Adaption. Ein adaptives Benutzermodell mit manuell festgelegten, *globalen* ESS-Werten ist nicht in der Lage, bessere Ergebnisse als die automatische, *differentielle* Variante zu produzieren. Zwar besteht die Möglichkeit, bei entsprechender ESS-Wahl nahe an die Performanz des differentiellen Modells heranzukommen, der dazu benötigte Aufwand, die beste ESS-Wahl zu treffen, ist typischerweise aber sehr hoch. Deshalb besteht der Hauptvorteil des Verfahrens der differentiellen Adaption in diesem Zusammenhang darin, die lokalen ESS-Werte automatisch, anhand der verfügbaren empirischen Daten, zu berechnen.

Vorhersage eines Ereignisses mit geringer Wahrscheinlichkeit Die Variable FEHLER? (Abbildung 6.7) stellt ein Beispiel einer Variablen dar, hinsichtlich derer wenig durch einen Adaptionsprozess an den individuellen Benutzer gewonnen werden kann. Fehler sind im Anweisungsexperiment sehr selten beobachtete Ereignisse: Die durchschnittliche Versuchsperson hat nur ungefähr sechs Fehler bei insgesamt 72 Anweisungssequenzen begangen, bzw. 0.5 pro jeder einzelnen der zwölf experimentellen Bedingungen. Aus diesem Grund ist es für ein System inhärent schwierig,

ein Modell der „Fehlerneigung“ des Benutzers zu erstellen, das bessere Vorhersagen als das allgemeine Benutzermodell liefern kann. Deshalb wurde keine entsprechende individuelle Parametervariable in das Bayes’sche Netz aufgenommen und die Ergebnisse des parametrisierten Modells entsprechen denjenigen des allgemeinen Modells. Der (differentiell) adaptive Ansatz bestätigt, dass es sich in dieser Situation nicht lohnt, eine Anpassung an den individuellen Benutzer im Verlauf der Interaktion vorzunehmen.

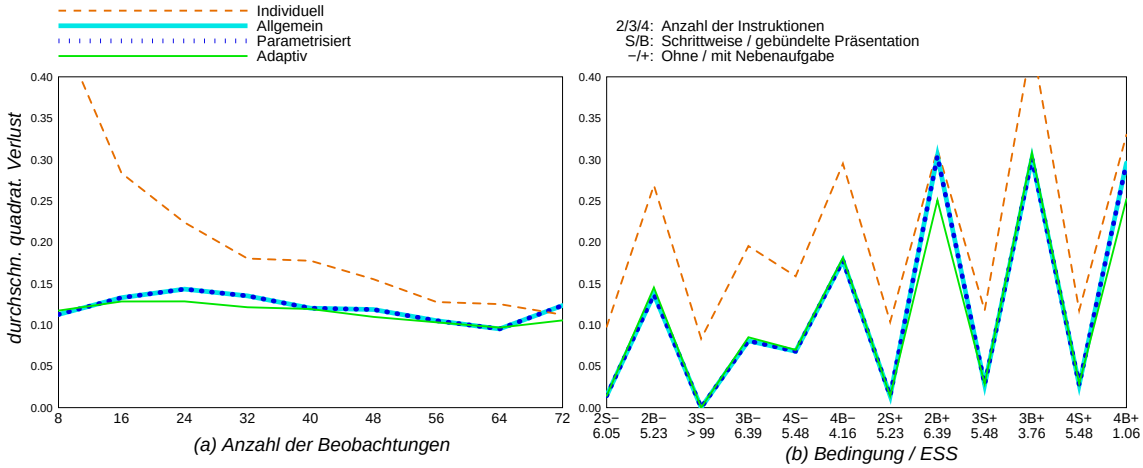


***Abbildung 6.7**: Vorhersagegenauigkeit für die Variable* FEHLER?

Abbildung 6.7 (b) zeigt, dass sich dieses Muster konsistent durch alle experimentellen Bedingungen bzw. Elternzustandskombinationen der Variablen FEHLER? zieht. Außerdem veranschaulicht der Graph ein Verhalten, das typisch für Variablen ist, die ein Ereignis mit geringer Eintrittswahrscheinlichkeit modellieren: Die Vorhersagequalität ist deutlich besser in denjenigen der sechs Bedingungen, die eine schrittweise Präsentation der Anweisungen aufweisen. Diese experimentellen Bedingungen produzierten wesentlich geringere Fehlerraten; und es gilt zu beachten, dass es wesentlich einfacher ist, Vorhersagen über ein Ereignis zu machen, das fast nie eintritt: Es wird immer vorhergesagt, dass das Ereignis nicht eintreten wird. Da es aufgrund des seltenen Eintretens nur zu wenigen fehlerhaften Vorhersagen kommt, ist der quadratische Fehler ebenfalls gering.

Die ESS-Werte fallen bei dieser Variablen deutlich höher aus als bei AUSFÜHRUNGSZEIT, d.h., im Rahmen der Initialisierung des Benutzermodells erkennt das System tatsächlich die geringere Varianz in den bedingten Wahrscheinlichkeiten der CPTs der zur Konstruktion des allgemeinen Ausgangsmodells verwendeten Einzelmodelle und legt damit fest, dass es nicht sinnvoll erscheint, die neuen CPTs (zu) schnell anhand eines gelegentlich auftretenden Adaptionsfalls, der einen Fehler repräsentiert, zu modifizieren.

Klassifikation der experimentellen Bedingung Die in Abbildung 6.8 dargestellten Ergebnisse beziehen sich auf eine andere Aufgabenstellung: Anstatt der Vorhersage eines bestimmten Aspektes des Benutzerverhaltens sollte das System anhand der gemachten Beobachtungen zum Benutzerverhalten die experimentelle (Teil-)Bedingung inferieren, d.h., eine Wahrscheinlichkeit dafür bestimmen, ob bei Kenntnis der beiden verbleibenden Teilaspekte der experimentellen Bedingung (Anzahl der Instruktionen, Präsentationsmodus) eine ablenkenden Nebenaufgabe vorliegt.

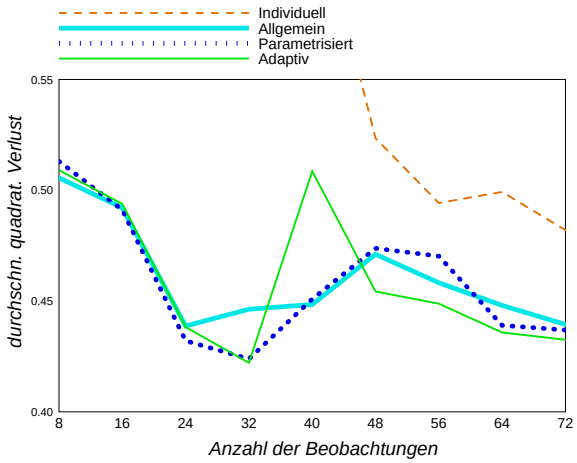


***Abbildung 6.8**: Klassifikationsgenauigkeit für die Variable* ABLENKUNG?

Auf den ersten Blick erscheint das Muster der Kurven des Graphen zu einem gewissen Grad inkonsistent mit den bisher vorgestellten Ergebnissen:

- Das individuelle Benutzermodell schließt nicht zu den anderen Modelle auf, im Gegensatz zu allen anderen bisher diskutierten Situationen.
- Das parametrisierte sowie das differentiell adaptive Benutzermodell zeigen keine bessere Qualität als der allgemeine Ansatz, obwohl sie dies in deutlicher Form hinsichtlich der Variablen AUSFÜHRUNGSZEIT tun.

Ein ähnliches Verhalten lässt sich für jede der beiden verbleibenden unabhängigen Variablen beobachten, wie in Abbildung 6.9 dargestellt. Ausnahmen sind dabei: Bei ANZAHL DER ANWEISUNGEN wird das parametrisierte Benutzermodell signifikant besser ($p < 0.01$) als das allgemeine, und hinsichtlich PRÄSENTATIONSMODUS beobachtet man ein signifikante Überlegenheit ($p < 0.05$) des differentiell adaptiven Ansatzes gegenüber dem allgemeinen und parametrisierten Modell. Obwohl die Adaption der Modelle an den individuellen Benutzer eine statistisch signifikante Verbesserung erzielt, sind die Vorteile im Vergleich zur Vorhersage von AUSFÜHRUNGSZEIT hier weniger deutlich. Die Gründe dieser Diskrepanz werden im Anschluss an die entsprechenden Ergebnisse des Flughafenexperiments diskutiert.

6.4.2.2 Flughafenexperiment

Das im Fall des Flughafenexperiments eingesetzte Bayes'sche Netz ist in Abbildung 6.10 zu sehen. Auch hier werden—wie bei der Diskussion des Anweisungsexperiments—die individuellen Parametervariablen nur im parametrisierten Modell verwendet.

Vorhersage einer Variablen mit einfachen individuellen Unterschieden Wie auch beim Anweisungsexperiment werden zuerst die Ergebnisse der Vorhersage der abhängigen Variablen vorgestellt. Abbildung 6.11 zeigt die Ergebnisse der Variablen ARTIKULATIONSGESCHWINDIGKEIT.

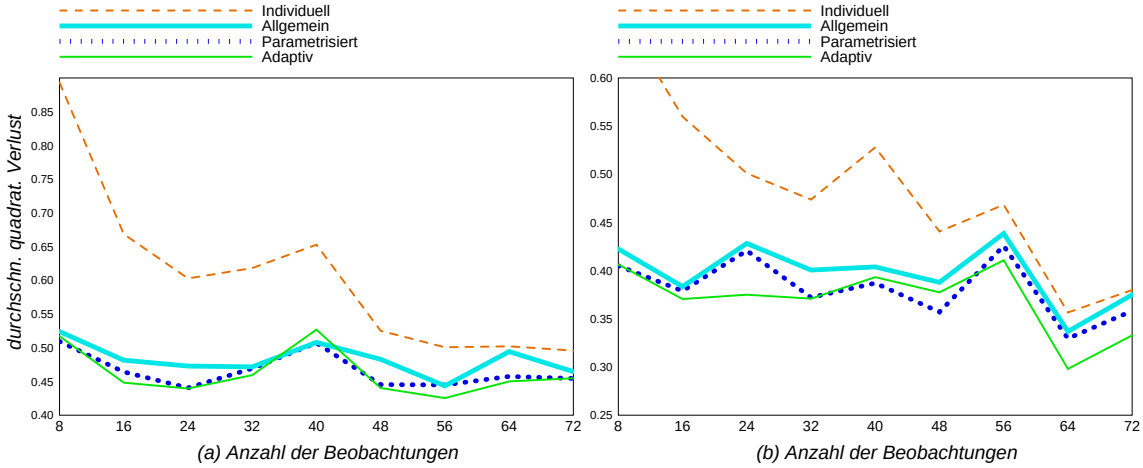


***Abbildung 6.9**: Klassifikationsgenauigkeit für die Variablen* ANZAHL DER ANWEISUNGEN *und* PRÄSENTATIONSMODUS

Es ist allgemein bekannt, dass bezüglich dieser Eigenschaft stabile Unterschiede zwischen einzelnen Personen existieren. Dies spiegelt sich in den Kurven von Abbildung 6.11 wider, die ähnlich zu den entsprechenden Ergebnissen der Variablen AUSFÜHRUNGSZEIT des Anweisungsexperiments sind: Das individuelle Benutzermodell schließt nach Berücksichtigung von ungefähr 30 der 80 Adaptionsfälle zum allgemeinen Modell auf, und ist in der Lage, die letzten 24 Fälle signifikant besser ($p < 0.05$) vorherzusagen. Sowohl das parametrisierte als auch das differentiell adaptive Benutzermodell arbeiten besser als das allgemeine Modell ($p < 0.01$). Der differentiell adaptive Ansatz kann das parametrisierte Modell auch nicht nach der Verarbeitung des Großteils der Adaptionsfälle schlagen. Es scheint deshalb hier im Vergleich zur Verwendung eines Parameters kein Vorteil zu sein, jede Elternzustandskombination separat im Rahmen des Adaptionsprozesses zu behandeln. Abbildung 6.11 (b) bestätigt diese Beobachtung für alle vier experimentellen Bedingungen bzw. die zugehörigen Elternzustandskombinationen.

Vorhersage einer Variablen mit komplexen individuellen Unterschieden Ebenfalls bekannt ist, dass unterschiedliche Personen tendenziell unterschiedlich „viel“ artikulieren, d.h., die Länge von Äußerungen in einer gegebenen Situation variiert personenbezogen. Die Ergebnisse der Variable SILBENANZAHL, die die Gesamtlänge der Äußerungen repräsentiert, werden in Abbildung 6.12 gegeben. In diesem Fall sind die individuellen Unterschiede deutlicher ausgeprägt als bei ARTIKULATIONSGESCHWINDIGKEIT: Das individuelle Modell schließt zum allgemeinen Benutzermodell innerhalb des dritten Blocks auf, und liegt in den letzten drei Blöcken gemeinsam mit ihm an erster Stelle der Performanzskala.

Weiterhin liefert das differentiell adaptive Benutzermodell während der letzten 24 Adaptionsfälle signifikant bessere Ergebnisse ($p < 0.02$) als das parametrisierte Modell. Abbildung 6.12 (b) macht deutlich, dass diese Überlegenheit nur in einer der vier experimentellen Bedingungen bzw. Elternzustandskombinationen vorliegt: derjenigen, in der die Versuchspersonen instruiert wurden, qualitativ hochwertige Äußerungen zu formulieren ohne gleichzeitig durch die simulierte Flughafenumgebung navigieren zu müssen. Einige der Versuchpersonen reagierten auf diese Forderung durch die Produktion langer, ausführlicher Äußerungen, während andere auf die Klarheit

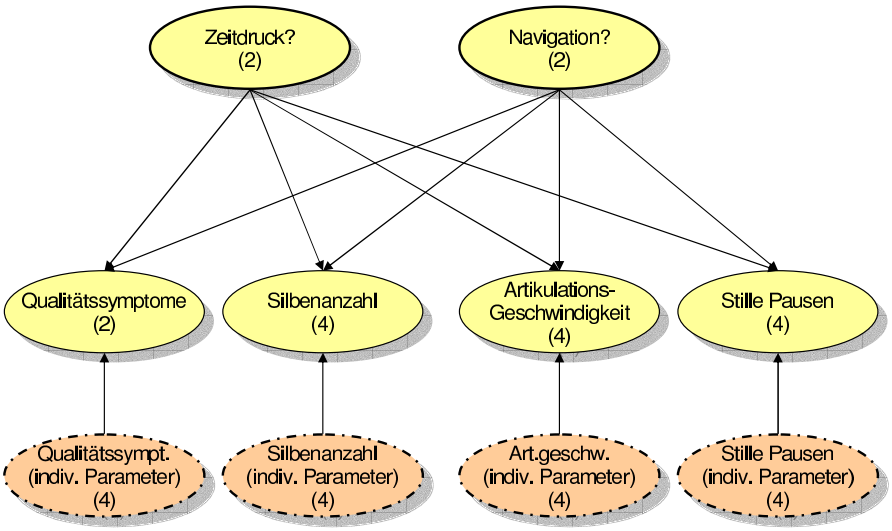


***Abbildung 6.10**: Zur Evaluation der differentiellen Adaption verwendetes Bayes'sches Netz für das Flughafenexperiment*

der Aussagen bei normaler Länge fokussierten. Damit sind diese individuellen Unterschiede nur schwer in einer Dimension der allgemeinen Tendenz der Silbenanzahlen zu erfassen, die durch die individuelle Parametervariablen repräsentiert wird. Die Fähigkeit des individuellen und differentiell adaptiven Benutzermodells, das Verhalten der Versuchsperson in jeder der verschiedenen Situationen separat zu modellieren, ist damit bei dieser Problemstellung als Vorteil anzusehen.

In diesem Zusammenhang ist es interessant, die Ergebnisse bei manueller Spezifikation globaler ESS-Werte zu betrachten. In Abbildung 6.13 stellt die durchgezogene Kurve die differentiell adaptive Methode dar, die aus Abbildung 6.12 (b) wiederholt wird. Jede der unterbrochenen Kurven repräsentiert die Ergebnisse der globalen ESS-Werte von 1, 5, 10 und 20. Die Resultate zu 1, 10 und 20 sind deutlich schlechter als bei der Wahl von 5 sowie bei der differentiellen Adaption, was bedeutet, dass die Wahl einer adäquaten ESS in der Tat von Bedeutung ist. Die Tatsache, dass ein Wert von 5 fast die Ergebnisse der differentiellen Adaption erreichen kann, ist nicht überraschend, da die im Rahmen der differentiellen Adaption ermittelten Werte nahe 5 liegen. Der Hauptbeitrag der differentiellen Methode in einer solchen Situation besteht in der automatischen Bestimmung adäquater Werte anhand der empirischen Daten ohne die Notwendigkeit, aufwendige Testreihen zur Ermittlung der ESS-Werte durchführen zu müssen. Die differentielle Adaption erzielt eine leicht verbesserte Vorhersagequalität bedingt durch die lokalen, unterschiedlichen ESS-Werte für jede der Elternzustandskombinationen; insbesondere wurde ein geringer Wert für die experimentelle Bedingung „qualitativ hochwertige Äußerungen, keine Navigationsaufgabe" bestimmt, die sich durch das Auftreten großer individueller Unterschiede auszeichnet.

Vorhersage von Ereignissen mit geringer Wahrscheinlichkeit Die Variable QUALITÄTSSYMPTOME (Abbildung 6.14 (a)) ist vergleichbar mit der Variablen FEHLER? aus dem Anweisungsexperiment: Da es sich hierbei um ein relativ seltenes Ereignis handelt, ist es schwierig, eine Verbesserung gegenüber dem allgemeinen Modell zu erzielen. Wie bei FEHLER? ist die Häufigkeit geringer—und damit die Vorhersage entsprechend einfacher—wenn keine Nebenaufgabe vorlag, wie in Abbildung 6.14 (b) zu erkennen.

Ebenso sind stille Pausen (Abbildung 6.15 (a)) innerhalb einer Äußerung relativ seltene Ereig-

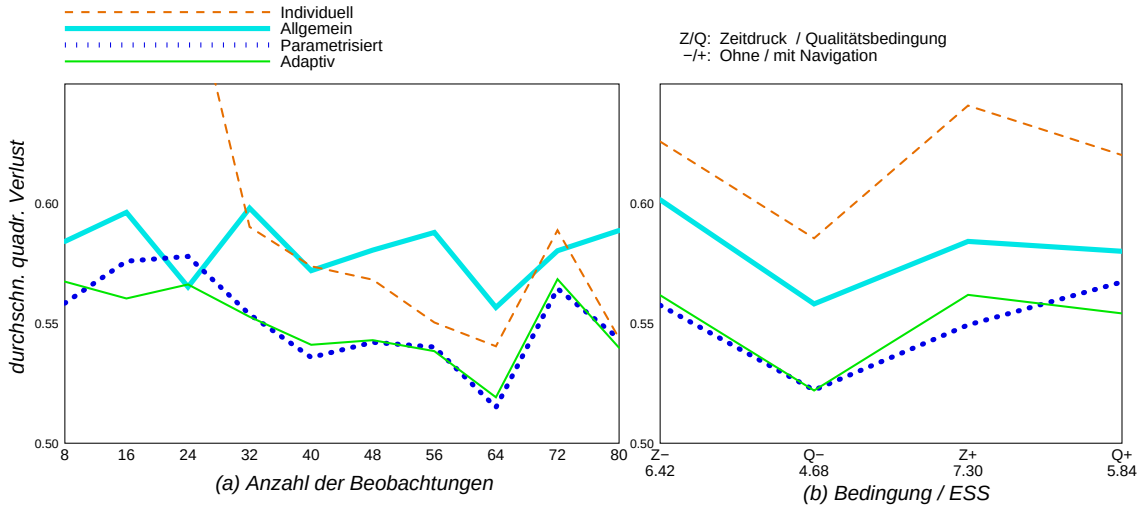


***Abbildung 6.11**: Vorhersagegenauigkeit für die Variable* ARTIKULATIONSGESCHWINDIGKEIT

nisse, die in ungefähr jeder fünften Äußerung auftreten. Die Variable STILLE PAUSEN modelliert die Gesamtlänge in Relation zur Gesamtlänge der Äußerung. Aus diesem Grund existieren vier mögliche Zustände der Variable (nach der Diskretisierung). Diese feinkörnige Aufteilung kann ein Grund sein, weshalb das parametrisierte Benutzermodell eine bessere Leistung als das allgemeine erzielt ($p < 0.02$). Abbildung 6.15 (b) legt nahe, dass der Nachteil des allgemeinen Modells auf die Bedingung mit qualitativ hochwertigen Äußerungen ohne Nebenaufgabe beschränkt ist, ähnlich wie bei SILBENANZAHL, wobei hier der Effekt weniger stark ausgeprägt ist.

Klassifikation der experimentellen Bedingung Im Flughafenexperiment existieren zwei unabhängige Variablen, die an dieser Stelle von Interesse sind. Abbildung 6.16 (a) stellt die Resultate für ZEITDRUCK? dar. Wie bei den drei unabhängigen Variablen des Anweisungsexperiments, zeigt das individuelle Modell auch hier eine schlechte Performanz, obwohl es hinsichtlich der Vorhersage der abhängigen Variablen in den letzten Blöcken gute Ergebnisse produzieren konnte. Wie bei der Variablen ANZAHL DER ANWEISUNGEN des Anweisungsexperiments kann das parametrisierte Benutzermodell einen Vorteil gegenüber dem allgemeinen Ansatz erlangen. Dieser Vorteil ist darauf zurückzuführen, dass hier zwei Symptome vorliegen, die deutliche individuelle Unterschiede aufweisen.

Die Resultate zu NAVIGATION? (Abbildung 6.16 (b)) stellen ähnlich schlechte Ergebnisse der Klassifikationsaufgabe wie beim Anweisungsexperiment dar: Das individuelle Modell verhält sich sehr schlecht, das parametrisierte und das differentiell adaptive können keine Verbesserung im Vergleich zum allgemeinen Benutzermodell erreichen—obwohl sie bei der Vorhersage der abhängigen Variablen zumindest gleich gut waren, in einigen Fällen sogar deutlich bessere Resultate erzielen konnten.

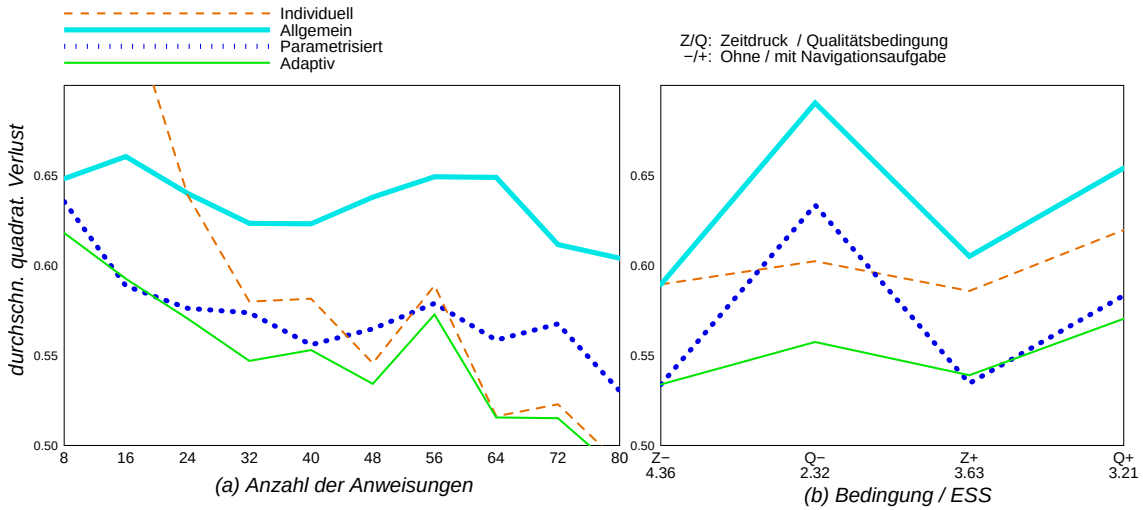


***Abbildung 6.12**: Vorhersagegenauigkeit für die Variable* SILBENANZAHL

6.4.2.3 Diskrepanz zwischen Vorhersage und Klassifikation

Insgesamt hat sich in den vorangehenden Evaluationen die Adaption an den individuellen Benutzer bei Klassifikationsaufgaben im Vergleich zur Vorhersage der abhängigen Variablen als weniger erfolgreich erwiesen. Insbesondere das individuelle Benutzermodell war nicht in der Lage, zu den anderen Adaptionsansätzen aufzuschließen. Auch haben das parametrisierte und das (differentiell) adaptive Modell ein geringeres Ausmaß an Verbesserung gezeigt.

Diese Beobachtung ist darauf zurückzuführen—wie in vielen Arbeiten berichtet wird—, dass es nicht notwendigerweise ein „bestes“ Modell zur Modellierung einer Datenmenge gibt. Beispielsweise diskutieren Friedman et al. (1997) die Gründe, warum ein Bayes’sches Netz, das hinsichtlich des globalen Kriteriums der Likelihood der Daten als optimal zu erachten ist, im Sonderfall der Klassifikation typischerweise nur suboptimale Ergebnisse erzielt: Im Wesentlichen beruht der Effekt darauf, dass ein auf Klassifikationsaufgaben spezialisiertes Bewertungskriterium als ein Anteil der Likelihood betrachtet werden kann und somit beim Lernen mit der allgemeineren Bewertungsfunktion—der Likelihood—andere Aspekte auf Kosten der Klassifikationsfähigkeit optimiert werden. Eine allgemeinere Sichtweise vertreten Greiner et al. (1997). Sie argumentieren, dass beim Lernen explizit die möglichen Anfragen berücksichtigt werden sollen, die zur Laufzeit des Systems anfallen und vom Bayes’schen Netz bearbeitet werden müssen.

Diese Problematik spielt in den meisten Anwendungsszenarien nur eine untergeordnete Rolle, da beim Einsatz Bayes’scher Netze typischerweise Mischformen von Anfragen auftreten, die teilweise Vorhersage- und Klassifikationsaufgaben wahrnehmen und somit mit der Likelihood-Bewertungsfunktion behandelt werden können.

6.5 Zusammenfassung und Diskussion

Tabelle 6.3 fasst die Diskussion der vorangehenden Abschnitte hinsichtlich (a) der theoretischen Stärken und Schwächen der Adaptionsvarianten und (b) der empirischen Resultate, die von ihnen

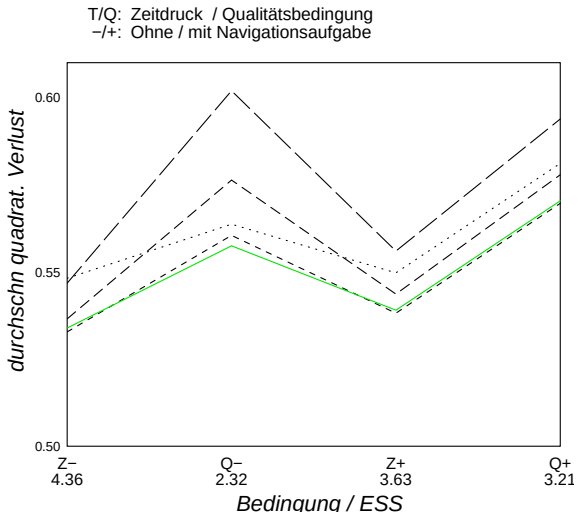


***Abbildung 6.13**: Vergleich manuell spezifizierter, globaler ESS und der differentiellen Adaption*

mit den Daten der beiden Experimente erzielt wurden, zusammen. Die Konsistenz der Ergebnisse gibt Grund zur Annahme, dass sie nicht auf eine spezifische Anwendungssituation beschränkt sind.

Tabelle 6.3 beinhaltet zusätzlich praktische Gesichtspunkte, die zu einer Entscheidung für den Einsatz eines der Modelle in einem gegebenen Szenario beitragen können. Im Wesentlichen handelt es sich um drei Kategorien:

1. die Menge der benötigten empirischen Daten und/oder des A-priori-Wissens
2. Anforderungen des Einsatzszenarios, beispielsweise unterstützt ein mobiles Gerät meist keine ähnlich ressourcen-intensive Berechnungen wie ein stationäres System.
3. die Möglichkeit, Langzeit-Benutzermodelle zu erheben, die in späteren Interaktionen und/oder anderen Anwendungsszenarien verwendet werden können.

Obwohl der parametrisierte und der differentiell adaptive Ansatz insgesamt die besten Ergebnisse aufweisen, können das allgemeine und das individuelle Modell in bestimmten Situationen durchaus vergleichbare Ergebnisse erzielen. Deshalb kann sich eines der beiden letztgenannten möglicherweise als beste Lösung herausstellen, wenn die entsprechenden praxis-relevanten Kriterien erfüllt werden.

Mit den in diesem Kapitel ausführlich untersuchten existierenden bzw. neu entwickelten Verfahren zur Adaption von Benutzermodellen in Form Bayes'scher Netze steht den Entwicklern benutzeradaptiver Systeme eine Sammlung alternativer Methoden zur Verfügung. Diese Verfahren können zentrale Bausteine der generischen Konzeption zum maschinellen Lernen Bayes'scher Netze für benutzeradaptive Systeme bedarfsgerecht instanziieren.

Das vorgestellte Verfahren der differentiellen Adaption Bayes'scher Netze sollte durch seine aus dem Benutzermodellierungskontext heraus motivierte Vorgehensweise in vielen Fällen die vorhandenen Interaktionsdaten besser für den zur Laufzeit anfallenden Adaptionsprozess ausnutzen als die bislang eingesetzen Alternativen.

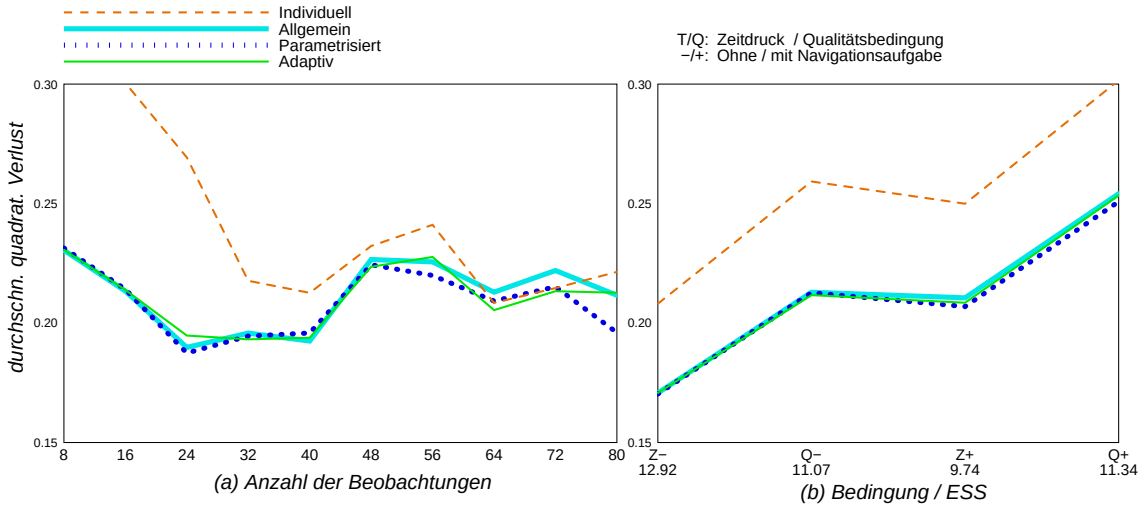


***Abbildung 6.14**: Vorhersagegenauigkeit für die Variable* QUALITÄTSSYMPTOME

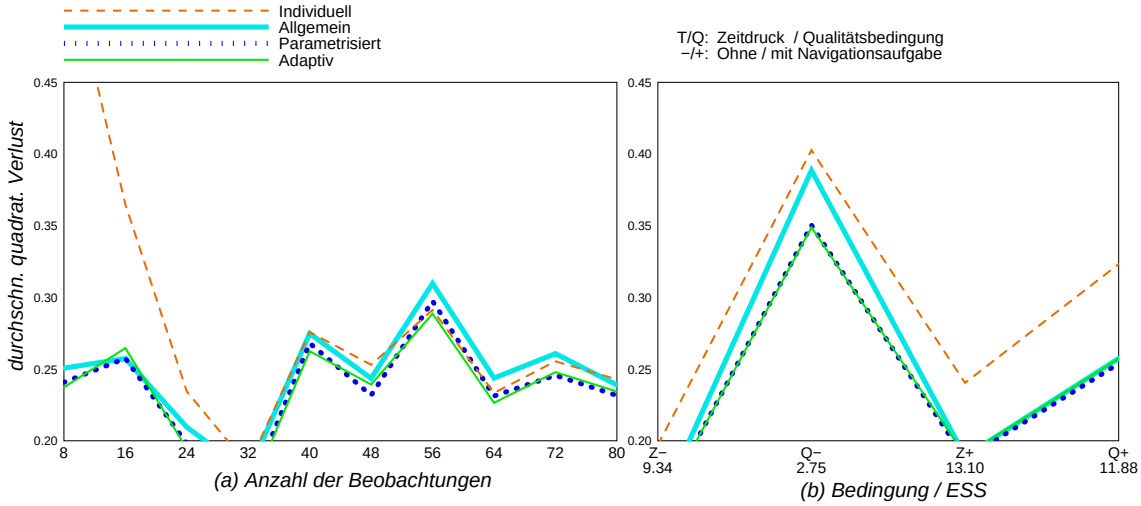


***Abbildung 6.15**: Vorhersagegenauigkeit für die Variable* STILLE PAUSEN

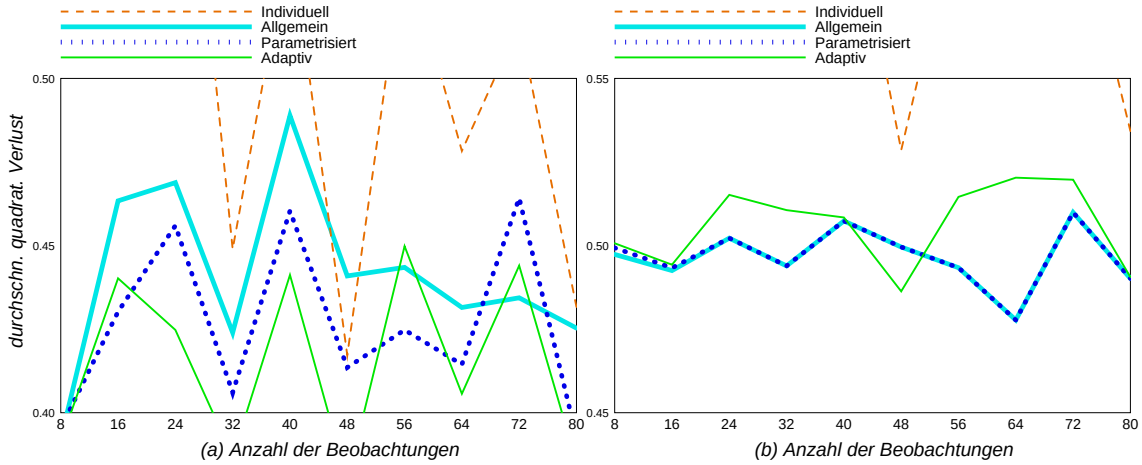


***Abbildung 6.16**: Klassifikationsgenauigkeit für die Variablen* ZEITDRUCK? *und* NAVIGATION?

Modell	Theoretische Aspekte	Empirische Ergebnisse	Praktische Aspekte
Individuell	- Da weder vorhandene Daten noch A-priori-Wissen ausgenutzt werden, sind die Inferenzergebnisse zu Beginn der Interaktion typischerweise sehr schlecht + Es können auch unerwartete Verhaltensweisen adäquat modelliert werden	- Sehr schlechte Ergebnisse in der initialen Einsatzphase - Schlechte Phase ist insbesondere bei Klassifikationsaufgaben sehr lang + Sehr gute finale Ergebnisse bei idiosynkratischen Verhaltensweisen	+ Kein A-priori-Wissen oder empirische Daten notwendig - Wiederholte Anwendung des Adaptionsmechanismus zur Laufzeit
Allgemein	- Keine Berücksichtigung individueller Unterschiede	- Auf lange Sicht schlechter als das parametrisierte und das adaptive Modell (manchmal auch als das individuelle Modell), außer in Situationen, in denen die individuellen Unterschiede schwierig zu erlernen sind	- Ausreichende Menge an empirischen Daten benötigt + Kein zusätzlicher Aufwand zur Laufzeit für den Adaptionsmechanismus
Parametrisiert	+ Wissen über die Art der individuellen Unterschiede muss explizit repräsentierbar sein - Viele Parameter benötigt bei komplexen individuellen Unterschieden	+ I.A. besser als das allgemeine und das individuelle Modell, manchmal ähnlich gut wie der adaptive Ansatz - Etwas schlechter als das adaptive oder individuelle Modell, wenn die individuellen Unterschiede komplex sind	- Ausreichende Menge an empirischen Daten benötigt - Die Verwendung dynamischer Bayes'scher Netze kann zu Komplexitätsproblemen führen + Individuelle Parametervariablen können auch in anderen Kontexten genutzt werden
Differentiell adaptiv	- Unterschiedliche Teile des Benutzermodells werden mit unterschiedlichen Geschwindigkeiten adaptiert + Erlaubt fließenden Übergang vom allgemeinen zum individuellen Modell - Anzahl der Freiheitsgrade des Lern- bzw. Adaptionsprozesses kann im Vergleich zum parametrisierten Ansatz unnötigerweise hoch sein	+ I.A. gute Performanz, insbesondere bei komplexen individuellen Unterschieden	- Ausreichende Menge an empirischen Daten benötigt + Kein *explizites* A-priori-Wissen über die Art der individuellen Unterschiede notwendig - Wiederholte Anwendung des Adaptionsmechanismus zur Laufzeit

***Tabelle 6.3**: Überblick der Vor- und Nachteile der alternativen Adaptionsansätze*

7 Strukturelles Lernen und strukturelle Adaption Bayes'scher Netze

In den vorangegangenen Kapiteln der vorliegenden Arbeit wurden Techniken zum Erlernen bzw. zur Adaption der bedingten Wahrscheinlichkeiten der als Benutzermodell verwendeten Bayes' schen Netze vorgestellt und diskutiert. Obwohl bzw. gerade weil es sich dabei aus praktischen Gesichtspunkten (vgl. auch Abschnitt 2.6) um die häufiger bearbeitete Teilaufgabe des maschinellen Lernproblems Bayes'scher Netze in benutzeradaptiven Systemen handelt, wird in diesem Kapitel untersucht, ob und gegebenenfalls inwieweit Verfahren des strukturellen Lernens Bayes'scher Netze sinnvoll im Benutzermodellierungskontext angewendet werden können.

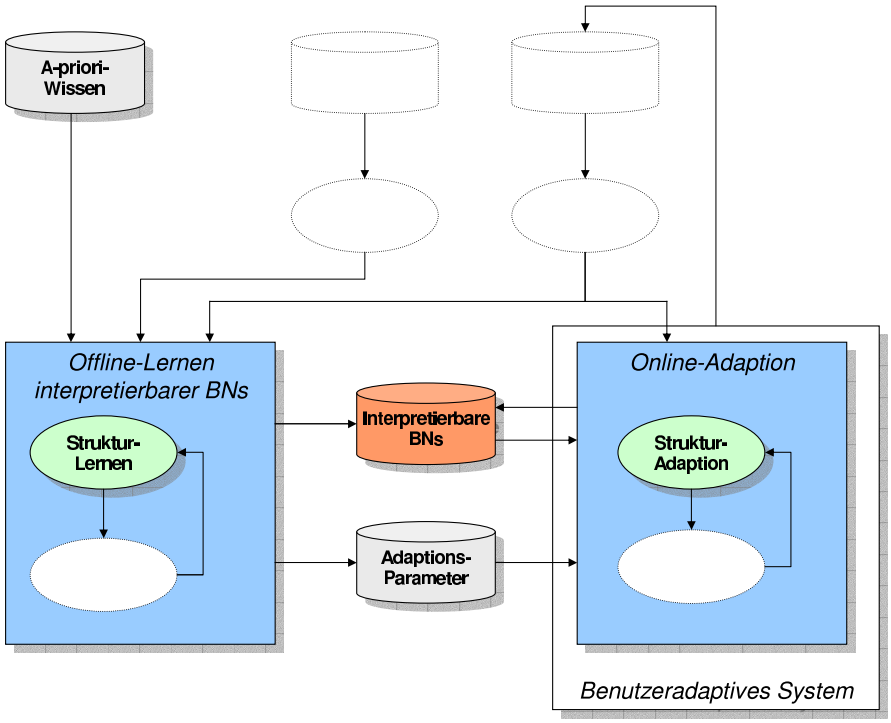


***Abbildung 7.1**: Einordnung des strukturellen Lernens und der strukturellen Adaption in die integrative Konzeption*

Konkrete Inhalte dieses Kapitels sind empirische Untersuchungen zu strukturellen Lernverfahren—sowohl mit dem Ziel der Performanzverbesserung der Benutzermodelle als auch im Sinne der Wissensentdeckung, um interessante, relevante Aspekte der Domänen zu identifizieren, die im Konstruktionsprozess eines Systems eine Rolle spielen. Im zweiten Teil des Kapitels wird das Adaptionsproblem der Struktur Bayes'scher-Netz-Benutzermodelle thematisiert. In diesem Rahmen wird ein neues strukturelles Adaptionsverfahren vorgestellt und evaluiert.

7.1 Strukturelles Lernen Bayes'scher Netze zur Akquisition der Benutzermodelle

Bislang wird beim Einsatz maschineller Lernverfahren Bayes'scher Netze in benutzeradaptiven Systemen dem Strukturfall kaum Beachtung geschenkt. Dies spiegelt sich auch in den in Abschnitt 2.6 im Rahmen des Überblicks des aktuellen Standes der Forschung angeführten Beispielsystemen wider: Meist wird sich auf das Erlernen der bedingten Wahrscheinlichkeiten bei Vorgabe einer festen Struktur konzentriert. Die Struktur wird dabei—falls notwendig in Zusammenarbeit mit Domänenexperten—manuell erstellt. In den wenigen Fällen, in denen strukturelle Aspekte intensiver untersucht werden, beschränkt sich dies weitgehend auf das Testen alternativer, manuell konstruierter Strukturen. Nur die neueren Arbeiten von Nicholson et al. (2001) und Horvitz et al. (2002) setzen explizit Strukturlernverfahren im Konstruktionsprozess der Benutzermodelle in Form Bayes'scher Netze ein.

Die Tatsache, dass der Strukturfall seltener behandelt wird, kann auf mehrere Gründe zurückgeführt werden, u.a.:

- In vielen Szenarien benutzeradaptiver Systeme ist es unter Ausnutzung der kausalen Interpretation der Kanten sehr einfach möglich, eine plausible Struktur zu spezifizieren—sei es durch einen Domänenexperten oder den Systementwickler.
- Die Qualität einer aufgrund der kausalen Interpretation vorgegebenen Struktur genügt oftmals bereits den gestellten Anforderungen. Die zugehörigen bedingten Wahrscheinlichkeiten können gegebenenfalls in der üblichen Weise maschinell erlernt werden.
- Die hohe Komplexität der Strukturlernverfahren insbesondere bei unvollständigen Trainingsdaten macht einen sinnvollen Einsatz oft unmöglich oder zumindest sehr aufwendig.
- Im Gegensatz zu einer Vielzahl existierender Implementationen von Lernverfahren für die bedingten Wahrscheinlichkeiten, gibt es bislang deutlich weniger Standardsoftwarepakete für Bayes'sche Netze, die Strukturlernverfahren anbieten. Diese Situation befindet sich zur Zeit im Umbruch, so dass in den nächsten Jahren mit einem verstärkten Einsatz struktureller Lernverfahren in der Praxis zu rechnen ist.

Es stellt sich also die Frage:

Macht der Einsatz maschineller Lernverfahren zum Erlernen der Struktur Bayes'scher Netze für benutzeradaptive Systeme überhaupt Sinn?

Diese zentrale Frage wird im Weiteren anhand des Beispiels des Flughafenexperiments untersucht.

7.1.1 Einbringen von A-priori-Wissen beim strukturellen Lernen

Wegen der hohen Dimensionalität des Lösungsraums ist es gerade beim strukturellen Lernen von Bedeutung, das zur kausalen Struktur der zu modellierenden Domäne vorhandene A-priori-Wissen in den Lernprozess einzubringen und damit die Interpretierbarkeit des erlernten Modells zu gewährleisten bzw. zu verbessern (siehe Abschnitt 3.1.3.7). Dies ist zumindest mit den im Folgenden aufgelisteten Ansätzen möglich (vgl. auch Wittig, 2001a):

- *Vorgabe einer mit dem vorhandenen Wissen konformen bzw. das vorhandene Wissen kodierenden Ausgangsstruktur für den Suchprozess im Raum der möglichen Strukturen:* Die bereits bekannte, zugrunde liegende Annahme ist dabei, dass die „richtige“ Struktur der vorgegebenen ähnlich ist und deshalb erwartet werden kann, dass sie in der Nachbarschaft im Suchraum angesiedelt ist. Insbesondere kann mit der Spezifikation der Ausgangsstruktur auch die Existenz verborgener Variablen vorgegeben werden.

- *Vorgabe struktureller Constraints für den Lernvorgang:* Solche *strukturellen Constraints* betreffen Teile bzw. Aspekte des den Ausgangspunkt der Suche bildenden Bayes'schen Netzes, die als korrekt angenommen werden, und im Rahmen des Lernens nicht modifiziert werden dürfen. Beispiele hierfür sind Vorgaben, die das (Nicht-)Vorhandensein einzelner Kanten oder das Fehlen von Eltern betreffen, wie es z.B. bei unabhängigen Variablen einer Experimentalsituation der Fall ist. Die Vorgabe struktureller Constraints resultiert in einer Einschränkung des Suchraums.

- *Explizite Modellierung individueller Unterschiede durch individuelle Parametervariablen:* Die Spezifikation von individuellen Parametervariablen erfordert Wissen über das Vorhandensein und die Art der individuellen Unterschiede (vgl. Tabelle 6.3). Diese Art von Wissen ist im Gegensatz zu den beiden vorher genannten Formen von A-priori-Wissen seltener verfügbar.

- *Anwenden einer auf dem Bayes'schen Lernansatz basierenden Bewertungsfunktion:* Wegen der mit dem Bayes'schen Ansatz verbundenen Notwendigkeit der aufwendigen Spezifikation einer A-priori-Wahrscheinlichkeitsverteilung über allen potenziell möglichen Strukturen (siehe Abschnitt 4.4.2), kommt diese Möglichkeit in der Praxis selten zum Einsatz. Meist wird in diesem Fall eine einfach vorzugebende A-priori-Wahrscheinlichkeitsverteilung verwendet, wie z.B. eine Gleichverteilung, mit der alle Strukturen a priori als gleichwahrscheinlich einschätzt werden. Eine Alternative ist die Vorgabe einer wahrscheinlichsten Struktur in Kombination mit „Bestrafungen“ abweichender Struktureigenschaften, wie etwa das Fehlen von Kanten. Solche Strukturen sind dann a priori weniger wahrscheinlich als die vorgegebene.

Jede dieser Möglichkeiten des Einbringens verfügbaren A-priori-Wissens kann mit dem SEM-Algorithmus aus Abschnitt 4.4.3 realisiert werden. Da der SEM-Algorithmus zusätzlich in der Lage ist, mit verborgenen Variablen und fehlenden Daten umzugehen, bietet er sich für einen Einsatz in benutzeradaptiven Systemen an.

7.1.2 Beispiel: Flughafenexperiment

Anhand der im Rahmen des Flughafenexperiments gesammelten Daten wurde eine Studie durchgeführt, welche die Ergebnisse des strukturellen Lernens mit denjenigen ohne den Einsatz von

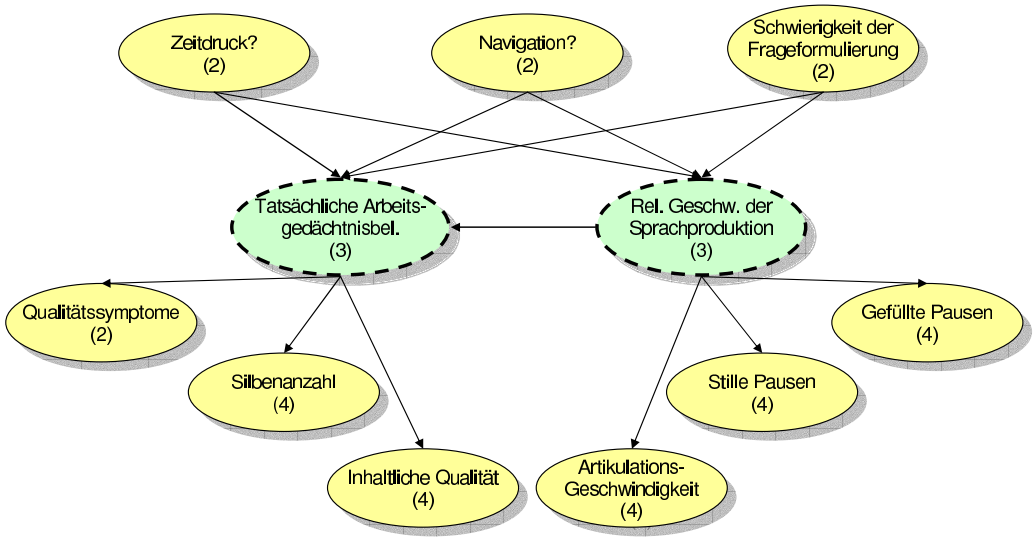


***Abbildung 7.2**: Ausgangsstruktur des strukturellen Lernprozesses am Beispiel des Flughafenexperiments*

Strukturlernverfahren vergleicht. Es wurde für beide Fälle analog zur grundlegenden Verfahrensweise in Abschnitt 5.3.2.1 jeweils eine 32fache Leave-one-out-Kreuzvalidierung durchgeführt—, sowohl beim strukturellen Lernen mit dem SEM-Algorithmus als auch in der zweiten Situation mit fest vorgegebener Struktur, in der nur die CPTs unter Verwendung des EM-Algorithmuses erlernt wurden. Als Ausgangsstruktur des Lernvorgangs bzw. fest vorgegebene Struktur wurde die in Abbildung 7.2 dargestellte verwendet. Als strukturelle Constraints zur Kodierung des A-priori-Wissens wurde gefordert, dass die unabhängigen Variablen des Experiments während des Lernprozesses keine Elternvariablen erhalten durften, d.h., dass sie auch im Resultat des Lernvorgangs unabhängig bleiben mussten.

Abbildung 7.3 zeigt die erzielten Ergebnisse bei einer Bewertung mit der durchschnittlichen negativen Log-Likelihood der Daten pro Testfall, d.h., der Fähigkeit der erlernten Netze, die (Test-)Daten zu repräsentieren. Um die Vergleichbarkeit der Resultate zu gewährleisten, wurde der EM-Algorithmus in der Kreuzvalidierung jeweils nach 100 Iteration sowie der SEM-Algorithmus nach jeweils 20 inneren EM-Iterationen (vgl. Abbildung 4.4) und fünf strukturellen Änderungen beendet ($5 \times 20 = 100$). In der Mehrzahl der 32 Kombinationen aus Trainings- und Testdaten führte der SEM-Algorithmus mehr als fünf strukturelle Veränderungen—verbunden mit weiteren Verbesserungen der Modellierung—durch. Um die auf allen 32 Kombinationen der Kreuzvalidierung basierenden Durchschnittswerte präsentieren zu können, erfolgte eine Beschränkung der beiden Kurven auf den minimal auftretenden Wert von fünf Strukturmodifikationen.

Die Resultate weisen eine Überlegenheit des strukturellen Lernens nach einer initialen Phase von zwei strukturellen Veränderungen auf. Die vom SEM-Algorithmus anhand der empirischen Daten durchgeführten strukturellen Modifikationen der auf der Basis theoretischer Überlegungen spezifizierten Ausgangsstruktur ermöglichen eine erhöhte Qualität der quantitativen Modellierung der gemeinsamen Wahrscheinlichkeitsverteilung. Es ist zu beachten, dass beim SEM-Lernvorgang nicht alleine diese Eigenschaft des zu erlernenden Bayes'schen Netzes optimiert wird, sondern mit dem BIC ein Tradeoff zwischen (Log-)Likelihood der Daten und Modellkomplexität zur Bewertung der untersuchten Bayes'schen Netze verbunden ist. Dies wirkt sich in der initialen Lernpha-

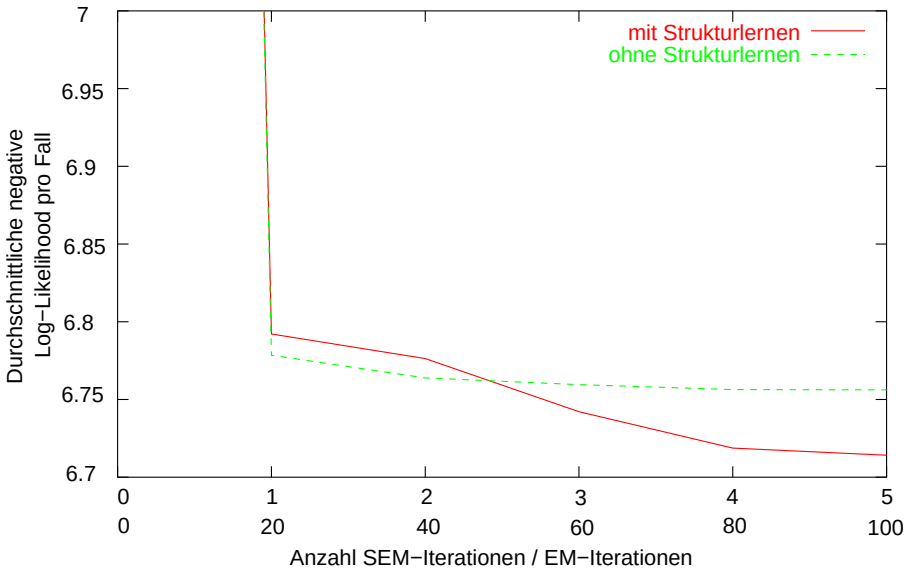


***Abbildung 7.3**: Vergleich der Ergebnisse mit vs. ohne strukturelles Lernen*

se, bestehend aus den ersten beiden Iterationen des SEM-Algorithmuses, aus: Zu Beginn werden in diesem Beispielszenario im Zuge des Genauigkeits-Komplexitäts-Tradeoffs Kanten entfernt, die unter Verwendung vieler bedingter Wahrscheinlichkeiten „schwache“ quantitative Zusammenhänge modellieren. Mit diesen Kantenlöschungen ist eine etwas schlechtere Modellierung der gemeinsamen Wahrscheinlichkeitsverteilung verbunden, die zugunsten der effizienteren Kodierung in einem einfacheren Bayes'schen Netz in Kauf genommen wird. Hier wird zu Beginn des Lernvorgangs eine mit der Entfernung der entsprechenden Kanten verbundene Isolierung der unabhängigen Variablen NAVIGATION? beobachtet. Dies korrespondiert mit den in Abschnitt 2.4.2 beschriebenen Ergebnissen, die auf—in Relation zum ausgeübten Zeitdruck—geringere Effekte der Navigationsaufgabe auf die Sprachsymptome hindeuten. Nach der initialen Lernphase werden vermehrt neue Kanten in die Struktur eingebracht, die das Netz in die Lage versetzen, die zugehörigen quantitativen Zusammenhänge explizit in den neuen CPTs zu repräsentieren und die somit zu einer verbesserten Kodierung der gemeinsamen Wahrscheinlichkeitsverteilung durch das Bayes'sche Netz beitragen. So wird beispielsweise in vielen der 32 Fälle eine Kante zwischen zwei zusammenhängenden Sprachsymptomen wie etwa SILBENANZAHL und ARTIKULATIONSGESCHWINDIGKEIT eingefügt.[1] Diese Änderungen korrigieren fehlerhafte Annahmen und/oder unvollständige Aspekte des Ausgangsmodells und tragen zu besseren Inferenzergebnissen bei. Abbildung 7.4 zeigt eine prototypische Struktur, die im Rahmen der Leave-one-out-Kreuzvalidierung vom SEM-Algorithmus erlernt wurde.

Insgesamt zeigt sich anhand des Beispiels des Flughafenexperiments, dass sich der Einsatz struktureller Lernverfahren in diesem Szenario lohnt, um die Qualität der erlernten Modelle zu erhöhen. Allgemein bietet es sich in diesem Zusammenhang an, eine vorhandene Vorstellung des Benutzermodells der Domänenexperten mit Hilfe von Strukturlernverfahren anhand der verfügbaren empirischen Daten der „Realität“ anzupassen. Dabei ist im Rahmen einer Kosten-Nutzen-

[1] Die zugehörige allgemeine Beobachtung besteht darin, dass Personen, die schnell reden, meist auch viel artikulieren, und umgekehrt.

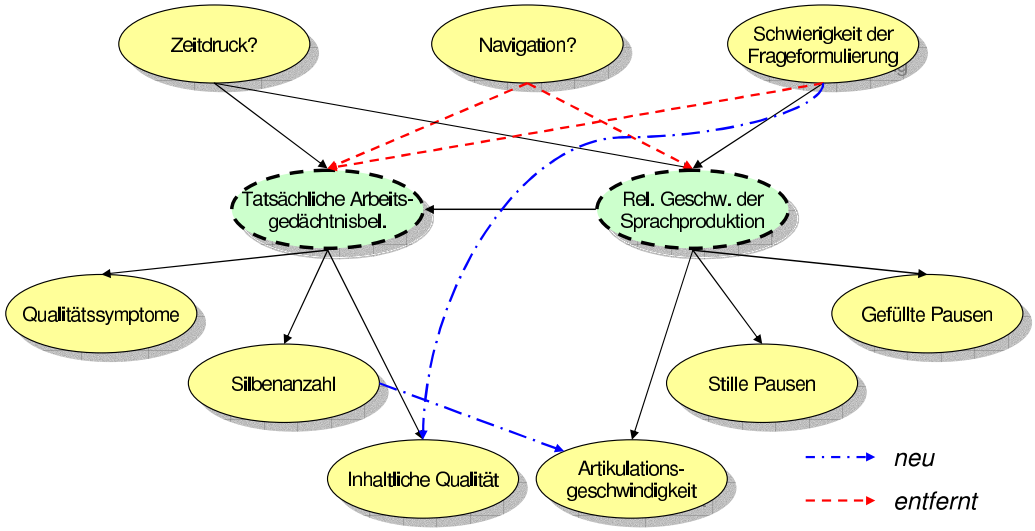


***Abbildung 7.4**: Typisches Resultat des strukturellen Lernprozesses*

Analyse abzuwägen, ob sich der zusätzliche Aufwand des Einsatzes der vergleichsweise komplexen Algorithmen im jeweiligen betrachteten Szenario auszahlt. Dies spielt insbesondere bei der Verwendung verborgener Variablen eine bedeutende Rolle, da in diesem Fall bereits die Teilaufgabe des Lernens der bedingten Wahrscheinlichkeiten nur mit rechenintensiven Methoden wie dem EM- oder APN-Algorithmus bearbeitet werden kann.

7.1.3 Strukturelle Aspekte bei der Erkennung kognitiver Ressourcenbeschränkungen mit empirisch basierten dynamischen Bayes'schen Netzen

Mit den in der vorangehenden Studie erzielten Resultaten konnte am Beispiel des Flughafenexperiments gezeigt werden, dass Strukturlernverfahren in bestimmten Situationen in der Lage sind, die Modellierung der gemeinsamen Wahrscheinlichkeitsverteilung im Vergleich zu CPT-Lernverfahren zu verbessern. Ob und in welcher Weise sich diese verbesserte Modellierung gegebenenfalls in der Performanz der Systeme bei der Inferenz niederschlägt, soll im Folgenden untersucht werden (vgl. Wittig, 2001b).

7.1.3.1 Methode

Zur Untersuchung dieser Fragestellung wurde die in Abschnitt 2.4.2 beschriebene Studie zur Erkennung kognitiver Ressourcenbeschränkungen anhand von Symptomen der gesprochenen Sprache reproduziert. Die folgenden Varianten der Untersuchung unterscheiden sich lediglich in der im Rahmen der Leave-one-out-Kreuzvalidierung verwendeten jeweiligen Netzstruktur bzw. den eingesetzen maschinellen Lernverfahren zur Ermittlung der Zeitscheiben des dynamischen Bayes' schen Netzes. Die zugrunde liegende Evaluationsprozedur, wie sie in Tabelle 2.3 beschrieben wurde, blieb in den im Folgenden diskutierten Untersuchungen erhalten.

7.1.3.2 Einbringen verborgener Variablen

In einem ersten Schritt der Gesamtstudie wurden die Auswirkungen des Einbringens verborgener Variablen in die gemeinsame Struktur der verwendeten Zeitscheiben (ohne die Verwendung individueller Parametervariablen) untersucht. Es wurden lediglich die bedingten Wahrscheinlichkeiten der CPTs mit dem EM-Algorithmus (jeweils 50 Iterationen) gelernt—Strukturlernen wurde nicht durchgeführt.

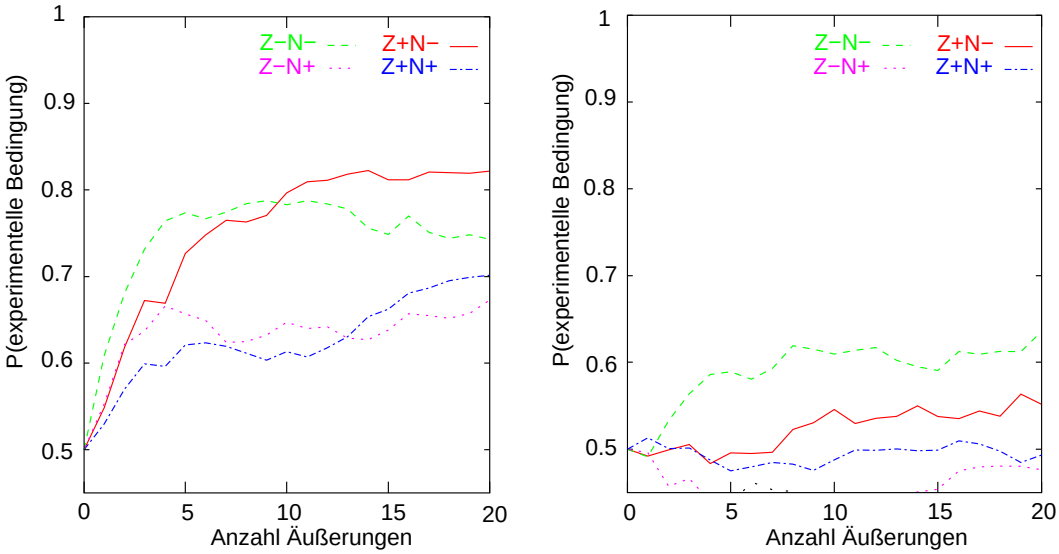


***Abbildung 7.5**: Erkennungsleistung mit verborgenen Variablen*
(Z+ / Z-: Zeitdruck vorhanden / nicht vorhanden, N+ / N-: Navigationsaufgabe vorhanden / nicht vorhanden)

Die in Abbildung 7.5 dargestellten Ergebnisse zeigen qualitativ ähnliche Eigenschaften, wie sie in der Analyse in Abschnitt 2.4.2 mit einem voll beobachteten Bayes'schen Netz erzielt wurden. Aus diesem Grund wird an dieser Stelle auf eine nochmalige Diskussion der allgemeinen Resultate verzichtet und nur auf die für die Betrachtung des Strukturfalls interessanten Unterschiede fokussiert. Insgesamt ist die Erkennungsleistung zwar quantitativ geringfügig schlechter, was aber im Wesentlichen auf das Fehlen der individuellen Parametervariablen zurückzuführen ist (wie in Abschnitt 7.1.3.4 gezeigt wird). Eine erhöhte Interpretierbarkeit der Benutzermodelle durch verborgene Variablen wird hier also nicht auf Kosten der Qualität der Inferenzergebnisse erkauft.

7.1.3.3 Einsatz von Strukturlernverfahren

Wird im Vergleich zur Studie des vorhergehenden Abschnitts zusätzlich der SEM-Algorithmus zum Erlernen der Struktur der Zeitscheibe in Kombination mit verfügbarem A-priori-Wissen angewendet, ergeben sich die Resultate aus Abbildung 7.6. Dabei wurden folgende strukturelle Constraints für den Lernprozess vorgegeben:

- Die drei unabhängigen Variablen des Experimentaldesigns ZEITDRUCK?, NAVIGATION? und SCHWIERIGKEIT DER FRAGESTELLUNG mussten elternlos bleiben.

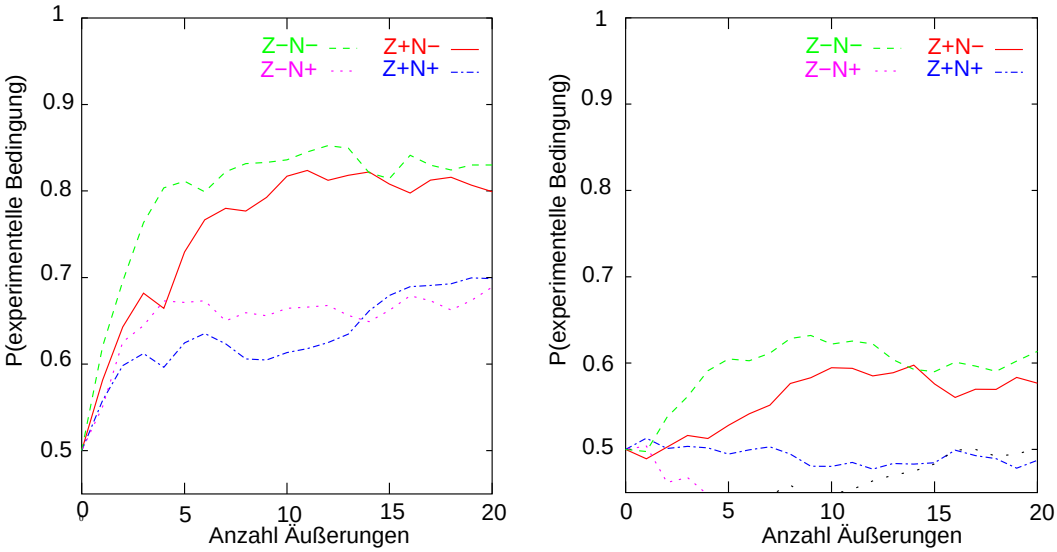


***Abbildung 7.6**: Erkennungsleistung mit verborgenen Variablen und Strukturlernen*

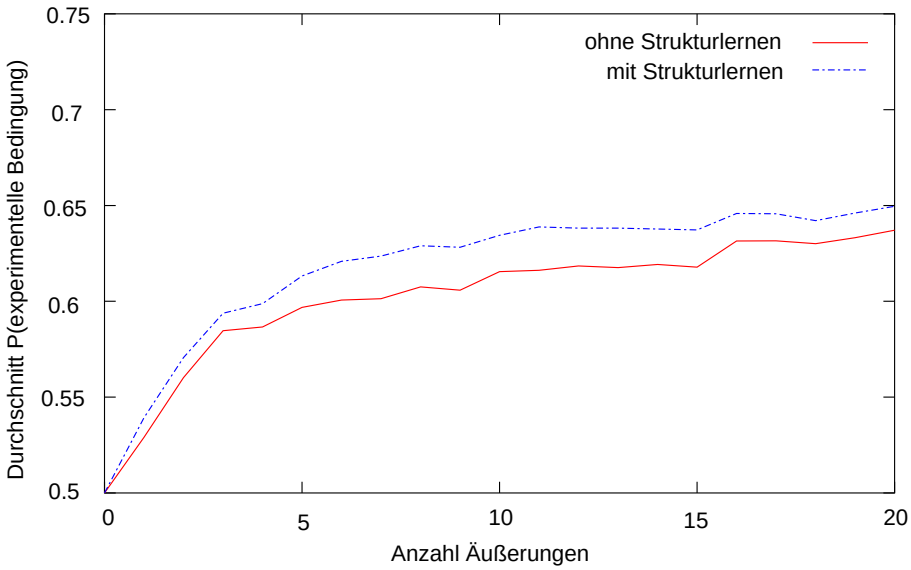


***Abbildung 7.7**: Durchschnittliche Erkennungsleistung mit verborgenen Variablen und Strukturlernen, gemittelt über beide unabhängigen Variablen und alle experimentellen Bedingungen*

- Die Kanten von ZEITDRUCK? und NAVIGATION? zu den beiden verborgenen Variablen TATSÄCHLICHE ARBEITSGEDÄCHTNISBELASTUNG und RELATIVE GESCHWINDIGKEIT DER SPRACHPRODUKTION mussten erhalten bleiben. Dies ist insbesondere hinsichtlich der von NAVIGATION? ausgehenden Kanten wichtig, da bekannt ist, dass die ablenkende Nebenaufgabe eine zusätzliche Belastung darstellt. Dieser Einfluss wird aber aufgrund des Genauigkeit-Komplexitäts-Tradeoffs des BIC und des geringen tatsächlichen Ausmaßes der Zusatzbelastung im Experiment aus dem Modell entfernt, wie in der in Abschnitt 7.1.1 beschriebenen Studie beobachtet wurde.

- Die Existenz der Kante zwischen RELATIVE GESCHWINDIGKEIT DER SPRACHPRODUKTION und TATSÄCHLICHE ARBEITSGEDÄCHTNISBELASTUNG wird gefordert, da sie die zentrale Annahme des zugrunde liegenden Modells, wie sie in Abschnitt 2.2.2.4 formuliert wurde, repräsentiert.

Die Anwendung des SEM-Algorithmuses in Kombination mit dem vorhandenen Hintergrundwissen führt zu einer Verbesserung der Erkennungsleistung des Modells (Abbildung 7.7).

Im Verlauf des Lernvorgangs wurden hier zwei strukturelle Veränderungen vorgenommen: (a) Entfernen der Kante von SCHWIERIGKEIT DER FRAGESTELLUNG zu TATSÄCHLICHE ARBEITSGEDÄCHTNISBELASTUNG und (b) Einfügen einer Kante von SILBENANZAHL zu ARTIKULATIONSGESCHWINDIGKEIT. Beide Änderungen sind nachvollziehbar: Im ersten Fall genügt eine Kante zwischen SCHWIERIGKEIT DER FRAGESTELLUNG und den beiden verborgenen Variablen, um die Auswirkungen der Komplexität der Anfragegenerierung in das Modell einzubringen. Die zusätzlich zwischen den beiden Sprachsymptomen aufgenommene direkte Verbindung dokumentiert den auch in der statistischen Analyse (vgl. Abschnitt 2.2.2) beobachteten starken Zusammenhang dieser beiden Variablen.

7.1.3.4 Einbringen individueller Parametervariablen

Durch das Hinzufügen einer individuellen Parametervariable zu jeder der Sprachsymptomvariablen (vgl. Abschnitt 2.4.2), wurden die Erkennungsleistungen der Abbildungen 7.8 und 7.9 im Fall ohne bzw. mit Strukturlernen erzielt.

Wie anhand Abbildung 7.10 zu erkennen ist, führt der Einsatz individueller Parametervariablen im Szenario des Flughafenexperiments zu einer deutlichen Steigerung der durchschnittlichen Erkennungsleistung. Dies war zu erwarten, da bekannt ist, dass Personen hinsichtlich der Produktion von Sprachsymptomen typische Unterschiede aufweisen. Diese Unterschiede sind aber nicht von solch heterogener Natur, dass individuelle Modelle benötigt würden, die nur anhand der Interaktionsdaten des Individuums konstruiert werden. Es genügt, die Benutzermodelle hinsichtlich der wesentlichen Aspekte zu parametrisieren.

Wider Erwarten kann die Anwendung eines Strukturlernverfahrens bei Verwendung individueller Parametervariablen keine Performanzsteigerung bewirken. Zum Teil verläuft die Erkennungsleistung sogar schlechter. Eine mögliche Erklärung besteht in der höheren Modellkomplexität durch das Einbringen der individuellen Parametervariablen: Es kann die Situation eintreten, dass es für den SEM-Algorithmus nicht mehr möglich ist, anhand der begrenzten Trainingsdaten die erhöhte Anzahl der bedingten Wahrscheinlichkeiten adäquat zu erlernen. Die größere Anzahl der bedingten Wahrscheinlichkeiten entsteht durch die zusätzliche Elternvariable, die jedes Sprachsymptom mit der zugehörigen individuellen Parametervariable bekommt. Für jede der Elternzustandskombinationen stehen weniger Trainingsdaten zur Verfügung, die mit der Zustands-

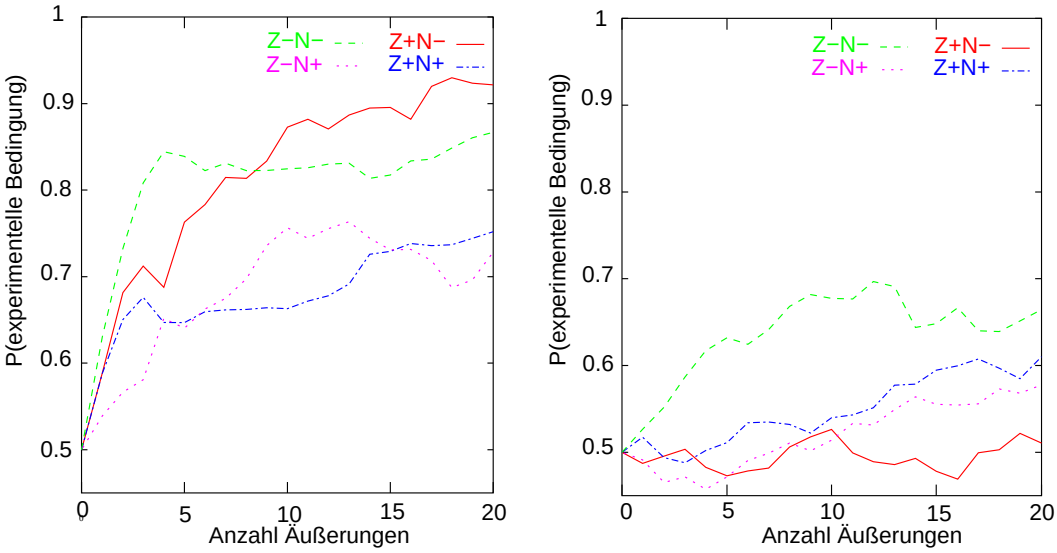


***Abbildung 7.8**: Erkennungsleistung mit verborgenen Variablen und individuellen Parametervariablen*

kombination konsistent sind, d.h., es müssen anhand der gleichen Menge an Trainingsdaten mehr bedingte Wahrscheinlichkeiten erlernt werden, was zu schlechteren Ergebnissen bei den einzelnen bedingten Wahrscheinlichkeiten führt.

7.1.3.5 Zusammenfassende Diskussion der Ergebnisse

Die vorgestellten Ergebnisse zeigen, dass strukturelle Aspekte—sei es die Anwendung von Strukturlernverfahren oder die manuelle Variation von Teilen der Struktur—im Rahmen der Modellkonstruktion eine Rolle für den Erfolg des Systems spielen können. Die in Abschnitt 7.1.1 beobachtete verbesserte Repräsentation der gemeinsamen Wahrscheinlichkeitsverteilung durch strukturelles Lernen wirkt sich hier auch beim Einsatz des erlernten Bayes'schen Netzes in der (simulierten) Anwendungssituation aus. Selbst bei der für ein erlerntes Bayes'sches Netz schwierigen Klassifikationsaufgabe (vgl. Abschnitt 6.4.2.3) wirken sich Modifikationen der Struktur teilweise deutlich in den erzielten Ergebnissen aus. Auch das bereits in der Ausgangsstruktur kodierte große Ausmaß an A-priori-Wissen kann im vorliegenden Beispielszenario durch die Anwendung maschineller Strukturlernverfahren noch durch Anpassung an die verfügbaren empirischen Daten verfeinert werden.

Im Wesentlichen beschränkten sich die Modifikationsmöglichkeiten des SEM-Algorithmuses aufgrund der spezifizierten strukturellen Constraints auf direkte Zusammenhänge bezüglich der Symptomvariablen. In anderen Szenarien, die mehr Freiheitsgrade für die Struktursuche bieten, kann sich der Einsatz von entsprechenden Lernverfahren noch deutlicher auswirken. Betrachtet man die Ergebnisse der unterschiedlichen (Teil-)Studien der vorhergehenden Abschnitte, so konnte hier selbst in einer schwierigen Lernsituation gezeigt werden, dass die Kombination aus Spezifikation eines Ausgangsmodells basierend auf A-priori-Wissen und maschinellem Lernen im Zusammenhang mit Bayes'schen Netzen einen brauchbaren Ansatz darstellt.

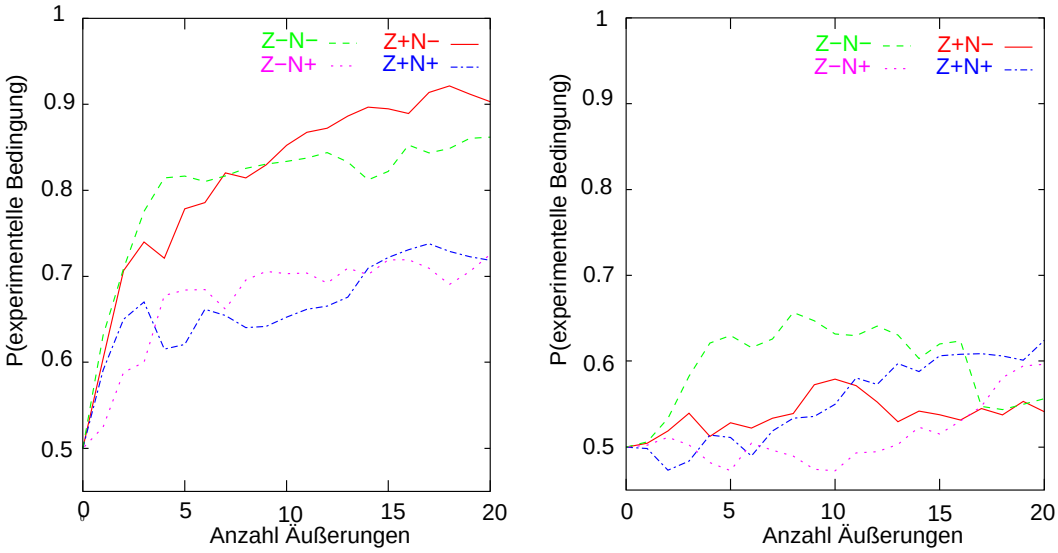


***Abbildung 7.9**: Erkennungsleistung mit verborgenen und individuellen Parametervariablen sowie Strukturlernen*

7.2 Strukturelles Lernen mit Meta-Netzen

Nachdem die vorangegangenen Abschnitte gezeigt haben, dass es sich durchaus lohnen kann, strukturelle Aspekte einer Domäne zu betrachten, sei es durch manuelle Variation der Struktur der eingesetzten Bayes'schen Netze oder durch den Einsatz maschineller Lernverfahren zur Akquisition adäquater Strukturen, wird im Folgenden ein Verfahren vorgestellt und im Benutzermodellierungskontext angewendet, das zu einem detaillierteren Verständnis der behandelten Domäne beitragen kann. Mit seiner Hilfe können Meta-Informationen zu kausalen Beziehungen zwischen den Variablen der Benutzermodelle ermittelt werden, mit deren Hilfe sich beispielsweise Aussagen über beobachtete individuelle Unterschiede zwischen den Benutzern machen lassen.

Das Verfahren wurde von Hofmann (2000) im Rahmen seiner Dissertation entwickelt und stellt die Ausgangsbasis eines im Rahmen der vorliegenden Arbeit neu entwickelten strukturellen Adaptionsverfahrens für Bayes'sche Netze dar.

7.2.1 Motivation: Geringe Menge an verfügbaren Trainingsdaten, Interpretierbarkeit durch explizite Repräsentation der strukturellen Unsicherheit

In der Struktur eines Bayes'schen Netzes wird die gemeinsame Wahrscheinlichkeitsverteilung der Variablen möglichst effizient durch Ausnutzen der (bedingten) Unabhängigkeiten kodiert. Folglich ist es das Ziel der entsprechenden Lernverfahren, diese Struktur der Domäne—falls möglich—eindeutig anhand der vorhandenen empirischen Daten zu identifizieren. Dies wird umso schwieriger, je weniger Trainingsdaten dem eingesetzen Lernverfahren zur Verfügung stehen. Der Overfitting-Effekt ist bei der Strukturlernaufgabe potenziell besonders stark ausgeprägt, da es verglichen mit der CPT-Lernaufgabe zusätzliche freie Parameter zu erlernen gilt.

Die Identifikation einer einzelnen „richtigen" Struktur ist eine sehr schwierige Aufgabe. Bei 12 Variablen existieren nach der in Abschnitt 4.4.2 angegebenen Formel bereits über 10^{20} verschiedene mögliche Strukturen. Es liegt auf der Hand, dass die Suche in einem solchen hochdi-

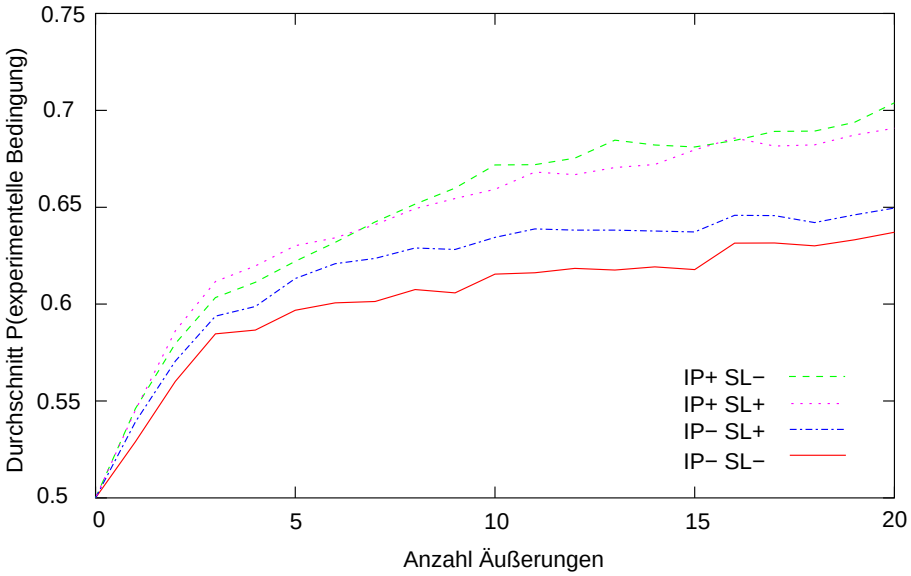


***Abbildung 7.10**: Durchschnittliche Erkennungsleistung mit/ohne verborgenen und individuellen Parametervariablen und Strukturlernen, gemittelt über beide unabhängigen Variablen und alle experimentellen Bedingungen*
(IP+ / IP-: individuelle Parameter vorhanden / nicht vorhanden, SL+ / SL-: mit / ohne Strukturlernen)

mensionalen Raum sehr aufwendig ist und insbesondere bei wenigen Trainingsdaten Gefahr läuft, in einem der vielen lokalen Maxima zu enden.

Eine mögliche Lösung dieses Problems besteht im Erlernen und der Verwendung einer Menge alternativer, „guter“ Strukturen im Rahmen des Model-Averaging-Konzepts (vgl. Abschnitt 4.4.2). Die Ergebnisse der verwendeten Strukturen werden im Inferenzprozess mit unterschiedlichen Gewichten versehen, die anhand ihres bisherigen Erfolgs der Vorhersage bestimmt werden. Im Bayes’schen Ansatz werden als Gewichte die A-posteriori-Wahrscheinlichkeiten der Modelle verwendet. Unter einem solchen Ansatz, der auf einer Menge von Netzen basiert, leidet allerdings die Eigenschaft der Interpretierbarkeit. Es ist selten intuitiv nachvollziehbar, welches der einzelnen Modelle hinsichtlich eines Aspektes des Inferenzergebnisses in welcher Form beigetragen hat. In der Benutzermodellierung ist es wünschenswert, ein einziges Bayes’sches Netz zu nutzen und zu verwalten, das zur Begründung der Adaptionsentscheidungen des benutzeradaptiven Systems herangezogen werden kann.

In diesem Zusammenhang wird im Folgenden ein existierendes Verfahren vorgestellt und im Kontext benutzeradaptiver Systeme angewendet, das Meta-Wissen aus empirischen Daten oder—alternativ—einer gegebenen Menge an Bayes’schen Netzen extrahiert. Es kann im Rahmen eines Wissensentdeckungsprozesses zur Identifikation interessanter struktureller Eigenschaften der modellierten Domäne eingesetzt werden. Die mit dem Verfahren ermittelten Informationen können wiederum im weiteren Konstruktionsprozess berücksichtigt werden. Dieses Meta-Wissen kann insbesondere zur Identifikation und adäquaten Modellierung individueller Unterschiede in mit Bayes’schen Netzen kodierten Benutzermodellen dienen. In Abschnitt 7.3 wird auf dieser Basis eine neue, in dieser Arbeit entwickelte Methode zur Adaption der Struktur eines Bayes’schen Netzes vorgestellt und diskutiert.

7.2.2 Meta-Netze

Die im Rahmen des maschinellen Lernens der Struktur Bayes'scher Netze eingesetzten *Meta-Netze* wurden in der im Folgenden verwendeten Form in Kapitel 5 der Dissertation von Reimar Hofmann (2000) eingeführt. Mit ihnen kann die strukturelle Unsicherheit, die beim Lernen mit wenigen Trainingsdaten vorliegt, erfasst und in kompakter Form repräsentiert werden (siehe z.B. auch Friedman & Koller, 2002). In einer solchen Lernsituation ist die Unsicherheit darüber, welche der vielen potenziell möglichen Strukturen die „richtige" ist—wie bereits diskutiert wurde—sehr hoch. Viele der infrage kommenden Strukturen besitzen bei einem Bayes'schen Ansatz des Strukturlernens typischerweise eine vergleichbare A-posteriori-Wahrscheinlichkeit. In den seltensten Fällen tritt die Situation ein, dass eine einzelne Struktur mit einer Wahrscheinlichkeit nahe Eins identifiziert werden kann (Hofmann, 2000; Friedman & Koller, 2002).

Beim Bayes'schen Lernen kann die A-posteriori-Wahrscheinlichkeit der erlernten Strukturen als Qualitäts- bzw. Unsicherheitsmaß verwendet werden. Damit können allerdings nur Strukturen als Ganzes verglichen werden; Effekte, die lediglich auf der Unsicherheit des (Nicht-)Vorhandenseins einer einzigen Kante beruhen, können hiermit beispielsweise nicht identifiziert werden.

In dem mit den Meta-Netzen verfolgten Ansatz wird die strukturelle Unsicherheit auf der Kantenebene betrachtet und ein probabilistisches Modell der Zusammenhänge zwischen dem Fehlen bzw. der Existenz der potenziellen Kanten auf der Basis ihrer A-posteriori-Wahrscheinlichkeiten zur Verfügung gestellt.

Ein *Meta-Knoten* X^M_{vw} eines (Bayes'schen) Meta-Netzes $B^M = (G^M, \boldsymbol{\theta}^M)$ repräsentiert eine potenzielle Kante zwischen zwei Variablen X_v und X_w eines Bayes'schen Netzes $B = (G, \boldsymbol{\theta})$, das zur Modellierung einer Domäne genutzt wird.[2] Jeder dieser Meta-Knoten besitzt drei Zustände $x^M_{vw_1}, x^M_{vw_2}$ und $x^M_{vw_3}$, die folgende Hypothesen abbilden: (i) das Nichtvorhandensein einer Kante, (ii) das Vorhandensein einer Kante von X_v zu X_w, und (iii) das Vorhandensein einer Kante von X_w zu X_v. Hinsichtlich komplexer Netze ist es aufgrund der hohen Anzahl der möglichen Strukturen möglich, dass nicht für jedes der Variablenpaare ein Meta-Knoten verwaltet werden kann. Im nächsten Abschnitt wird gezeigt, wie in heuristischer Weise eine adäquate Auswahl der sinnvollerweise zu verwendenden Meta-Knoten getroffen werden kann.

Meta-Kanten sind Kanten zwischen Meta-Knoten von B^M, die direkte Abhängigkeiten zwischen den Kanten von B repräsentieren. Beispielsweise können durch das Einbringen einer neuen Kante in B eine oder mehrere andere Kanten überflüssig werden, die dann aus G entfernt werden können.

Abbildung 7.11 zeigt ein Beispiel eines Bayes'schen Netzes mit dem zugehörigen Meta-Netz. Für jede der im Netz vorhandenen Kanten existiert ein korrespondierender Meta-Knoten. Zusätzlich besitzt das Meta-Netz einen Meta-Knoten $C \rightarrow B$, der zu einer potenziellen Kante zwischen C und B gehört, die in der aktuellen Struktur des Bayes'schen Netzes nicht auftritt. Das Meta-Netz besitzt eine Meta-Kante zwischen den beiden Meta-Knoten $B \rightarrow D$ und $C \rightarrow D$. Mit ihr kann etwa der Sachverhalt modelliert werden, dass die Existenz der Kante von B nach D im Bayes'schen Netz voraussetzt, dass die Kante von C nach D ebenfalls existiert. Analog könnte die Situation repräsentiert werden, dass das Vorhandensein einer der beiden Kanten die Existenz der jeweiligen anderen verbietet. Typischerweise werden anstelle der in diesem Beispiel diskutierten deterministischen Zusammenhänge zwischen potenziellen Kanten probabilistische, d.h., (bedingte) Wahrscheinlichkeiten, betrachtet. Die Existenz einer Kante könnte z.B. dazu führen, dass eine weitere mit einer Wahrscheinlichkeit von 0.8 ebenfalls existiert bzw. nicht existiert.

[2] Dabei sind die Knoten $\boldsymbol{X}$ von Beginn an bekannt. Die verbleibenden Komponenten von B (Kanten, bedingte Wahrscheinlichkeiten) werden im weiteren Verlauf des Verfahrens erlernt.

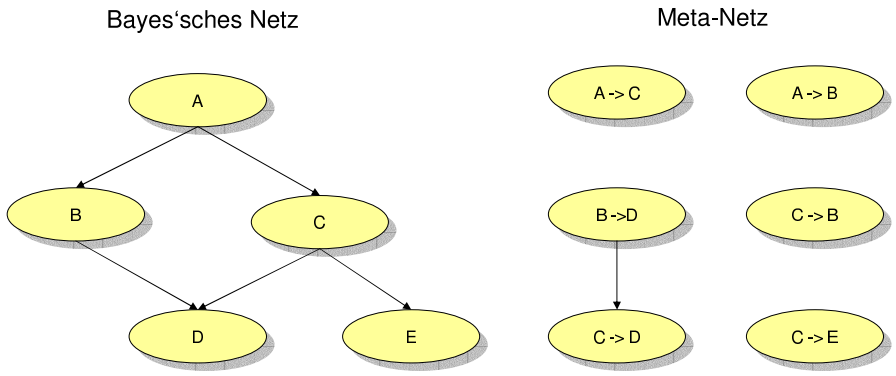


***Abbildung 7.11**: Beispiel eines Meta-Netzes*

Ein Meta-Netz B^M ist in der Lage, eine gemeinsame Wahrscheinlichkeitsverteilung über dem Raum der Kanten bzw. möglichen Strukturen zu modellieren. Im Folgenden wird beschrieben, wie ein solches Meta-Netz anhand empirischer Daten ermittelt werden kann.

7.2.3 Lernen der Meta-Netze

Hofmann (2000) betrachtet Meta-Netze im Zusammenhang mit erschöpfender Suche beim Strukturlernen, d.h., er untersucht sehr einfache Netze mit fünf bzw. zwölf Variablen, die es ermöglichen, sämtliche Strukturen des Lösungsraums zu bewerten. Weiterhin betrachtet er nur den Fall voll beobachteter Daten. Nachfolgend wird eine in dieser Arbeit entwickelte Erweiterung der Methode vorgestellt, die Hofmann's Ansatz zum Erlernen der Meta-Netze in der allgemeinen Situation des Lernens mit komplexeren Netzstrukturen sowie fehlender Daten ermöglicht.

Zum Erlernen eines Meta-Netzes $B^M = (G^M, \boldsymbol{\theta}^M)$ eines zu bestimmenden Bayes'schen Netzes $B = (G, \boldsymbol{\theta})$ müssen drei Teillernaufgaben gelöst werden:

1. die Entscheidung, welche Meta-Knoten betrachtet werden sollen, d.h., welche potenziellen Kanten in B vorhanden sein können;
2. das Erlernen eines DAG G^M, um die (Un-)Abhängigkeiten zwischen den Kanten in G zu modellieren; und
3. das Erlernen der CPTs $\boldsymbol{\theta}^M$ von B^M.

Diese Aufgabenstellungen werden in zwei Schritten gelöst, wobei die letzten beiden Lernaufgaben gemeinsam behandelt werden.

In der initialen Phase wird eine Menge $\boldsymbol{G} = \{G_1, \ldots, G_m\}$ von m Netzstrukturen ermittelt, die als Stichprobe der infrage kommenden Netzstrukturen dient. Eine vollständige Aufzählung bzw. Analyse aller Strukturen ist wegen der Super-Exponentialität in der Anzahl der Variablen in den interessanten Fällen nicht möglich. Es existieren mehrere Verfahren eine solche Stichprobe—mehr oder minder hoher Qualität—zu erzeugen: Die einfachste Variante, die u.a. von Madigan und Raftery (1994) und Madigan und York (1995) vorgeschlagen und verwendet wird, besteht in der Approximation durch m hochbewertete Strukturen G_i. Dies ist möglich, da sich die wahrscheinlichen Strukturen typischerweise durch mehrere Zehnerpotenzen in ihren A-posteriori-Wahrscheinlichkeiten unterscheiden, und es somit im Normalfall genügt, die mit einer nicht vernachlässigbaren Wahrscheinlichkeit bewerteten Strukturen zu betrachten. Eine einfache Möglichkeit, solche

Strukturen zu finden, ist das Aufzeichnen der entsprechenden Strukturen während des Suchvorgangs beim Strukturlernen. Nachteil ist hierbei, dass die Auswahl der Strukturen stark vom eingesetzen Suchmechanismus abhängig ist und eventuell schlechte, da sehr ähnliche, im Suchraum benachbarte Resultate liefern kann. Alternative, teilweise deutlich aufwendigere Selektionsprozeduren, wurden von Madigan und Raftery (1994) und Madigan und York (1995) entwickelt, die darauf zielen, repräsentativere Stichproben zu erzeugen. Eine weitere Diskussion des Problems und ein Lösungsvorschlag findet sich bei Friedman, Goldszmidt und Wyner (1999).

Mit der Menge $\boldsymbol{G}$ an Strukturen können zur Behandlung von Punkt 1. des Meta-Lernvorgangs die Meta-Knoten wie folgt festgelegt werden: Geht man von der Annahme aus, dass eine Kante, die eine gewisse Rolle in der betrachteten Domäne spielt, zumindest in einer der Strukturen in $\boldsymbol{G}$ auftritt, sollte zu jeder Kante, die in wenigstens einer der m Strukturen G_i vorhanden ist, der entsprechende Meta-Knoten in das Meta-Netz aufgenommen werden. In den meisten realistischen Szenarien ist diese Annahme plausibel. Die Wahrscheinlichkeit, dass alle interessanten Kanten in $\boldsymbol{G}$ auftreten, kann durch eine Erhöhung der Anzahl m der betrachteten Strukturen vergrößert werden, was in vielen Fällen keinen wesentlichen zusätzlichen Rechenaufwand darstellt. Es muss gegebenenfalls lediglich eine größere Anzahl an Strukturen während des Lernvorgangs gespeichert werden.

An diesem Punkt in der Prozedur sind die Meta-Knoten des Meta-Netzes B^M festgelegt. Es verbleiben die Punkte 2. und 3. des Meta-Lernens: das Erlernen der Struktur G^M—der Kanten des Meta-Netzes—und der zugehörigen bedingten Wahrscheinlichkeiten der CPTs $\boldsymbol{\theta}^M$. Der in der vorliegenden Arbeit verfolgte Lösungsansatz basiert im Gegensatz zu demjenigen von Hofmann, der eine erschöpfende Suche durchführt, auf der Verwendung der Menge $\boldsymbol{G}$ der potenziellen Strukturen als Trainingsdaten eines Meta-Strukturlernvorgangs: Jede der Strukturen G_i kann als ein Vektor aufgefasst werden, der Informationen zum Fehlen bzw. Vorhandensein der betrachteten (gerichteten) Kanten kodiert. Ein solcher Vektor—und damit die zugehörige Struktur—stellt einen Trainingsfall für den Meta-Strukturlernvorgang dar. Jeder dieser *Meta-Trainingsfälle* wird mit der A-posteriori-Wahrscheinlichkeit der korrespondierenden Struktur $P(G_i \mid \boldsymbol{D})$ gewichtet. Für den Meta-Strukturlernvorgang schlägt Hofmann (2000) anstelle des Einsatzes des BIC als Bewertungsfunktion die Verwendung eines Bayes'schen Qualitätsmaßes aus Gleichung 4.21 vor, um sinnvolle Mengen von Meta-Kanten zu erhalten. Das BIC ist hierfür nicht geeignet, da es im Rahmen seines Genauigkeit-Komplexitäts-Tradeoffs zu wenige Meta-Kanten produziert. Der Anteil des BIC, der die Komplexität, d.h., die Anzahl der Kanten einer Struktur, „bestraft", ist für einen Einsatz im Meta-Strukturlernen relativ zu stark gegenüber der Bewertung der Qualität gewichtet.

Der Meta-Strukturlernvorgang basiert auf der Berechnung der A-posteriori-Wahrscheinlichkeit $P(L \mid \boldsymbol{D})$ des Vorhandenseins einer Kante L gemäß:

$$P(L \mid \boldsymbol{D}) = \sum_{G} P(G \mid \boldsymbol{D}) L(G), \tag{7.1}$$

mit $L(G) = 1$ wenn L in G vorhanden ist und $L = 0$ andernfalls (vgl. beispielsweise Friedman & Koller, 2002), d.h., die A-posteriori-Wahrscheinlichkeit ergibt sich als die Summe der A-posteriori-Wahrscheinlichkeiten derjenigen Strukturen, die die betrachtete Kante besitzen.

Die benötigten A-posteriori-Wahrscheinlichkeiten $P(G \mid \boldsymbol{D})$ können wie in Abschnitt 4.4.2 beschrieben ermittelt werden. Im Fall vollständiger Trainingsdaten $\boldsymbol{D}$ kann der Wert anhand Formel 4.19 in geschlossener Form bis auf den konstanten Faktor $P(\boldsymbol{D})^{-1}$ bestimmt werden, welcher aber für die Optimierungsaufgabe keine Rolle spielt. Im Rahmen der Interpretation der Lernergebnisse will man aber oft den absoluten Wert betrachten, der aufgrund der notwendigen Norma-

lisierung über die exponentielle Anzahl möglicher Strukturen nicht mehr geschlossen berechnet werden kann. In solchen Fällen, kann als Approximation das Prinzip der „relativen Masse“ auf Basis der Stichprobe $\boldsymbol{G}$ der Strukturen unter Verwendung des Satzes von Bayes angewendet werden (vgl. z.B. Murphy, 2001):

$$P(G \mid \boldsymbol{D}) \approx \frac{P(\boldsymbol{D} \mid G)P(G)}{\sum_{G_i \in \boldsymbol{G}} P(\boldsymbol{D} \mid G_i)P(G_i)}. \tag{7.2}$$

Man erhält mit dieser Methode Schätzwerte der A-priori-Wahrscheinlichkeiten, die auf das Intervall $[0, 1]$ normiert sind. Die Normierung auf der Basis einer Stichprobe führt zu einem Überschätzen der tatsächlichen Wahrscheinlichkeitswerte.

Eine rechenintensivere und genauere Alternative zur Approximation der A-posteriori-Wahrscheinlichkeit stellen *Markov-Ketten-Monte-Carlo*-Methoden (engl. *Markov-Chain-Monte-Carlo, MCMC*) dar (siehe z.B. Friedman & Koller, 2002). Diese Methoden können ebenfalls im Fall unvollständiger Trainingsdaten $\boldsymbol{D}$—neben der Approximation mit dem BIC, wie sie in Abschnitt 4.4.2 beschrieben wurde—verwendet werden. Aufgrund der hohen Komplexität eignen sie sich nicht zum Einsatz zur Laufzeit eines (benutzeradaptiven) Systems und kommen deshalb in dieser Arbeit nicht zur Anwendung.

Es ist denkbar, dass Meta-Netze auch in Situationen erlernt werden, in denen keine empirischen Daten vorhanden sind, sondern statt dessen eine Sammlung unabhängig voneinander konstruierter Bayes’scher Netze zur Lösung des gleichen oder zumindest ähnlicher Probleme vorhanden ist. Ein solches Szenario wird beispielsweise von Borth (2002) anhand des Beispiels auf Bayes’schen Netzen basierender Expertensysteme im Entwicklungsprozess technischer Systeme bei DAIMLERCHRYSLER beschrieben. Es werden dort in verschiedenen Einsatzsituationen (z.B. bei der Konstruktion unterschiedlicher Fahrzeug-Modellreihen) Netze zur Lösung der gleichen Aufgaben konstruiert, die für das Meta-Lernen zusammengeführt werden können und in ihrer Gesamtheit als Menge $\boldsymbol{G}$ im beschrieben Meta-Strukturlernprozess dienen würden.

7.2.4 Beispiel: Flughafenexperiment

Anhand des kombinierten Datensatzes der beiden Varianten des Flughafenexperiments—d.h. sowohl ohne als auch mit Lautsprecherdurchsagen (Abschnitt 2.2.2.5)—soll an dieser Stelle das Konzept der Meta-Netze veranschaulicht werden. Um das Meta-Modell einer größeren Domäne untersuchen zu können, wurde eine höhere Anzahl an Symptomvariablen in die folgende Studie aufgenommen, die in Abbildung 7.12 aufgeführt sind. Die abgebildete Struktur diente gleichzeitig als Ausgangspunkt des Strukturlernproblems. Während des im vorhergehenden Abschnitt beschriebenen Meta-Lernvorgangs wurden die 60 höchstbewerteten Strukturen ($m = 60$) betrachtet und die Bayes’sche Metrik mit einer gemäß Gleichung 4.21 vorgegebenen A-priori-Verteilung über den potenziellen Strukturen verwendet ($\kappa = 0.9, \delta = \#\text{Kanten}$).

Die gemäß Gleichung 7.2 approximierten A-posteriori-Wahrscheinlichkeiten der 60 bestbewerteten Strukturen reichten von $< 10^{-6}$ bis 0.103473. Es wurden 25 potenzielle Kanten—und damit auch die zugehörigen Meta-Knoten—identifiziert, wovon 20 in der anhand des Meta-Netzes ermittelbaren (siehe nächster Abschnitt) in Abbildung 7.13 dargestellten wahrscheinlichsten Struktur[3] auftreten. Auch in dieser Studie wird der bereits bekannte Effekt der Isolation der Va-

[3] Berücksichtigt man die Likelihood-Äquivalenz von Strukturen (vgl. Abschnitt 4.4.2), so existiert eine Äquivalenzklasse der wahrscheinlichsten Strukturen wovon die in Abbildung 7.13 angeführte einen Repräsentanten darstellt.

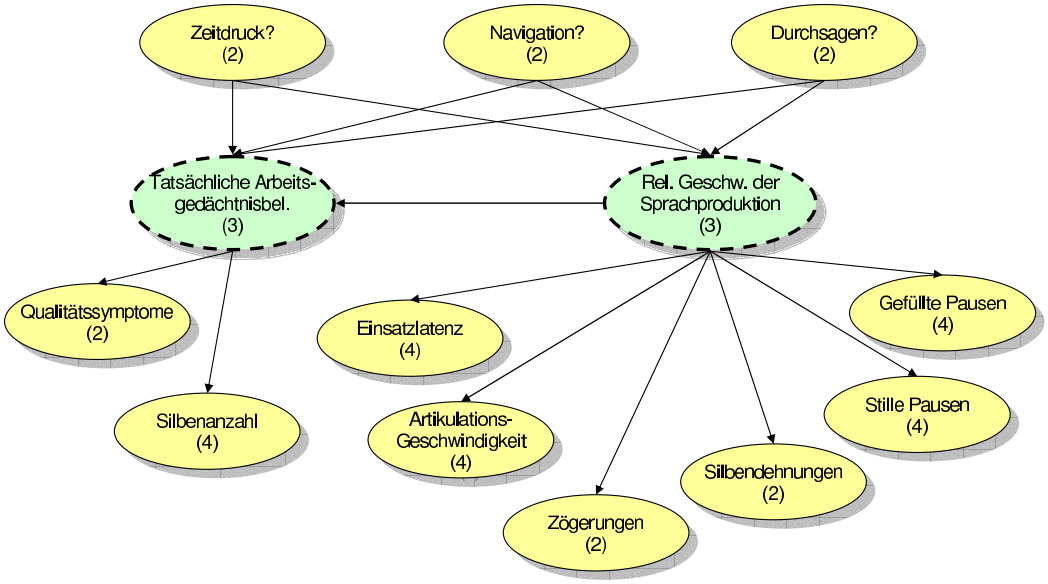


***Abbildung 7.12**: Ausgangsnetz des Meta-Lernprozesses*

riablen NAVIGATION? beobachtet: Das Lernverfahren kann lediglich einen Zusammenhang zwischen dem Sprachsymptom ARTIKULATIONSGESCHWINDIGKEIT und der im Experimentaldesign unabhängigen Variablen feststellen. Es können keine weiteren direkten Zusammenhänge mit anderen Variablen vom Lernverfahren ermittelt werden. Weiterhin werden—durchaus plausible—starke Beziehungen zwischen den beiden Pausenvariablen sowie den Variablen ZEITDRUCK? und EINSATZLATENZ bzw. QUALITÄTSSYMPTOME erkannt. Die Tatsache, dass die im Meta-Netz kodierte Information andererseits a priori das Vorhandensein der direkten Kante zwischen DURCHSAGEN? und GEFÜLLTE PAUSEN repräsentiert, kann zum Teil darauf zurückzuführen sein, dass die Versuchspersonen versuchen, die Lautsprecherdurchsagen mit gefüllten Pausen zu maskieren. Eine weitere Erklärungsmöglichkeit beruht auf den Kodierungsunterschieden der Teildatenmengen der beiden Varianten des Flughafenexperimentes (vgl. Abschnitt 2.2.2.5). Ähnliche Effekte werden auch bei den Symptomen der Zögerungen und Silbendehnungen beobachtet. Um endgültig zu entscheiden, worauf die Beobachtungen beruhen, könnten weitere Studien mit Versuchspersonen durchgeführt werden, die sowohl mit als auch ohne Lautsprecherdurchsagen die Experimentalaufgabe bearbeiten müssen.

Das erlernte Meta-Netz selbst besteht aus 25 Meta-Knoten mit einer einzigen Meta-Kante. Die Meta-Kante repräsentiert einen direkten Zusammenhang zwischen dem (Nicht-)Vorhandensein der Kante zwischen DURCHSAGEN? und GEFÜLLTE PAUSEN und dem (Nicht-)Vorhandensein der Kante zwischen GEFÜLLTE PAUSEN und RELATIVE GESCHWINDIGKEIT DER SPRACHPRODUKTION. Eine Analyse der zugehörigen Meta-CPT liefert die Erklärung, dass diese Meta-Kante der Sicherung der Konsistenz der Richtungen der beiden potenziellen Kanten in B dient. Die wahrscheinlichste Kombination besteht erwartungsgemäß darin, dass beide Kanten in Richtung der Symptomvariablen verlaufen. Wird die Richtung der Kante zwischen DURCHSAGEN? und GEFÜLLTE PAUSEN umgekehrt, so würde dies zu einem Zyklus in der Struktur führen. Die Meta-CPT bewirkt in dieser Situation, dass die Wahrscheinlichkeit einer Umkehrung der Richtung der zweiten Kante sinnvollerweise auf Eins ansteigt. Obwohl analoge Situationen auch im Zusammen-

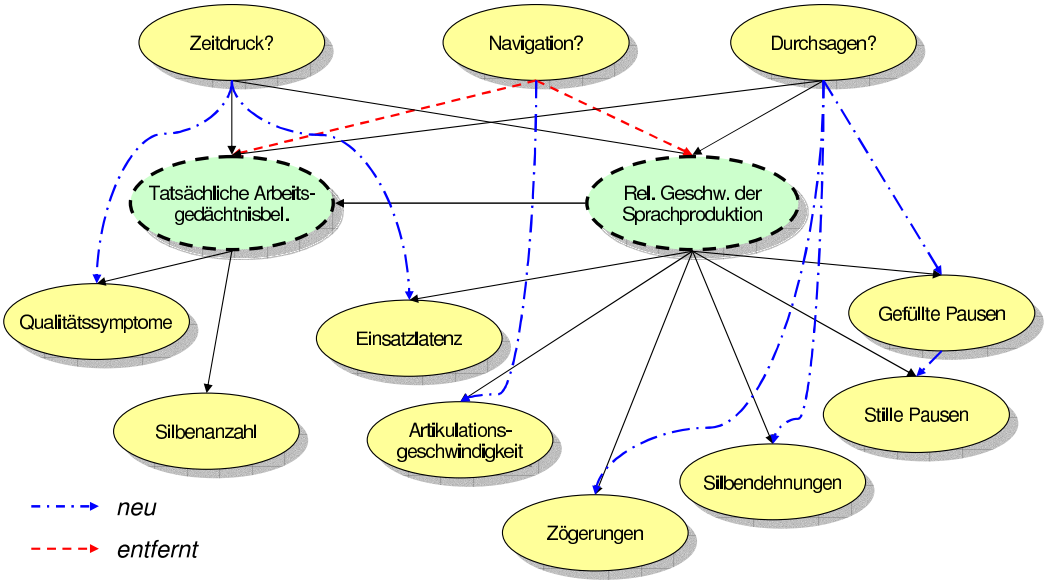


***Abbildung 7.13**: Wahrscheinlichste Struktur nach dem Meta-Lernprozesses*

hang mit anderen Symptomvariablen denkbar sind, ist GEFÜLLTE PAUSEN die einzige Variable bei der im Verlauf des Strukturlernvorgangs eine andere Kantenrichtung in den Zwischenergebnissen mehrfach beobachtet und somit in Form einer Meta-Information kodiert wurde. Hinsichtlich aller anderen Symptomvariablen wurde nur die Richtung von der unabhängigen zur abhängigen Variable des Experiments beobachtet, weshalb keine Unsicherheit über die Kantenrichtung modelliert wird.

7.3 Strukturelle Adaption mit Meta-Netzen

Nachdem bislang das Erlernen der Struktur im Vordergrund der Diskussion dieses Kapitels stand, wird im verbleibenden Teil die Adaption der Struktur Bayes'scher Netze thematisiert. Es wird ein neues Verfahren vorgestellt, das in einer empirischen Analyse mit alternativ einsetzbaren Ansätzen verglichen wird—ähnlich wie in dem in Kapitel 6 beschriebenen Fall der Adaption der bedingten Wahrscheinlichkeiten.

7.3.1 Motivation: Dynamische Domänen, inter-individuelle Unterschiede

Die strukturelle Adaption (vgl. Abschnitt 4.5.2), d.h., das Hinzufügen und/oder Löschen von Kanten der Struktur des Bayes'schen Netzes, sowie das damit in zwei Schritten ebenso zu realisierende Umkehren der Richtung einer Kante, kann wie die Adaption der CPTs dazu genutzt werden, individuelle Unterschiede zwischen den Benutzern zu erkennen und im Modell zu berücksichtigen.

Im Gegensatz zur Adaption der bedingten Wahrscheinlichkeiten, die in der Lage ist, in vergleichsweise kurzer Zeit, d.h., auf der Basis einer geringen Anzahl von Adaptionsfällen, adäquate Modifikationen vorzunehmen, können strukturelle Adaptionsmechanismen längerfristige, schwerwiegendere Anpassungen der direkten Zusammenhänge zwischen den Variablen in den Modellen

vornehmen. Potenziell notwendige Kantenmodifikationen können nicht anhand einzelner Adaptionsfälle detektiert werden. Um Abweichungen zwischen den direkten Beziehungen der Variablen des Modells und der Realität aufzudecken, ist im Normalfall eine größere Menge an Adaptionsdaten notwendig. Deshalb können die entsprechenden Verfahren nutzbringend in solchen Szenarien eingesetzt werden, die sich durch längerfristige dynamische Veränderungen auszeichnen bzw. solche Veränderungen erwarten lassen.

Zusätzlich zur Frage, wie das Modell strukturell modifiziert werden kann, muss entschieden werden, auf welcher Menge von Adaptionsfällen die strukturellen Entscheidungen getroffen werden sollen. Eine Untersuchung diesbezüglich alternativ ausgerichteter Methoden steht im Mittelpunkt der vorgestellten empirischen Studie. Ein in dieser Hinsicht flexibles, neu entwickeltes Verfahren, das auf den in den vorangehenden Abschnitten betrachteten Meta-Netzen basiert, wird im Folgenden ausführlich beschrieben.

Ein technischer Aspekt, den es bei der Adaption der Struktur zu beachten gilt, besteht darin, dass hinsichtlich der numerischen Genauigkeit der Modellierung lediglich entscheidend ist, welche fehlenden Kanten noch in das Modell aufgenommen werden müssen, um die entsprechenden direkten Zusammenhänge repräsentieren zu können. Kanten, die nicht zur Modellierung der Einflüsse zwischen den Variablen benötigt werden, stellen im Zusammenhang mit der Vorhersagegenauigkeit der Netze üblicherweise kein Problem dar. Sie führen lediglich zu komplexeren Modellen und damit in manchen Fällen zu Overfitting. Unter dem Gesichtspunkt der Interpretierbarkeit der Modelle, sollte der Adaptionsmechanismus dennoch jederzeit solche zur adäquaten Modellierung überflüssige Kanten entdecken und aus der Struktur entfernen.

7.3.2 Überblick über das Verfahren

Bevor im anschließenden Abschnitt die Details des Adaptionsvorgangs beschrieben werden, wird ein Überblick des in dieser Arbeit neu entwickelten Verfahrens der *strukturellen Adaption Bayes'scher Netze mit Meta-Netzen* gegeben.

Die zugrunde liegende Idee des verfolgten Ansatzes besteht im Einsatz eines Meta-Netzes B^M zur Kodierung der strukturellen Information über die betrachtete Domäne. Insbesondere werden in dieser Weise Informationen zur Existenz bzw. zum Fehlen von Kanten in der Struktur des zur eigentlichen Modellierung verwendeten Bayes'schen Netzes B repräsentiert. Bei Zugriff auf neue Adaptionsfälle wird das Meta-Netz genutzt, um strukturelle Veränderungen der Domäne zu inferieren, d.h., neu aufzunehmende, umzukehrende oder zu entfernende Kanten desjenigen Netzes zu erkennen, das in der Performanzkomponente des Systems zum Einsatz kommt. Kurz gesagt besteht die vorgestellte Methode in einer Anwendung von Standard-CPT-Adaptionsmethoden wie beispielsweise AHUGIN auf der Meta-Ebene, d.h., in der Adaption der (bedingten) Wahrscheinlichkeiten des Meta-Netzes.

Abbildung 7.14 beinhaltet den Grundaufbau der strukturellen Adaption mit Meta-Netzen. Als manuell zu spezifizierender Parameter muss eine globale ESS s vorgegeben werden, die die Adaptionsrate angibt und die wie üblich die Einschätzung des Systementwicklers repräsentiert, inwieweit die Trainingsdaten $\boldsymbol{D}$ in der Lage sind, den aktuellen Einsatzkontext widerzuspiegeln.

Der erste Schritt des Verfahrens besteht in der Konstruktion eines Meta-Netzes B^M anhand der verfügbaren Trainingsdaten. Auf der Basis dieses Meta-Netzes wird dann ein initiales Bayes'sches Netz B zum Einsatz in der Performanzkomponente des Systems ermittelt. Nach einem „Fenster" von k neuen Adaptionsfällen $\boldsymbol{D}^{adapt}$ (die gleichzeitig zur Adaption der CPTs verwendet werden können) wird diese Menge genutzt, um einen Adaptionsschritt der CPTs $\boldsymbol{\theta}^M$ des Meta-Netzes B^M

STRUKTURELLE ADAPTION MIT META-NETZEN($\boldsymbol{D}, s$)
$B^M \leftarrow$ *lerne_Meta-Netz*$(\boldsymbol{D}, s)$
$B \leftarrow$ *bestimme_Bayes'sches_Netz*(B^M)
while ¬**exit** **do**
  $\boldsymbol{D}^{adapt} \leftarrow \emptyset$
  for $i = 1$ **to** k **do**
    case $\leftarrow$ *nächster_Adaptionsfall*$()$
    $\boldsymbol{D}^{adapt} \leftarrow \boldsymbol{D}^{adapt} \cup$ *case*
    $B \leftarrow$ *adaptiere_CPTs*$(B,$ *case*$)$
  $B^M \leftarrow$ *adaptiere_CPTs_des_Meta-Netzes*$(\boldsymbol{D}^{adapt})$
  $B \leftarrow$ *bestimme_Bayes'sches_Netz*(B^M)

***Abbildung 7.14**: Strukturelle Adaption mit Meta-Netzen*

durchzuführen. Die aktualisierte Meta-Information kann anschließend gegebenenfalls ein strukturell verändertes Bayes'sches Netz B liefern.

Aufgrund seiner Arbeitsweise, die im Wesentlichen durch die Einteilung des Adaptionsprozesses in Fenster bestehend aus k Adaptionsfällen charakterisiert ist, bietet sich das Verfahren in Situationen an, in denen ein Verwalten einer (zu) großen Datenmenge nicht erwünscht oder praktikabel ist. Bei der strukturellen Adaption mit Meta-Netzen genügt es, die in Form des Meta-Netzes kodierte Information zur strukturellen Unsicherheit und die aktuellen Adaptionsdaten vorzuhalten, um ein an den aktuellen Kontext adaptiertes Modell ermitteln zu können. Dies kann je nach vom System zu erfüllenden Randbedingungen entweder zur Laufzeit oder semi-offline, d.h., beispielsweise zwischen zwei Benutzersitzungen unter Ausnutzung freier Rechenkapazitäten, geschehen.

7.3.3 Adaptionsprozedur

In Abschnitt 7.2.3 wurde bereits beschrieben, wie die Meta-Netze erlernt werden können. Um die strukturelle Adaption mit Meta-Netzen zu vervollständigen, verbleiben noch (i) die Beschreibung des Vorgehens zur Bestimmung des aktuellen Bayes'schen Netzes B sowie (ii) die Darstellung des Adaptionsprozesses von B unter Verwendung des Meta-Netzes B^M. Es wird mit Letzterem begonnen.

Wie bereits angedeutet wurde, wird ein Standardverfahren zur Adaption der CPTs $\boldsymbol{\theta}^M$ wie das AHUGIN-Verfahren auf der Meta-Ebene eingesetzt. Um dies zu ermöglichen, müssen die k Adaptionsfälle $\boldsymbol{D}^{adapt}$ des letzten Beobachtungsfensters derart transformiert werden, dass sie für eine Verwendung im Zusammenhang mit dem Meta-Netz geeignet sind. Eine mögliche Lösung besteht darin, eine Menge bestehend aus den m Strukturen anhand von $\boldsymbol{D}^{adapt}$ zu erlernen—analog zum Vorgehen beim Erlernen der Meta-Netze. Gemeinsam dienen diese Netze im Rahmen der nachfolgend beschriebenen Methode als ein Adaptionsfall für den Adaptionsprozess des Meta-Netzes. Anhand dieses Adaptionsfalls werden die bedingten Wahrscheinlichkeiten $\boldsymbol{\theta}^M$ des Meta-Netzes gemäß der gewählten CPT-Adaptionsmethode angepasst.

Die A-posteriori-Wahrscheinlichkeit jedes Meta-Zustandes $x^M_{vw_j}$ bzw. der zugehörigen Kante in B kann mit Gleichung 7.1 berechnet werden:

$$P(x^M_{vw_j} \mid \boldsymbol{D}^{adapt}) = \sum_{i=1}^{m} P(G_i \mid \boldsymbol{D}^{adapt}) \cdot k(G_i, x^M_{vw_j}), \tag{7.3}$$

wobei $k(G_i, x^M_{vw_j})$ die Indikatorfunktion

$$k(G_i, x^M_{vw_j}) := \begin{cases} 0 & \text{, wenn } G_i \text{ nicht konsistent ist mit } x^M_{vw_j} \\ 1 & \text{, wenn } G_i \text{ konsistent ist mit } x^M_{vw_j} \end{cases} \tag{7.4}$$

bezeichnet. Eine Struktur G_i wird als *konsistent* mit $x^M_{vw_j}$ bezeichnet, wenn die Existenz oder das Fehlen der Kante, die bzw. das durch den Meta-Zustand kodiert wird, in der Tat in G_i so auftritt, d.h., modelliert ein Meta-Zustand das Fehlen einer Kante zwischen zwei Variablen X_v und X_w, dann sind alle Strukturen, die diese Kante nicht aufweisen, konsistent mit diesem Meta-Zustand.

Die in dieser Weise ermittelten A-posteriori-Wahrscheinlichkeiten können als (Likelihood-) Evidenzen[4] (siehe z.B. Jensen, 2001) für die Meta-Knoten dienen, um einen Adaptionsprozess der Meta-CPTs anzustoßen. Danach wird die *wahrscheinlichste Hypothese* (engl. *most probable hypothesis*, siehe ebenfalls Jensen, 2001) des Meta-Netzes berechnet, d.h., die Zustandskombination der Meta-Knoten, die a posteriori am wahrscheinlichsten ist. Das Resultat ist ein Vektor von Meta-Zuständen und damit gleichzeitig ein Vektor von Kanten, die in ihrer Gesamtheit die wahrscheinlichste Struktur für B definieren. Um die Eigenschaft der Zyklenfreiheit zu gewährleisten, muss diese bei der Bestimmung der wahrscheinlichsten Hypothese berücksichtigt werden, d.h., das endgültige Ergebnis ergibt sich als die wahrscheinlichste Hypothese, die ein zyklenfreies Netz repräsentiert. Diese wahrscheinlichste, zyklenfreie Struktur G' wird als neue—möglicherweise adaptierte—Struktur verwendet, die das von der Performanzkomponente des Systems eingesetzte Bayes'schen Netz B besitzt.

In einem letzten Schritt müssen im Fall einer vorgenommenen Strukturmodifikation die zugehörigen CPTs $\boldsymbol{\theta}'$ (neu) berechnet werden. Im Normalfall bleiben große Teile der CPTs $\boldsymbol{\theta}'$ nach einem strukturellen Adaptionsschritt unverändert. Nur diejenigen $\boldsymbol{\theta}'_i$ müssen neu ermittelt werden, die im direkten Zusammenhang mit einer strukturellen Veränderung stehen, d.h., dort wo neue CPT-Einträge entstanden oder weggefallen sind. Diese Werte können als $P(x_{ij} \mid pa_k^{old}(X_i))$ mit den Standardinferenzverfahren unter Verwendung des Bayes'schen Netzes B^{old}—das Netz bevor der Adaptionsschritt durchgeführt wurde—bestimmt werden. Ebenso werden die neuen ESS-Werte s'_{ik} dieser veränderten Teile der CPTs benötigt. In vielen praktisch relevanten Fällen ist es möglich, die potenziell im Verlauf des Adaptionsvorgangs benötigten ESS-Werte unter Verwendung einer entsprechenden Datenstruktur zu verwalten (Moore & Lee, 1998). Ist dies nicht der Fall, so muss auf Heuristiken zurückgegriffen werden. Eine Möglichkeit besteht in der Verwendung der entsprechenden ESS-Werte des Meta-Netzes, d.h., dem ESS-Wert, der Teil der Modellierung des (Nicht-)Vorhandenseins der betrachteten Kante ist. Diese Vorgehensweise, die auch in den folgenden empirischen Studien angewendet wurde, ist durch die Interpretation der ESS als Konfidenzmaß der Modellierung motiviert: Das Vertrauen in die Wahrscheinlichkeit des (Nicht-)Vorhandenseins einer Kante (wie im Meta-Netz kodiert) wird häufig in engem Zusammenhang mit dem Vertrauen in die CPT-Werte des eingesetzen Netzes stehen. Diese Heuristik tendiert (fälschlicherweise) zu einer zu geringen Einschätzung des ESS-Wertes, da ein Adaptionsfall auf der Metaebene k echte Adaptionsfälle aggregiert. Eine Alternative bildet deshalb die Multiplikation der ESS des Meta-Netzes mit k.

[4] Eine *Likelihood-Evidenz* ist ein Vektor von Wahrscheinlichkeiten, die den Zuständen der betrachteten Zufallsvariable zugeordnet werden. Sie repräsentiert Aussagen wie z.B. „Zustand 1 ist mit einer Wahrscheinlichkeit von 0.4 eingetreten, während Zustand 2 mit 0.6 eingetreten ist.". „Normale" Evidenzen können als Likelihood-Evidenzen interpretiert werden, die aus einer einzigen Eins und weiteren Nullen bestehen: Es wurde genau ein Zustand beobachtet. Alle anderen sind damit ausgeschlossen.

7.3.4 Diskussion

Die vorgestellte Methode der strukturellen Adaption mit Meta-Netzen besitzt hinsichtlich einiger Komponenten generischen Charakter. Es ist möglich, den Strukturlernalgorithmus inklusive der eingesetzten Bewertungsfunktion, die Methode zur Approximation der A-posteriori-Wahrscheinlichkeiten bzw. die Auswahl der Strukturstichprobe sowie das verwendete Verfahren zur Adaption der bedingten Wahrscheinlichkeiten auszutauschen—ohne das Grundgerüst des Verfahrens modifizieren zu müssen. Die vorgestellte Methode kann als ein Rahmen interpretiert werden, der entsprechend den Anforderungen der Anwendungsituation instanziiert werden kann, wie z.B. unter Berücksichtigung von Genauigkeitskriterien, Laufzeit- oder Speicherressourcen.

Die Komplexität ist dementsprechend von den eingesetzten Algorithmen bestimmt. Grundsätzlich ist die Methode im Einklang mit der in Kapitel 4 vorgestellten Gesamtkonzeption in einen Offline- und einen Online-Anteil getrennt. Die Offline-Phase besteht aus der Anwendung eines Strukturlernverfahrens sowie gegebenenfalls zusätzlich aus der Ermittlung der repräsentativen Stichprobe der Strukturen $\boldsymbol{G}$. Der Online-Anteil arbeitet ebenfalls mit Strukturlernverfahren und CPT-Adaptionsmethoden. Für den Fall, dass der Strukturlernprozess zu aufwendig wird, um zur Laufzeit durchgeführt werden zu können, kann er in vielen Szenarien benutzeradaptiver Systeme in den Zeitraum zwischen zwei Interaktionsphasen mit dem Benutzer ausgelagert werden. Die während der letzten Interaktionsphase gesammelten Daten dienen als Adaptionsdaten, so dass für die nächste Interaktion mit dem Benutzer ein aktualisiertes Modell zur Verfügung steht. Gegebenenfalls kann die Durchführung des strukturellen Adaptionsschrittes sogar auf zusätzlicher Hardware bearbeitet werden.

In entsprechenden Anwendungsszenarien ist eine differentielle Variante der strukturellen Adaption mit Meta-Netzen denkbar. Analog zur in Kapitel 6 vorgestellten Methode zur differentiellen Adaption der CPTs können anstelle der manuellen Spezifikation eines globalen ESS-Parameters *lokale* ESS-Werte für die bedingten Wahrscheinlichkeiten des Meta-Netzes maschinell erlernt werden. Diese lokal unterschiedlichen ESS-Werte legen wie bei der differentiellen Adaption der bedingten Wahrscheinlichkeiten unterschiedliche Adaptionsgeschwindigkeiten für die verschiedenen potenziellen Kanten des betrachteten Bayes'schen Netzes fest.

Im Vergleich mit den existierenden Methoden zur Strukturadaption wie sie in Abschnitt 4.5.2 diskutiert wurden, besitzt die strukturelle Adaption mit Meta-Netzen einige Vorteile:

- Keine der anderen Methoden konstruiert und verwaltet ein explizites Meta-Modell der Domäne. Mit dem vorgestellten Verfahren können die Zusammenhänge zwischen einzelnen Teilaspekten der modellierten Domäne untersucht und interpretiert werden.

- Da die Bestimmung der potenziellen Kanten—und damit die Festlegung des zu betrachtende Suchraums—offline durchgeführt wird und die Ermittlung der adaptierten Strukturen zur Laufzeit mit Standardinferenzalgorithmen unter Verwendung des Meta-Netzes erfolgt, ist es möglich, eine große Menge an Strukturkandidaten zu betrachten. Die Kandidatenmenge kann über die anhand der Adaptionsdaten explorierten Menge hinausgehen und damit bei zeitlichen Veränderungen der Domäne früher beobachtete Modellierungsaspekte berücksichtigen, die alleine aufgrund der aktuellen Adaptionsdaten nicht zu erkennen sind. Dies steht in engem Zusammenhang mit dem nächsten Punkt.

- Im Unterschied zu den anderen Verfahren werden bei der strukturellen Adaption mit Meta-Netzen nicht nur lokale Veränderungen, d.h., das Einfügen oder Entfernen einzelner Kan-

ten, betrachtet, sondern im Zusammenhang mit dem Meta-Schlussfolgerungsprozesses Beziehung zwischen der Existenz bzw. dem Fehlen mehrerer Kanten explizit berücksichtigt (durch die Meta-Kanten).

- Das vorgestellte Verfahren ist im Gegensatz zu der Mehrzahl der anderen Methoden der strukturellen Adaption in der Lage, mit fehlenden Daten in der Trainings- bzw. Adaptionmenge umzugehen. Im Rahmen des generischen Charakters der Methode ist dazu lediglich ein entsprechendes Strukturlernverfahren (samt adäquater Bewertungsfunktion) wie etwa der SEM-Algorithmus in Kombination mit dem BIC einzusetzen.

- Das Verfahren setzt a priori keine Ordnung der Variablen oder Einschränkungen der Strukturen—beispielsweise eine Einschränkung auf Baumstrukturen—voraus. Das bestimmende Kriterium ist diesbezüglich die Wahl des eingesetzten Strukturlernverfahrens.

7.3.5 Analysen

Im Folgenden wird die strukturelle Adaption mit Meta-Netzen im Rahmen eines Vergleichs mit alternativen Lösungsmöglichkeiten des Adaptionsproblems evaluiert. Dies geschieht anhand synthetischer Daten, da die erhobenen Experimentaldaten hier im Wesentlichen aus zwei Gründen nicht geeignet sind: (a) Mit 72 bzw. 80 aufgezeichneten Trainings- bzw. Adaptionsfällen pro Versuchsperson stehen nicht genügend Daten für eine ausführliche Untersuchung struktureller Adaptionsverfahren zur Verfügung. Wie in Abschnitt 4.5.2 erläutert wurde, können strukturelle Veränderungen oder Abweichungen nur anhand einer größeren Menge an Adaptionsfällen erkannt werden. Es ist (b) nicht klar, ob genügend stark ausgeprägte *strukturelle* Unterschiede zwischen den Versuchspersonen der beiden Experimente existieren, d.h., es ist nicht unbedingt zu erwarten, dass eine Versuchsperson besser mit einer anderen Netzstruktur modelliert werden kann als die verbleibenden Personen. Zum Einsatz kommen deshalb (a) das auch von Hofmann (2000) im Rahmen der Einführung der Meta-Netze genutzte Netz (Abbildung 7.15), (b) das in Abbildung 7.16 dargestellte Bayes'sche Netz, das dem häufig in Lernstudien verwendeten ASIA-Netz entspricht (Lauritzen & Spiegelhalter, 1988) und (c) ein Beispielnetz eines hypothetischen benutzeradaptiven Systems, das den Nutzen der strukturellen Adaption in der Benutzermodellierung veranschaulichen soll. Das Beispielszenario wird im folgenden Abschnitt beschrieben.

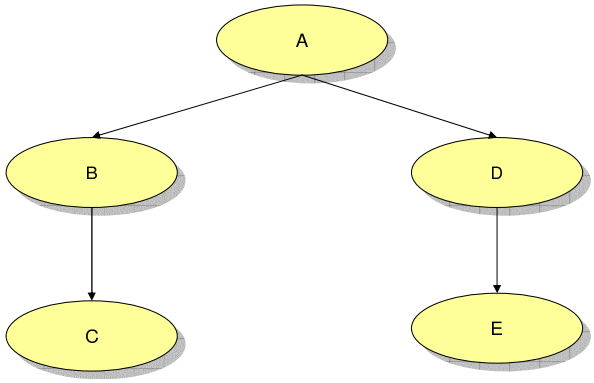


***Abbildung 7.15**: Beispielnetz von Hofmann (2000)*

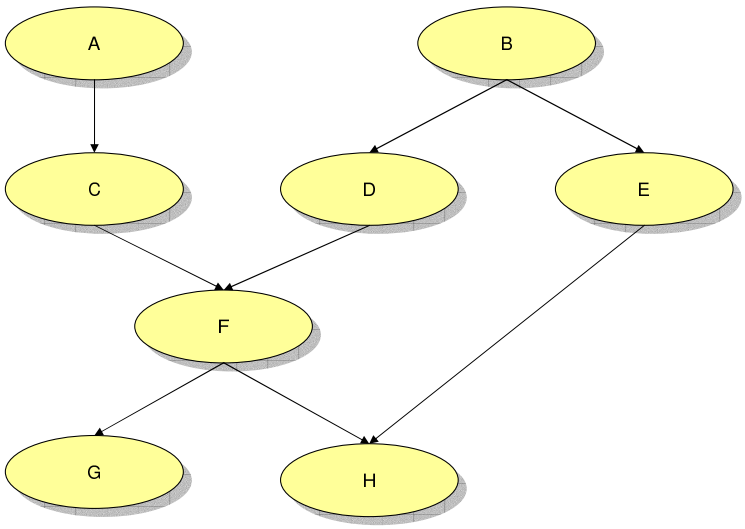


***Abbildung 7.16**: Beispielnetz* ASIA

7.3.5.1 Beispielszenario: Erweiterter naiver Bayes'scher Klassifizierer in benutzeradaptiven Systemen

Der naive Bayes'sche Klassifizierer wird häufig in benutzeradaptiven Systemen—insbesondere in Empfehlungssystemen—eingesetzt. In NEWSDUDE realisiert er beispielsweise das Langzeitgedächtnis des Systems (vgl. Abschnitt 2.6.3), das zur Bewertung der Nachrichtenartikel anhand der bislang erkannten Interessen des Benutzers dient. Breese et al. (1998) beschreiben, in welcher Weise diese besondere Variante eines Bayes'schen Netzes im Zusammenspiel mit maschinellen Lernverfahren als Inferenzmechanismus kollaborativer oder inhaltlich-basierter Empfehlungssysteme verwendet werden kann. Die dem naiven Bayes'schen Klassifizierer zugrunde liegende Annahme ist die bedingte Unabhängigkeit zwischen den Merkmalsvariablen: Ist die Klassenzugehörigkeit bekannt, so hat eine Veränderung des Wertes einer der Merkmalsvariablen keine Auswirkungen auf die Wahrscheinlichkeiten der verbleibenden.

Ein *erweiterter naiver Bayes'scher Klassifizierer* (ENBK, Friedman et al., 1997) hebt diese Beschränkung auf: Bei dieser Variante sind Kanten, d.h., direkte Einflüsse, zwischen den Merkmalsvariablen erlaubt (vgl. Abbildung 7.17). Friedman et al. (1997) konnten zeigen, dass mit diesem Ansatz die Klassifikationsleistung des naiven Bayes'schen Klassifizierers gesteigert werden kann.

In Empfehlungssystemen können damit Zusammenhänge zwischen den Merkmalen modelliert werden, die bei der Feststellung, ob ein Objekt für einen Benutzer interessant ist, eine Rolle spielen. Beispielsweise hängt die Bewertung eines Films mit dem Hauptdarsteller Sylvester Stallone bei einigen Kinobesuchern davon ab, ob es sich um eine Komödie oder einen Actionfilm handelt. In diesem Fall sollte das Einfügen einer Kante zwischen den zugehörigen Merkmalsvariablen HAUPTDARSTELLER und FILMART in einer Performanzverbesserung resultieren. Als Nebeneffekt erhält man mit erweiterten naiven Bayes'schen Klassifizierern oft besser interpretierbare Modelle, die die Transparenz des Empfehlungsprozesses erhöhen.

Das den folgende Analysen zugrunde liegende allgemeine Szenario kann am Beispiel von NEWSDUDE veranschaulicht werden: Zur Klassifikation der Nachrichten eines neuen Benutzers

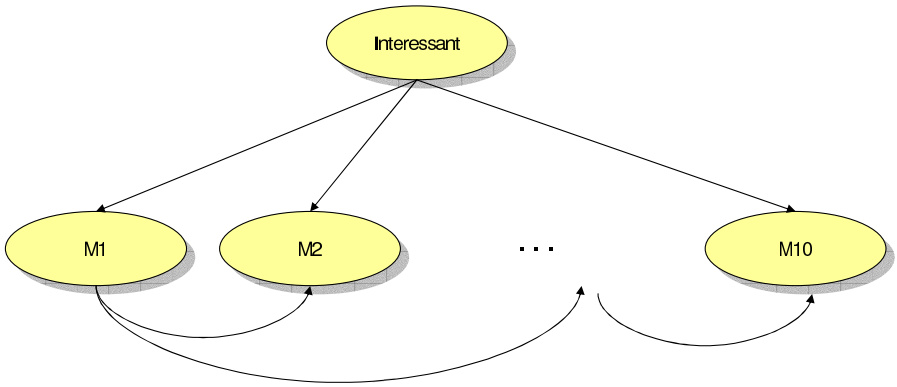


***Abbildung 7.17**: Erweiterter naiver Bayes'scher Klassifizierer*

wird ein als allgemeines Benutzermodell erlernter naiver Bayes'scher Klassifizierer benutzt, was den Vorteil hat, dass auf Testbewertungen von Artikeln verzichtet werden kann. Anhand der Rückmeldungen des Benutzers wird im Anschluss das Modell individualisiert, indem unter Verwendung struktureller Adaptionstechniken ein Übergang vom naiven Bayes'schen Klassifizierer zur erweiterten Variante erfolgt. Wird das System auf einer täglichen Basis genutzt, werden sehr schnell die zur Entscheidung beitragenden individuellen Zusammenhänge zwischen den Merkmalen von Nachrichtenartikeln in das Modell eingeflossen sein. Hier kann z.B. das Interesse eines Benutzers an Sportpolitik durch das Einfügen einer Kante zwischen den Merkmalen SPORT und POLITIK realisiert werden.

7.3.5.2 Methode

Zu jedem der drei Beispielnetze der Situationen (a) - (c) (Hofmann, ASIA, ENBK) wurden fünf strukturell in zufälliger Weise modifizierte Netze erzeugt: In Fall (a) wurden durchschnittlich zufällig 1.6 bzw. 1.0 Kanten hinzugefügt bzw. entfernt, entsprechend 1.8 bzw. 2.2 bei (b) und 1.5 bzw. 1.5 bei (c). Beim erweiterten naiven Bayes'schen Klassifizierer wurde dafür gesorgt, dass dessen Grundstruktur—die Kanten von der Klassenvariablen zu den Merkmalsvariablen—erhalten blieb. Es wurde also lediglich Strukturen erzeugt, die entweder zusätzliche Kanten zwischen Merkmalsvariable besitzen und/oder bei denen Kanten zwischen Merkmalsvariablen entfernt wurden. Mit Hilfe jedes dieser 15 zufällig modifizierten Bayes'schen Netze wurden je ein Datensatz generiert.

Die Evaluationsprozedur sah in jedem der drei Szenarien folgendermaßen aus: Je einer der Datensätze wurde zum Lernen eines Meta-Netzes genutzt, die restlichen vier dienten separat als Adaptionsdaten, d.h., insgesamt wurden pro Szenario 20 (5×4) Adaptionssituationen simuliert. Es werden die durchschnittlichen Ergebnisse aller 20 Kombinationen der Datensätze präsentiert. Um die Eigenschaften des Verfahren genauer zu untersuchen, wurden zwei prototypische Szenarien betrachtet: (i) ein erlerntes Bayes'sches Netz wird mit einem neuen statischen Einsatzkontext konfrontiert und (ii) ein erlerntes Netz wird einem sich verändernden Kontext ausgesetzt. Fall (i) stellt u.a. die Situation dar, dass ein erlerntes allgemeines Benutzermodell als Ausgangspunkt eines Adaptionsprozesses an einen neuen Benutzer genutzt wird. Bei (ii) kommt zusätzlich hinzu, dass eine Veränderung im Benutzerverhalten beobachtet wird. Letztere Situation wurde dadurch simuliert, dass in der Evaluationsprozedur zu einem bestimmten Zeitpunkt während des Adapti-

onsvorgangs der Adaptionsdatensatz ausgetauscht wurde, d.h., dass das zur Erzeugung der Daten verwendete Modell abrupt wechselte.

Als Performanzkriterium wurde das Standardmaß des (durchschnittlichen) *normalisierten logarithmischen Verlustes* verwendet, das ähnlich wie die Likelihood der Daten die Fähigkeit der adaptierten Netze bewertet, inwieweit sie in der Lage sind, die kompletten Adaptionsfälle zu modellieren:

$$\frac{1}{k}\sum_{i=1}^{k}(\ln P^*(D_i^{adapt}) - \ln P(D_i^{adapt})), \tag{7.5}$$

wobei P^* die anhand des zur Erzeugung des Datensatzes genutzten Netzes spezifizierte Wahrscheinlichkeit repräsentiert. Das aktuelle Bayes'sche Netz wird mit den k Adaptionsfällen des folgenden Adaptionsfensters bewertet und der zugehörige Durchschnitt ermittelt.

Der durch die strukturelle Adaption mit Meta-Netzen gegebene Rahmen wurde folgendermaßen instanziiert: Als Strukturlernverfahren wurde der SEM-Algorithmus eingesetzt und die Approximation der A-posteriori-Wahrscheinlichkeit wurde durch Berechnung der relativen Masse unter Verwendung der BIC-Bewertungsfunktion zur Approximation der marginalen Likelihood durchgeführt. Als CPT-Adaptionsverfahren kam AHUGIN zum Einsatz. Der Wert für m wurde auf 60 festgelegt und die globalen ESS-Werte zu den Beispielnetzen wurden für (i) und (iii) 5 bzw. 3 für (ii) vorgegeben, um unterschiedliche ESS-Werte zu betrachten.

Die alternativen Methoden, das Adaptionsproblem zu lösen, die zum Vergleich mit der strukturellen Adaption mit Meta-Netzen herangezogen wurden, sind:

- *Wiederholtes Batch- bzw. Neulernen des kompletten Bayes'schen Netzes:* Das aktuelle Bayes'sche Netz wird jeweils anhand der kompletten Menge an Adaptionsdaten, die bis zum betrachteten Zeitpunkt bekannt sind, erlernt. Diese Methode kann als Vergleichsmaßstab der Evaluation dienen, da sie alle verfügbaren Daten des aktuellen Kontexts nutzt, ohne durch einen vorhergehenden Offline-Teil mit in einer potenziell anderen Situation erhobenen Daten in Berührung gekommen zu sein. In der Praxis wird dieses Verfahren relativ schnell zu unakzeptablen Laufzeiten führen, da das wiederholte Lernen mit Trainingsdaten zunehmender Größe sehr zeitaufwendig wird.

- *Adaption der bedingten Wahrscheinlichkeiten mit der* AHUGIN*-Methode*: In diesem Fall wird die AHUGIN-Methode ohne jegliche Betrachtung struktureller Modifikationen angewendet. Es ist bekannt, dass die Methode in der Lage ist, einige der strukturellen Unzulänglichkeiten einer Struktur zu kompensieren (siehe z.B. Friedman & Goldszmidt, 1997)—insbesondere wenn die benötigte Kante (fälschlicherweise) bereits in der Struktur vorhanden ist, obwohl sie zuvor nicht in der Modellierung benötigt wurde. Die Performanz dieser Methode ist damit abhängig von der Qualität der festen Struktur des Netzes.

- *Lernen des kompletten Bayes'schen Netzes anhand der Fälle des letzten Adaptionsfensters:* Diese Methode stellt im Vergleich zum wiederholten Batchlernen den entgegengesetzten Extrempunkt des Spektrums der möglichen Adaptionsansätze dar: Das aktuelle Bayes'sche Netz wird jeweils nur anhand der k Adaptionsfälle des letzten Adaptionsfensters erlernt. Das entscheidende Kriterium ist hierbei die optimale Wahl von k. Eine zu kleine Wahl resultiert in geringer Qualität der Ergebnisse, wohingegen ein zu großer Wert (zu) lange Laufzeiten benötigt. Außerdem kann die Methode starken zufallsbedingten Schwankungen unterliegen, da es ihr nicht möglich ist, Wissen über die Domäne über einen längeren Zeitraum zu aggregieren.

- *Unmodifizierte Struktur nach dem Meta-Lernen, Lernen der bedingten Wahrscheinlichkeiten anhand des letzten Adaptionsfensters:* Dieser Ansatz verwendet die initial (nach dem Meta-Lernen) ermittelte Struktur des Bayes'schen Netzes, ohne weitere Modifikationen im Verlauf des Adaptionsprozesses vorzunehmen. Die bedingten Wahrscheinlichkeiten werden wiederholt anhand der Daten des letzten Adaptionsfensters erlernt.

- *Adaption der Struktur, Lernen der bedingten Wahrscheinlichkeiten anhand des letzten Adaptionsfensters:* Hier wird die Struktur des aktuellen Bayes'schen Netzes gemäß der beschriebenen Prozedur modifiziert; anstelle des AHUGIN-Verfahrens zur Adaption der CPTs werden die bedingten Wahrscheinlichkeiten anhand des letzten Adaptionsfensters erlernt.

Die beiden letztgenannten Methoden wurden in die vergleichende Evaluation aufgenommen, um die Netto-Auswirkungen der entwickelten strukturellen Adaptionsmethode ohne den Beitrag der (bereits existierenden) CPT-Adaptionstechniken (hier: AHUGIN) untersuchen zu können.

7.3.5.3 Ergebnisse

Die Abbildungen 7.18, 7.19 und 7.20 zeigen die Ergebnisse der drei Beispielszenarien bei unterschiedlichen Fenstergrößen k.

Wie erwartet produzierte das wiederholte Neulernen insgesamt die besten Ergebnisse. Die strukturelle Adaption mit Meta-Netzen war in der Lage, bessere Ergebnisse als die verbleibenden Alternativen zu erzielen. Ignoriert man den strukturellen Part, d.h., wendet man AHUGIN auf die CPTs der Ausgangsstruktur an, so beobachtet man im Fall weniger aggregierter Adaptionsdaten, d.h., bei kleinen k-Werten, eine zumindest vergleichbare, in den meisten Fällen sogar bessere Performanz mit den meisten der Ansätze. In Situationen mit einer größeren Anzahl an für einen strukturellen Adaptionsschritt verfügbaren Daten, sei es bei struktureller Adaption oder bei wiederholtem Neulernen, beobachtet man eine schlechtere relative Performanz von AHUGIN, obwohl die absoluten Ergebnisse erwartungsgemäß gleich bleiben. Weiterhin verhalten sich AHUGIN und die strukturelle Adaption mit Meta-Netzen in einer initialen Phase sehr ähnlich. Diese Phase endet mit dem Adaptionsschritt, zu dem das strukturelle Adaptionsverfahren zum ersten Mal in der Lage ist, das initiale—durch das Meta-Lernen ermittelte—Modell strukturell zu modifizieren, d.h., wenn es genügend Adaptionsdaten verarbeitet hat, um das mit der durch die ESS-Werte festgelegten initialen Konfidenz versehene Modell zu verändern. Dies lässt sich daran ablesen, dass dieser Zeitpunkt nach 6 bzw. 4 strukturellen Veränderungen eintritt, jeweils einen Schritt später als durch die vorgegebenen ESS-Werte—der äquivalenten Beispielgröße—von 5 bzw. 3 vorgegeben. Das wiederholte Strukturlernen anhand der Daten des letzten Adaptionsfensters zeigt starke Schwankungen bei einer geringen Anzahl an aggregierten Adaptionsfällen. Diese Schwankungen können mit den zufälligen Variationen der kleinen Adaptionsdatensätze erklärt werden. Dieser Effekt ist bei größeren Adaptionsfenstern nicht mehr so stark ausgeprägt. Wie erwartet kann diese Methode die stärksten absoluten Verbesserungen bei einer Vergrößerung des Adaptionsfensters erzielen. Hinsichtlich der beiden verbleibenden Varianten zur Untersuchung der strukturellen Adaption mit Meta-Netzen (unter Verzicht auf die CPT-Adaption) lässt sich feststellen, dass die Adaption der Struktur alleine betrachtet schon eine deutliche Verbesserung der Performanz bewirkt (auch hier nach einer initialen Phase, d.h., sobald genügend Adaptionsfälle vom Verfahren gesehen wurden, um eine strukturelle Veränderung durchführen zu können).

Zu den in den Beispieldomänen erlernten Meta-Netzen lässt sich sagen, dass durchschnittlich ein bis zwei Meta-Kanten zur Kodierung der Abhängigkeiten zwischen den eigentlichen Kanten

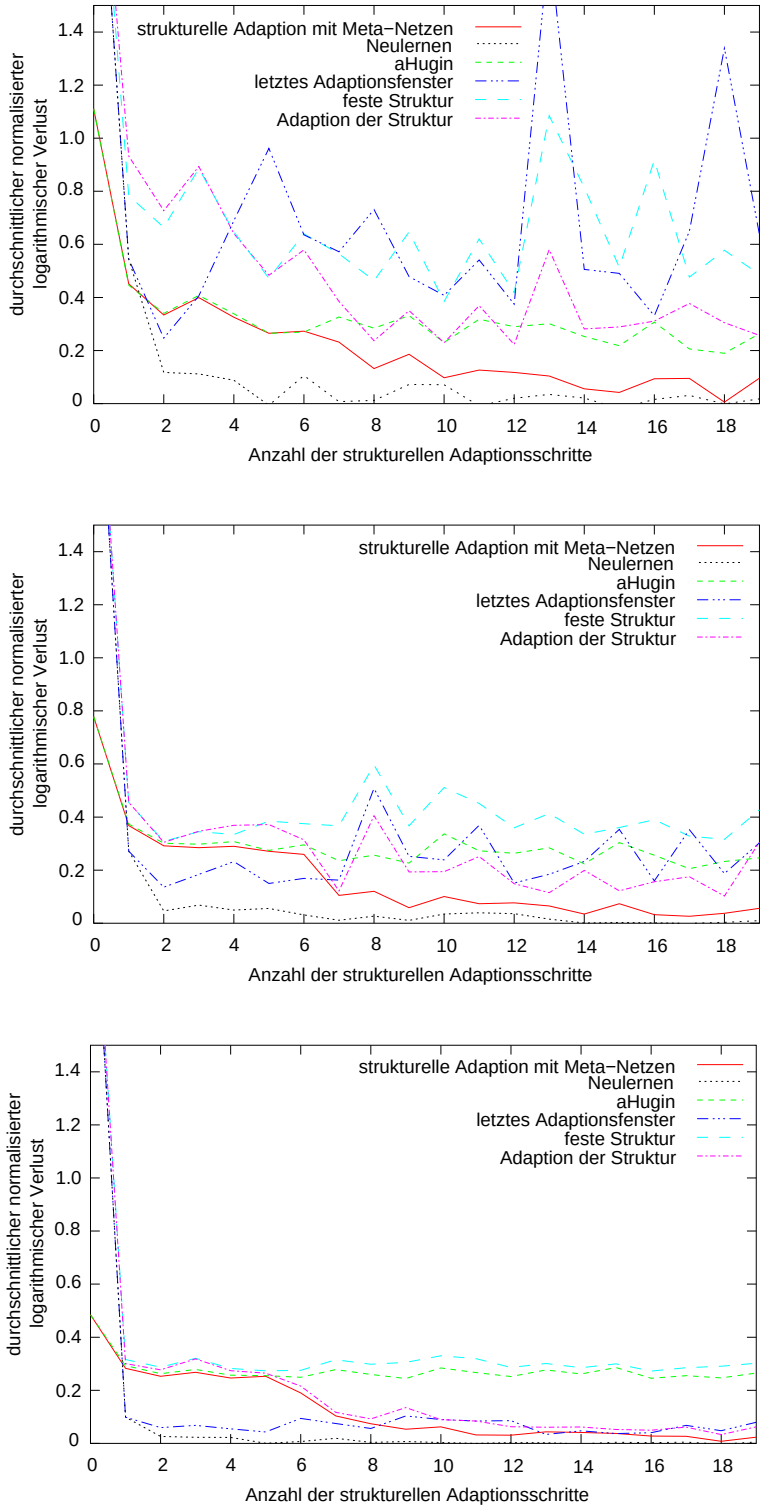


***Abbildung 7.18**: Ergebnisse der strukturellen Adaption (Hofmann-Netz), $k = 25, 50, 150$.*

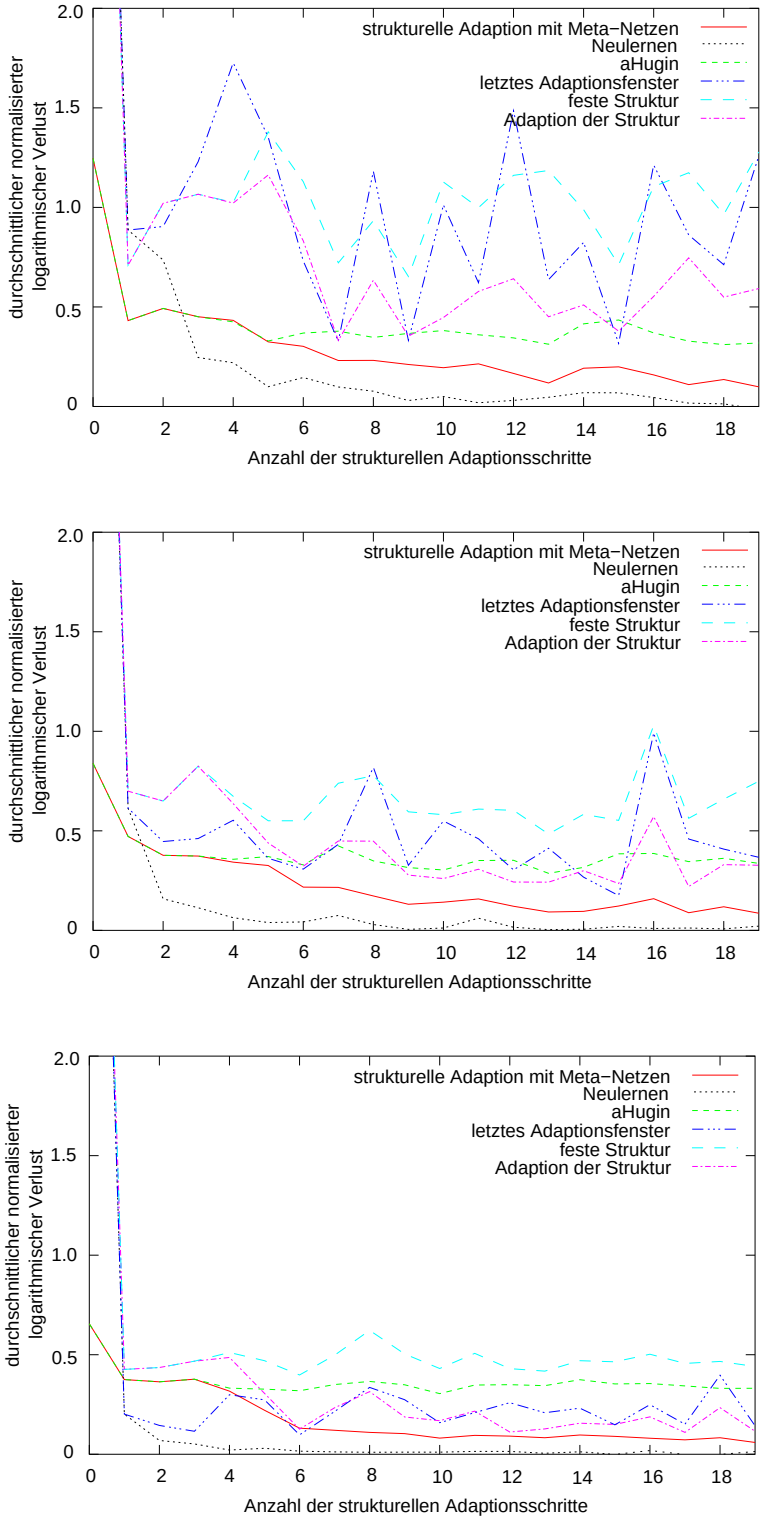


***Abbildung 7.19**: Ergebnisse der strukturellen Adaption (*ASIA*-Netz), $k = 25, 50, 100$.*

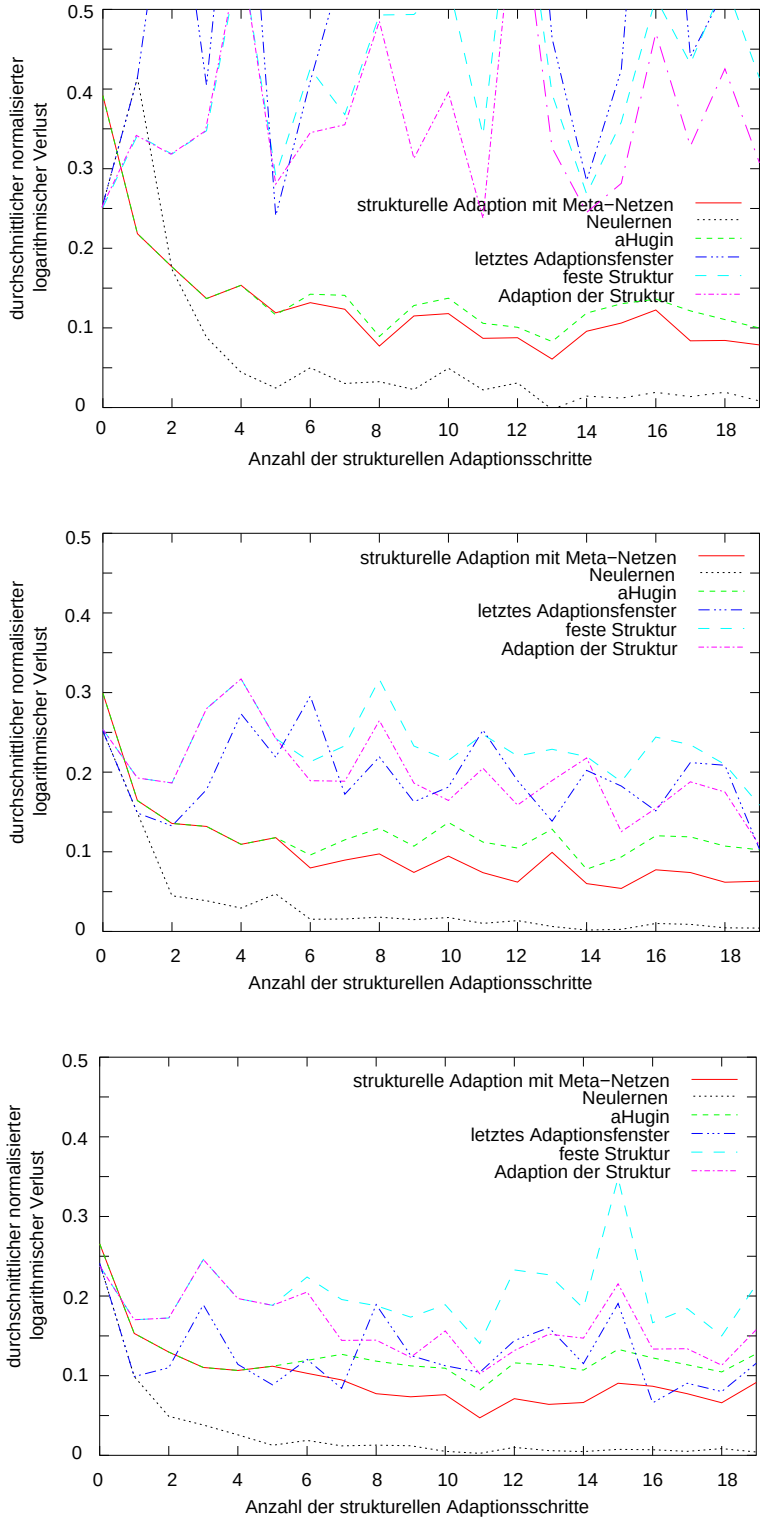


***Abbildung 7.20**: Ergebnisse der strukturellen Adaption (ENBK), $k = 75, 150, 200$.*

der Bayes'schen Netze erkannt werden. Dies korrespondiert bei (i) mit den von Hofmann (2000) berichteten Ergebnissen.

Die zweite Studie simulierte eine abrupte Veränderung der zu modellierenden Situation. Dazu wurde nach 20 Adaptionschritten die zur Adaption verwendete Adaptionsmenge ausgetauscht und durch eine Sammlung von Fällen ersetzt, die mit einem anderen der fünf zufällig modifizierten Netze erzeugt wurde. Es wurde ein Fading Factor von 0.98 spezifiziert, um den „Vergessensmechanismus" von AHUGIN zu steuern. Für k wurde 25 bzw. 75 gewählt, d.h., Situationen mit relativ wenigen Adaptionsdaten, und ein initialer ESS-wert von 5 vorgegeben. Abbildung 7.21 zeigt die Resultate.

Bis zum 20. Adaptionschritt wird das gleiche Verhalten wie in den vorhergehenden Analysen beobachtet. Dann bewirkt die abrupte Veränderung der Adaptionsdaten—wie erwartet—eine schlechte Performanz aller Adaptionsalternativen. Danach versuchen die Methoden diese Veränderung in ihren Modellen zu erfassen. Eine Ausnahme bildet das wiederholte Lernen anhand des letzten Adaptionsfensters: Diese Methode zeigt weiterhin die bekannten hohen Variationen in den Ergebnissen. Abgesehen von einer Ausnahme bleiben die relativen Resultate gleich. Bis zu einem gewissen Grad berücksichtigen die das AHUGIN-Verfahren in irgendeiner Weise nutzenden Adaptionsmethoden die alten Daten, die bis zum Auftreten der abrupten Veränderung gesehen wurden. Das Ausmaß wird durch den festgelegten Fading Factor bestimmt. Die angesprochene Ausnahme und das interessanteste Ergebnis der Untersuchung stellt man im Zusammenhang mit der strukturellen Adaption mit Meta-Netzen bei den beiden ersten Beispielnetzen fest: Sie ist in der vorliegenden Situation in der Lage, bessere Ergebnisse zu erzielen als das wiederholte Neulernen. Da das Neulernen die alten Adaptionsdaten nicht „vergessen" kann, kann die Methode hier nicht ihre Überlegenheit bei der Vorhersage neuer Daten aufrecht erhalten—zusätzlich zum wachsenden Bedarf an Rechenzeit.

Die Ergebnisse des erweiterten naiven Bayes'schen Klassifizierers weichen von denjenigen der ersten beiden Analysesituationen ab. In dieser Situation steht mit jeweils 75 Adaptionsfällen mehr Information für die Verfahren zur Verfügung. Dadurch werden von allen Methoden absolut gesehen relativ gute Ergebnisse erzielt. Das wiederholte Neulernen kann seine theoretische Überlegenheit in der Praxis über den gesamten Analysehorizont von 50 Adaptionsschritten realisieren. Die strukturelle Adaption mit Meta-Netzen und AHUGIN produzieren sehr ähnliche Ergebnisse. Einer der Gründe dafür ist die Vorgabe des A-priori-Wissens, das viele der potentielle Strukturmodifikationen ausschließt. Dies spiegelt sich auch in der relativ geringen absoluten Überlegenheit des Modells wider, das Strukturmodifikation erlaubt, im Vergleich zu fester Struktur—bei gleichzeitigem Erlernen der bedingten Wahrscheinlichkeiten anhand des letzten Adaptionsfensters. Dennoch stellt man anhand Betrachtung der letzten beiden Adaptionalternativem einen Mehrwert der strukturellen Adaption fest.

In solchen Situationen, in denen genügend Adaptionsdaten bzw. A-priori-Wissen für die alternativen Adaptionsmethoden zur Verfügung stehen, besteht der Vorteil der strukturellen Adaption mit Meta-Netzen im Wesentlichen in der erhöhten Interpretierbarkeit der Netze. Mit dieser Methode können von den Systemen jeweils aktuelle Strukturen eingesetzt werden, die zwar möglicherweise keine Performanzsteigerung gegenüber beispielsweise dem AHUGIN-Verfahren erzielen können, aber kausale Zusammenhänge adäquat repräsentieren. In diesem konkreten Beispiel können entsprechende Zusammenhänge zwischen den Merkmalvariablen des erweiterten naiven Bayes'schen Klassifizierers erkannt und im Modell abgebildet werden.

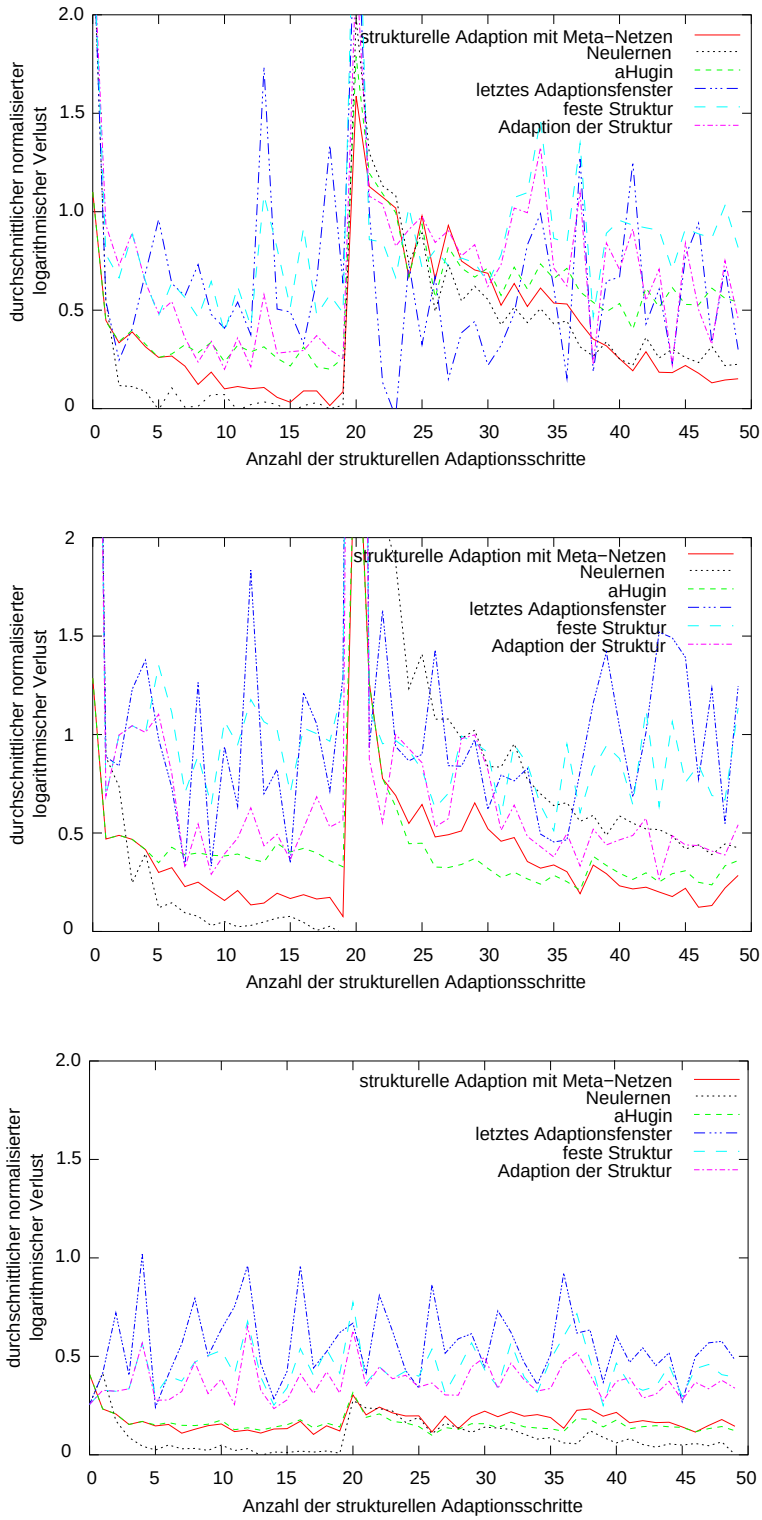


***Abbildung 7.21**: Ergebnisse der strukturellen Adaption bei abrupter Veränderung der Situation; Hofmann $k = 25$, ASIA $k = 25$, ENBK $k = 75$, $ff = 0.98$*

7.3.5.4 Diskussion

Eine für den Erfolg des Verfahrens entscheidende Problemstellung ist die möglichst optimale Wahl der Größe k des Adaptionsfensters. Sie kann mit Standardtechniken aus dem Bereich des maschinellen Lernens erfolgen: Anhand von Daten der Einsatzdomäne kann durch Kreuzvalidierungsmethoden ein brauchbarer Wert ermittelt werden. Dabei wird die Gesamtmenge der verfügbaren Daten in Trainings- und Adaptionsdaten separiert, die beispielsweise im Rahmen eines Hillclimbing-Verfahrens dazu dienen, solche k-Werte zu bestimmen, die zu guten Adaptionsergebnissen führen.

Die in dieser Studie beschriebenen Resultate können durch den Einsatz alternativer Verfahren zur Behandlung der Subkomponenten des Grundgerüsts der strukturellen Adaption mit Meta-Netzen verbessert werden—jedoch auf Kosten der benötigten Rechenkapazitäten. Hierbei handelt es sich im Wesentlichen um aufwendigere Verfahren zur Ermittlung einer repräsentativen Stichprobe an Strukturen $\boldsymbol{G}$ und exakteren Verfahren zur Berechnung der A-posteriori-Wahrscheinlichkeiten. Eine Möglichkeit besteht diesbezüglich in der Anwendung von MCMC-Techniken. Unter Berücksichtigung dieser Aspekte kann die vorgestellte Untersuchung als eine untere Schranke der Performanz der strukturellen Adaption mit Meta-Netzen hinsichtlich der Modellierungsqualität interpretiert werden. Eine höhere Qualität wird im Allgemeinen durch eine höhere Komplexität des Verfahrens erkauft.

Die Eigenschaft, die das Verfahren insbesondere gegenüber anderen strukturellen Adaptionsverfahren (vgl. Abschnitt 4.5.2) für einen Einsatz in benutzeradaptiven Systemen qualifiziert, ist die explizite Repräsentation der strukturellen Unsicherheit in Form der Meta-Netze. Mit ihnen wird die Transparenz der Adaptionsentscheidungen erhöht, wie anhand des Beispiels in Abschnitt 7.2.4 erläutert wurde. Hinzu kommt, dass die in einem Adaptionsschritt anfallende Arbeit semi-offline erledigt werden kann, d.h., z.B. zwischen zwei Interaktionsphasen des Systems mit dem gleichen Benutzer. Wird dies zu Zeiten mit geringer Systemlast oder auf zusätzlicher Hardware durchgeführt, kann eine Beeinträchtigung des Laufzeitverhaltens des Systems vermieden werden.

7.4 Zusammenfassende Diskussion

In diesem Kapitel wurde gezeigt, dass sich die Betrachtung struktureller Aspekte im Rahmen des Lern- bzw. Adaptionsvorgangs der Benutzermodelle in benutzeradaptiven Systemen lohnen kann. Obwohl in den meisten Anwendungsdomänen die Behandlung der bedingten Wahrscheinlichkeiten eine genügend hohe Qualität der Benutzermodelle liefert, kann der (zusätzliche) Einsatz von Strukturlern- und/oder -adaptionsverfahren zu einem verbesserten Verständnis der zu modellierenden Domäne führen, wovon im Konstruktionsprozess des benutzeradaptiven Systems profitiert werden kann.

Gerade hinsichtlich der Kombination mit Erklärungskomponenten für Bayes'sche Netze erscheint der Strukturfall von besonderer Bedeutung, um die Interpretierbarkeit der betrachteten Bayes'schen Netze zu erhöhen. Für Erklärungskomponenten ist es wichtig, jederzeit eine adäquate Struktur zur Verfügung zu haben, die als Grundlage der Generierung von Begründungen zu Adaptionsentscheidungen des benutzeradaptiven Systems dienen kann. Dazu werden Strukturadaptionsverfahren benötigt, die möglichst in der Lage sind, anhand einer geringen Adaptionsmenge, Veränderungen des Benutzerverhaltens erfassen und in das Benutzermodell einfließen lassen zu können. Mit der strukturellen Adaption mit Meta-Netzen wurde ein entsprechendes Verfahren vorgestellt, das neben dem eigentlichen Adaptionsmechanismus auch Meta-Informationen zum Benutzerverhalten liefert. In dieser Weise wird sowohl die Interpretierbarkeitseigenschaft der Modelle an aktuelle Veränderungen angepasst, als auch in vielen Fällen eine Verbesserung der Performanz erzielt.

8 ZUSAMMENFASSUNG UND AUSBLICK

8.1 Zusammenfassung

In der vorliegenden Arbeit wurde der Einsatz maschineller Lernverfahren für Bayes'sche Netze in benutzeradaptiven Systemen behandelt. Auf der Grundlage der Definition Bayes'scher Netze sowie wichtigen Verfahren bzw. relevanten Erweiterungen dieses Konzepts wurde in Kapitel 2 ein Überblick der aktuellen Forschung zur Anwendung Bayes'scher Netze in benutzeradaptiven Systemen gegeben. Im Vordergrund standen dabei die auf Bayes'schen Netzen aufbauenden Einflussdiagramme und dynamischen Bayes'schen Netze sowie eine Untersuchung, inwieweit maschinelle Lernverfahren bereits in den entsprechenden Systemen zum Einsatz kommen. Dabei zeigte sich, dass Bayes'sche Netze in einer Vielzahl unterschiedlicher Szenarien, die eine Behandlung von Unsicherheit erfordern, eingesetzt werden, dabei aber oftmals manuell anhand theoretischer Überlegungen spezifiziert werden. In denjenigen Fällen, in denen maschinelle Lernverfahren Bayes'scher Netze verwendet wurden, um die vorhandenen empirischen Daten auszunutzen, kamen Verfahren zum Einsatz—meist beschränkt auf das Erlernen der bedingten Wahrscheinlichkeiten der Netze—, die nicht auf die speziellen Anforderungen des Benutzermodellierungskontexts optimiert sind. Verfahren, die diese Anforderungen berücksichtigen bzw. entsprechende Eigenschaften der Domänen benutzeradaptiver Systeme ausnutzen können, wurden bislang nicht entwickelt.

Es folgte in Kapitel 3 eine Übertragung des allgemeinen maschinellen Lernproblems auf den Kontext benutzeradaptiver Systeme. Diesbezüglich wurden Kriterien identifiziert, deren Berücksichtigung in der Entwurfsphase eines solchen Systems von entscheidender Bedeutung für einen erfolgreichen Einsatz maschineller Lernverfahren sein können. Im Einzelnen sind dies:

- *geringe Anzahl an verfügbaren Trainingsdaten*,
- *inter-individuelle Unterschiede*,
- *dynamische Domänen*,
- *Komplexität bzw. Effizienz der Verfahren*,
- *Interpretierbarkeit der Benutzermodelle*,
- *besondere Eigenschaften der Trainings- bzw. Adaptionsdaten*,
- *Integration von A-priori-Wissen*,
- *Evaluation der erlernten Modelle im Rahmen der Evaluation der benutzeradaptiven Systeme*.

Es wurden allgemeine Lösungsansätze diskutiert, die bei diesem Problemstellungen zur Anwendung kommen können.

Generische Benutzermodellierungsumgebungen stellen zum Teil Implementationen häufig benötigter Lernverfahren zur Verfügung. Es wurden zwei solche Systeme vorgestellt, die den Schwerpunkt auf die Integration maschineller Lernverfahren in den Gesamtansatz der Benutzermodellierung legen. Ebenso wurde der wichtige Fall der Empfehlungssysteme diskutiert, die große kommerzielle Bedeutung erlangt haben und wesentlich zum Transfer der Forschungsergebnisse der Benutzermodellierung mit Schwerpunkt auf der Anwendung maschineller Lernverfahren beigetragen haben. Einige erfolgreich eingesetzte maschinelle Lernverfahren wurden bezüglich ihrer Eignung für benutzeradaptive Systeme anhand ausgewählter Beispielsysteme untersucht und hinsichtlich der angeführten Kriterien bewertet.

Den in der vorliegenden Arbeit entwickelten Methoden liegt die in Kapitel 4 vorgestellte Gesamtkonzeption des maschinellen Lernens Bayes'scher Netze in benutzeradaptiven Systemen zugrunde. Es handelt sich dabei um einen integrativen Rahmen, der die grundsätzlichen Zusammenhänge zwischen der Art der vorhandenen Daten, dem A-priori-Wissen, der offline stattfindenden Akquisition von Benutzermodellen in Form Bayes'scher Netze sowie der im Laufzeitbetrieb vorgenommenen Adaption der Modelle zusammenfasst. Das damit verfolgte Ziel besteht in der Behandlung der angeführten Kriterien eines Einsatzes maschineller Lernverfahren in benutzeradaptiven Systemen im speziellen Fall Bayes'scher Netze. Aus einem Repertoire existierender und in dieser Arbeit neu entwickelter Methoden können bei der Konstruktion benutzeradaptiver Systeme auf der Basis Bayes'scher Netze gemäß den Anforderungen des Einsatzszenarios adäquate Verfahren ausgewählt und im Rahmen der Gesamtkonzeption eingeordnet werden. Ein benutzeradaptives System, das maschinelle Lernverfahren für Bayes'sche Netze verwendet, bildet in dieser Weise eine Instanziierung der generischen integrativen Konzeption. Typischerweise muss nur ein Teil der Gesamtkonzeption im zu entwickelnden System implementiert werden, um den vorhandenen Anforderungen zu genügen.

Tabelle 8.1 gibt einen Überblick über die Beiträge der in der vorliegenden Arbeit entwickelten Einzelverfahren unter Berücksichtigung der identifizierten Kriterien des maschinellen Lernens in benutzeradaptiven Systemen.

Aspekte	**Qualitative Constraints**	**Differentielle Adaption**	**Strukturelle Adaption**
Wenige Trainingsdaten	⋆	⋆	⋆
Fehlende Daten	⋆		
Inter-individuelle Unterschiede		⋆	⋆
Dynamische Domänen		⋆	⋆
Komplexität / Effizienz im Online-Betrieb		⋆	⋆
Interpretierbarkeit	⋆		⋆
Integration von A-priori-Wissen	⋆		

***Tabelle 8.1**: Übersicht über die Beiträge der in der vorliegenden Arbeit entwickelten Verfahren zum maschinellen Lernen Bayes'scher Netze für benutzeradaptive Systeme*

Mit dem in dieser Arbeit neu entwickelten Verfahren des *Lernens mit qualitativen Constraints* werden wichtige Teile der Gesamtkonzeption bzw. der genannten Kriterien behandelt. Das Verfah-

ren ermöglicht das Erlernen interpretierbarer Bayes'scher Netze hinsichtlich der wichtigen Aufgabe des Lernens der bedingten Wahrscheinlichkeiten. Durch das Ausnutzen von vorhandenem A-priori-Wissen über qualitative Zusammenhänge zwischen den im Bayes'schen Netz betrachteten Variablen können gerade bei wenigen, unvollständigen Trainingsdaten die Ergebnisse des Lernvorgangs im Vergleich zu den bislang verwendeten Verfahren sowohl hinsichtlich der (numerischen) Qualität der Modellierung als auch bezüglich des Aspektes der Interpretierbarkeit deutlich verbessert werden.

Das vorgestellte Verfahren basiert auf der Erweiterung der zum Erlernen der bedingten Wahrscheinlichkeiten eingesetzten Bewertungsfunktion durch einen „Strafterm", der potenzielle Lösungen, die nicht mit dem spezifizierten A-priori-Wissen konsistent sind, schlechter bewertet. Dadurch wird der Suchvorgang so durch den Lösungsraum geführt, dass „schlechte" lokale Optima der ursprünglichen Bewertungsfunktion vermieden werden. Auf diese Art und Weise wird der beim maschinellen Lernen bekannte Overfitting-Effekt vermindert. Es wurde gezeigt, dass das Verfahren als eine Variante des Bayes'schen Lernens interpretiert werden kann.

Das Verfahren des Lernens mit qualitativen Constraints wurde sowohl anhand synthetisch erzeugter als auch empirischer Daten evaluiert. Die Ergebnisse dieser Evaluation zeigen, dass das Verfahren in der Lage ist—gerade in den interessanten Situationen mit wenigen Lerndaten—, das Overfitting zu eliminieren bzw. deutlich zu verringern und die Interpretierbarkeit der erlernten Benutzermodelle zu gewährleisten oder zumindest deutlich zu erhöhen. Alleine das letztere Ergebnis rechtfertigt bereits den Einsatz des Verfahrens auch in Lernsituationen, in denen genügend Trainingsdaten vorhanden sind, um ein Benutzermodell ohne wesentliches Overfitting zu erlernen—zur Gewährleistung der Interpretierbarkeit des vom System eingesetzten Benutzermodells.

Anschließend wurde ein zweiter zentraler Bestandteil der integrativen Gesamtkonzeption diskutiert: die Identifikation und Behandlung individueller Unterschiede im Rahmen der Adaption des offline erlernten Bayes'schen Netzes zur Laufzeit des Systems an den individuellen Benutzer. Dabei stellt sich die zentrale Frage, in welcher Art und Weise die Anpassung vorgenommen werden soll. Es wurden diesbezüglich für den Fall der bedingten Wahrscheinlichkeiten alternative Ansätze untersucht sowie ein neues Verfahren entwickelt, das speziell auf den Benutzermodellierungskontext zugeschnitten ist. Betrachtet wurden das *individuelle*, das *allgemeine*, das *parametrisierte* sowie die Neuentwicklung, das *differentiell adaptive* Benutzermodell. Dabei stellen die beiden ersten Ansätze die entgegengesetzten Extrempunkte des Spektrums der Adaptionsmöglichkeiten dar, d.h., beim individuellen Ansatz werden nur Daten eines einzigen Benutzer zur Akquisition des Benutzermodells verwendet. Im Gegensatz dazu basiert das allgemeine Modell nur auf Daten anderer Benutzer, es findet keine Adaption statt. Das parametrisierte Benutzermodell nutzt dynamische Bayes'sche Netze mit individuellen Parametervariablen, welche die Eigenschaften der Benutzer charakterisieren, die in unterschiedlichen Ausprägungen vorliegen können.

Die neu entwickelte *Methode der differentiellen Adaption der bedingten Wahrscheinlichkeiten* nutzt existierende Adaptionsverfahren, um unterschiedliche Aspekte des Benutzermodells mit verschiedenen Adaptionsgeschwindigkeiten anzupassen. Modellbereiche, die sich durch große individuelle Unterschiede auszeichnen, werden schneller anhand der gesammelten Adaptionsdaten modifiziert als Bereiche, in denen die meisten Benutzer größtenteils übereinstimmen. Dazu werden—vereinfacht dargestellt—anhand der Varianzen der individuellen Benutzermodelle Adaptionsparameter in Form von lokalen ESS-Werten bestimmt, welche die Adaptionsgeschwindigkeiten im Rahmen des Bayes'schen Adaptionsvorgangs festlegen.

Die alternativen Adaptionsansätze wurden im Rahmen einer empirischen Evaluation anhand von experimentell im READY-Szenario erhobenen Datensätze verglichen. Es zeigte sich, dass die

beiden adaptiven Ansätze, d.h., das parametrisierte und das differentiell adaptive Modell, insgesamt die beste Performanz aufweisen. Dennoch wurden auch einzelne Situationen beobachtet, in denen das individuelle oder das allgemeine Modell die besten Ergebnisse erreicht. Deshalb ist bei der Entscheidung für einen Ansatz zu beachten, ob Anforderungen der betrachteten Domäne existieren, die zu einer Präferenz eines der Verfahren führen. In diesem Zusammenhang spielen Laufzeitanforderungen und die Möglichkeiten zur Datenerhebung eine Rolle.

Der in Abschnitt 2.6 präsentierte Überblick zum Stand der Forschung des Einsatzes Bayes'scher Netze in benutzeradaptiven Systemen zeigt, dass sich bislang meist auf das Erlernen und die Adaption der bedingten Wahrscheinlichkeiten konzentriert wurde. In dieser Arbeit wurde der Strukturfall sowohl hinsichtlich des Lern- als auch des Adaptionsproblems untersucht. Neben den dadurch potenziell zu erzielenden Performanzsteigerungen ist der Einsatz entsprechender Verfahren möglicherweise sinnvoll, um ein besseres Verständnis der Zusammenhänge zwischen den Variablen zu erlangen, wie in Abschnitt 7.2 beschrieben. Der Ansatz des *strukturellen Lernens mit Meta-Netzen* von Hofmann (2000) wurde *im Kontext benutzeradaptiver Systeme* angewendet, mit dem Ziel, das Verständnis der der modellierten Domäne zugrunde liegenden Struktur zu erhöhen. Meta-Netze bieten die Möglichkeit, die strukturelle Unsicherheit, die insbesondere beim Strukturlernen mit wenigen Trainingsdaten eine Rolle spielt, kompakt zu repräsentieren und auszuwerten. Aufbauend auf dieser Methode wurde mit der *strukturellen Adaption mit Meta-Netzen* ein Adaptionsverfahren entwickelt, das die Struktur eines Bayes'schen Netzes an Veränderungen des Kontexts anpassen kann. Eine Evaluierung anhand alternativer Methoden sowie die zugehörige Diskussion der Eigenschaften der Verfahren unterstützen die prinzipielle Eignung für einen Einsatz in benutzeradaptiven Systemen.

Die zur Evaluation der betrachteten Verfahren verwendeten empirischen Daten stammen aus psychologisch motivierten Experimenten des READY-Projekts, die die Untersuchung der (subjektiv empfundenen) kognitiven Belastung von Personen in unterschiedlichen Szenarien zum Ziel hatten, wie z.B. die Interaktion mit technischen Geräten unter Zeitdruck oder die Interaktion mit einem mobilen System während situativ bedingter Ablenkungen.

Mit der vorliegenden Arbeit wurden folgende konkreten Beiträge geleistet:

- *Entwicklung einzelner, speziell auf den Kontext benutzeradaptiver Systeme zugeschnittener maschineller Lernverfahren für Bayes'sche Netze:*
  - Lernen interpretierbarer bedingter Wahrscheinlichkeiten mit qualitativen Constraints
  - Differentielle Adaption bedingter Wahrscheinlichkeiten zur Erfassung und Behandlung individueller Unterschiede zwischen den Benutzern
  - Strukturelle Adaption von Benutzermodellen in Form Bayes'scher Netze mit Meta-Netzen
- *Integration existierender und neu entwickelter Verfahren in einer Gesamtkonzeption des maschinellen Lernens Bayes'scher Netze für und in benutzeradaptiven Systemen*
- *Identifikation von Kriterien hinsichtlich der Anwendung maschineller Lernverfahren in benutzeradaptiven Systemen und deren Behandlung im Fall Bayes'scher Netze mit den entwickelten Methoden* (vgl. Tabelle 8.1).
- *Empirische Fundierung der Benutzermodelle des* READY*-Szenarios:*

- kognitive Ressourcenbeschränkungen eines Benutzers können mit Hilfe erlernter dynamischer Bayes'scher Netze anhand von Symptomen seiner gesprochenen Sprache erkannt werden
- Empirisch fundierte Adaption des Präsentationsmodus eines ressourcenadaptiven Dialogsystems anhand eines erlernten Bayes'schen Netzes zur Fehlervermeidung bzw. Erhöhung der Arbeitsgeschwindigkeit

8.2 Konzeptuelle Aspekte möglicher weiterer Forschung

Obwohl die entwickelten Verfahren aus konkreten Problemstellungen des Kontexts benutzeradaptiver Systeme heraus motiviert sind, besitzen sie auch in anderen Szenarien ein erhebliches Anwendungspotenzial. Das Verfahren des Lernens interpretierbarer bedingter Wahrscheinlichkeiten mit qualitativen Constraints kann ohne Modifikation in allen Situationen zum Einsatz kommen, in denen erklärbare Bayes'sche Netze (mit verborgenen Variablen) von Vorteil sind. Die Transparenz des Schlussfolgerungsprozesses ist eine wünschenswerte Eigenschaft aller Expertensysteme (Wahlster, 1981; Teach & Shortliffe, 1984), so dass damit potenziell alle Expertensysteme, die maschinelle Lernverfahren für Bayes'sche Netze einsetzen, von dem vorgestellten Verfahren profitieren können. Darüber hinaus eignet sich das Verfahren für alle Lernaufgaben, die sich durch eine geringe Menge an Trainingsdaten auszeichnen, um die im maschinellen Lernen bekannte Overfitting-Problematik zu behandeln.

Die differentielle Adaption Bayes'scher Netze kann auch dann eingesetzt werden, wenn die individuellen Netze keine Benutzermodelle repräsentieren, sondern andere ausgezeichnete Objekte, die zwar in ihrer Grundstruktur übereinstimmen, aber einige individuell variierende Dimensionen besitzen. Eine Übertragung auf den allgemeinen Ansatz objekt-orientierter Bayes'scher Netze sowie insbesondere die aktuell im Fokus der Forschung stehenden probabilistischen relationalen Modelle erscheint damit sinnvoll.

Das Verfahren der strukturellen Adaption mit Meta-Netzen ist allgemein einsetzbar, wenn die vorgegebenen Rahmenbedingungen entsprechend der Diskussion aus Abschnitt 7.3.5.4 für einen Einsatz gegenüber anderen existierenden Adaptionsverfahren sprechen.

Wegen der vielfältigen Instanziierungsmöglichkeiten des durch die Gesamtkonzeption des maschinellen Lernens Bayes'scher Netze für benutzeradaptive Systeme gegebenen Rahmens wurde in der vorliegenden Arbeit auf eine *detaillierte Betrachtung des Zusammenspiels der Einzelverfahren* verzichtet. Dazu ist zu sagen, dass es in den wenigsten Fällen zu einer Instanziierung der kompletten Konzeption kommen wird; in den meisten Fällen genügt ein Teil der Verfahren, um die angestrebte Funktionalität des benutzeradaptiven Systems zu erzielen. So wird in vielen benutzeradaptiven Systemen, die sehr wenige Interaktionen—im Extremfall eine einzige—mit ihren Benutzern aufweisen, auf den gesamten Adaptionsteil der Konzeption verzichtet. Andere Systeme wiederum verzichten mangels verfügbarer Trainingsdaten auf die Offline-Akquisition eines Benutzermodells und basieren auf einem manuell spezifizierten allgemeinen Ausgangsmodell, das im Verlauf der Interaktion an den individuellen Benutzer adaptiert wird. Es ist diesbezüglich von Interesse, praxis-relevante Instanziierungen der Konzeption in kommerziellen Systemen zu identifizieren und in entsprechender Weise unter Berücksichtigung des Zusammenspiels der Einzelmethoden zu evaluieren.

In diesem Zusammenhang ist es auch wünschenswert, *weitere empirische Evaluationen* der vorgestellten Einzelverfahren in verschiedenen potenziellen Einsatzszenarien, die sich durch un-

terschiedlichste Eigenschaften und Anforderungen auszeichnen, durchzuführen. Hierzu zählt beispielsweise der Einsatz in benutzeradaptiven Systemen auf Desktopsystemen im Gegensatz zur Anwendung auf mobilen Geräten, die sich durch sehr unterschiedliche Möglichkeiten der Datenerhebung mit Sensoren und technischen Ressourcen wie Speicherkapazität und Rechenleistung voneinander unterscheiden.

Eine konsequente Erweiterung der Arbeit hinsichtlich des Schwerpunktes der Interpretierbarkeit bzw. Erklärbarkeit der Bayes'schen Netze besteht im *Aufsatz existierender bzw. in der Entwicklung neuer geeigneter Erklärungskomponenten*, welche die mit den entwickelten Verfahren erzielten Lern- und Adaptionsergebnisse gezielt ausnutzen können. Dabei eignen sich im Zusammenhang mit dem Verfahren der qualitativen Constraints existierende Verfahren, wie in Abschnitt 2.1.7 beschrieben. Für den Fall der differentiellen Adaption oder auch der strukturellen Adaption mit Meta-Netzen bietet es sich an, neue Methoden zu entwickelt, die explizit auf die vorgenommenen Modifikationen eingehen, um individuelle Unterschiede zu erklären. In diesem Zusammenhang könnte untersucht werden, ob eine Fokussierung der Erklärung des Schlussfolgerungsprozesses auf die Unterschiede zwischen den Benutzern möglicherweise zu einer erhöhten Akzeptanz führt. Betrachtet man beispielsweise ein Empfehlungssystem, so sind es meist individuelle Präferenzen, die die letztendliche Entscheidung aus einer Auswahl an alternativen Produkten begründen. Gerade bei falschen Empfehlungen eines solchen Systems, könnten Erklärungen, wie „Im Allgemeinen bevorzugen Personen mit den Eigenschaften ... Produkt A. Sie unterscheiden sich davon aber in der Eigenschaft ..., weshalb für Sie Produkt B von Interesse erscheint.", für einen Benutzer hilfreich sein, um das Verhalten des Systems zu verstehen.

Konzeptuell ist eine *Übertragung des Grundprinzips der differentiellen Adaption auf andere Formalismen* möglich. Beispielsweise könnte das häufig in Empfehlungssystemen eingesetzte Verfahren der nächsten Nachbarn um eine entsprechende Komponente erweitert werden. Bei der Bestimmung geeigneter Nachbarn werden im Allgemeinen meist mehrere Kriterien betrachtet und in einer Bewertung kombiniert. Es müssen typischerweise Parameter festgelegt werden, die z.B. als Schwellwerte dienen, ab denen eine Person oder ein Objekt als geeigneter Nachbar angesehen wird. Die Werte dieser Parameter können sich von Situation zu Situation unterscheiden. Betrachtet man z.B. ein kollaboratives CD-Empfehlungssystem, dann könnte für Klassik-CDs ein modifiziertes Kriterium zur Bestimmung von Käufern mit ähnlichem Geschmack gelten als bei Jazz-CDs. Die Parameter des Ähnlichkeitsmaßes spielen dabei die Rolle der lokalen ESS bei der differentiellen Adaption Bayes'scher Netze. Es ist vorstellbar, dass sie anhand verfügbarer Daten mit ähnlichen Techniken ermittelt werden können.

8.3 Technische Aspekte möglicher weiterer Forschung

Neben den angeführten eher konzeptuellen Punkten kann die vorliegende Arbeit als Ausgangspunkt weiterer technischer Entwicklungen des maschinellen Lernens Bayes'scher Netze für benutzeradaptive Systeme dienen.

Aus Sicht der Konstruktion benutzeradaptiver Systeme bietet sich eine Integration der vorgestellten Methoden in das Konzept der in Abschnitt 2.5 vorgestellten objekt-orientierten Bayes'schen Netze und die probabilistischen relationalen Modelle an. Mit der Verfügbarkeit entsprechender Softwarewerkzeuge wird auch der Einsatz dieser Ansätze in benutzeradaptiven Systemen an Bedeutung gewinnen. Gerade die Möglichkeit situationsspezifische Netze aufzubauen, erscheint im Zusammenhang mit der wachsenden Zahl mobiler Systeme interessant und vielversprechend.

Ähnliches gilt für die probabilistischen relationalen Modelle, die in Bezug auf große relationale Nutzerdatenbanken insbesondere im E-Commerce an Bedeutung gewinnen werden.

Die diskutierten Verfahren gehen in der vorgestellten Form bei kontinuierlichen Werten von bereits diskretisierten Trainings- bzw. Adaptionsdaten aus. In weiteren Arbeiten sollten zumindest die existierenden *automatischen Methoden zur Diskretisierung von Daten* (Friedman & Goldszmidt, 1997; Kozlov & Koller, 1997) in die präsentierten Verfahren integriert werden. Gegebenenfalls kann untersucht werden, ob sich eine Anpassung dieser Algorithmen an den Benutzermodellierungskontext lohnt. Geeignete Diskretisierungen wirken sich sowohl auf die Komplexität der Verfahren und die Genauigkeit der Ergebnisse, als auch auf die Modellierung der individuellen Unterschiede zwischen Benutzern aus.

In der vorliegenden Arbeit wurde die Existenz erklärender, verborgener Variablen manuell in die Struktur eingebracht. In den letzten Jahren wurden Verfahren zur *automatischen Detektion verborgener Variablen anhand der verfügbaren Daten* entwickelt (Elidan, Lotner, Friedman & Koller, 2000). Es erscheint interessant, zu untersuchen, ob bzw. inwieweit sich der Einsatz solcher Verfahren für die Benutzermodellierung eignet. Eine Frage, die hierbei u.a. im Vordergrund steht, ist die semantische Interpretation der gefundenen verborgenen Variablen: Welche Aspekte des Benutzermodells repräsentieren solche vom Lernalgorithmus postulierten Variablen, was kann über ihre Zusammenhänge mit den anderen Variablen ausgesagt werden?

Allgemeiner können Verfahren untersucht werden, die zusätzlich zu den bedingten Wahrscheinlichkeiten und der Struktur auch alle Variablen des Bayes'schen Netzes erlernen—nicht nur die verborgenen. Dies stellt ein allgemeines Problem im maschinellen Lernen dar. Ein Spezialfall ist das Feature-Selection-Problem—beispielsweise im Zusammenhang mit der Merkmalsextraktion des naiven Bayes'schen Klassifizierers (vgl. Abschnitt 3.3). Übertragen auf den Benutzermodellierungskontext lautet die Fragestellung: Welche Aspekte des Benutzerverhaltens, der Ziele, der Interessen usw. sind für ein erfolgreiches benutzeradaptives System im Schlussfolgerungsprozess relevant?

In einigen Szenarien werden im Rahmen der Akquisition der initialen Benutzermodelle dem Benutzer Testfragen gestellt, wie z.B. die Bewertung von Testobjekten in Empfehlungssystemen. Hier könnte sich der Einsatz von Verfahren des *aktiven Lernens* (engl. *active learning*) auszahlen. Solche Verfahren berücksichtigen bei der sequentiellen Auswahl der Testobjekte, welche unter ihnen die größte erwartete Qualitätsverbesserung bewirken können. Tong und Koller (2000) haben ein solches allgemeines Verfahren für Bayes'sche Netze entwickelt. Eine entsprechende Übertragung auf den Kontext benutzeradaptiver Systeme kann das Ziel weiterführender Arbeit sein. Als Ausgangspunkt könnte das Verfahren der differentiellen Adaption dienen: Durch die Interpretation der lokalen ESS-Werte als ein Qualitätsmaß für die Konfidenz der Modellierung hat man eine Entscheidungshilfe, welche Teile der Benutzermodelle noch verbessert werden können bzw. welche Teile noch auf einer unsicheren Basis stehen. Allerdings müssen noch andere Faktoren berücksichtigt werden, wie etwa der erwartete Nutzen einer Verbesserung der infrage kommenden Aspekte der Benutzermodelle.

Zwar wurden in der vorliegenden Arbeit Methoden betrachtet, die eine Adaption der Struktur Bayes'scher Netze ermöglichen, diese gehen jedoch davon aus, dass die auftretenden Strukturen zumindest ähnlich zueinander sind. In Fällen, die sich durch strukturell deutlich unterschiedliche Netze—und damit Benutzermodelle—auszeichnen, könnte der *Einsatz von Model-Averaging-Verfahren* (vgl. Abschnitt 4.4.2) zum Erfolg führen. Dies spielt insbesondere bei solchen benutzeradaptiven Systemen eine Rolle, deren potenzielle Benutzer sehr heterogene Eigenschaften bzw. Verhaltensweisen zeigen. Einen Aspekt, der im READY-Projekt verfolgt wird, stellen Gruppen von

jüngeren vs. älteren Systembenutzern dar. Es ist bekannt, dass es deutliche Unterschiede im Interaktionsverhalten dieser beiden Benutzergruppen gibt. Mit dem Model-Averaging wäre es möglich, gewisse allgemeine Modelle vorzuhalten, die im Schlussfolgerungsprozess entsprechend gewichtet würden. Eine Kombination mit den Verfahren des Lernens mit qualitativen Constraints und der differentiellen Adaption ist denkbar.

Die in der dieser Arbeit verwendeten empirischen Daten und der Quellcode der entwickelten Verfahren ist im WWW über die READY-Projekt-Webseite (http://w5.cs.uni-sb.de/∼ready, Stand Dezember 2002) erhältlich, so dass die in dieser Arbeit vorgestellten Experimente jederzeit von interessierten Forschern nachvollzogen werden können.

A Versuch der Herleitung einer geschlossenen Darstellung des M-Schrittes mit qualitativen Constraints

In diesem Abschnitt wird gezeigt, dass eine Herleitung einer Formel in geschlossener Darstellung für den M-Schritt des EM-Verfahrens, die qualitative Constraints berücksichtigt, in der üblichen Form nicht möglich ist. Dies war der Grund für die Entwicklung des in Abschnitt 5.2.3.2 beschriebenen hybriden EM-Ansatzes.

Im Allgemeinen besteht die Aufgabe des M-Schritts des EM-Algorithmuses in der Maximierung der erwarteten Log-Likelihood. überträgt man die Herleitung aus Bishop (1995) auf den Kontext des Lernens Bayes'scher Netze, so erhält man folgende Maximierungsaufgabe:

$$\begin{aligned}\boldsymbol{\theta}^{neu} = \arg\max_{\theta} \sum_{ijk} \sum_{l=1}^{s} & P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}^{alt}) \\ & \times \ln[P(D_l \mid x_{ij}, pa_k(X_i)) P(x_{ij}, pa_k(X_i) \mid \boldsymbol{\theta}^{alt})].\end{aligned} \tag{A.1}$$

Die erweiterte Log-Likelihood bringt einen zusätzlichen Term in die Bewertungsfunktion ein, der die „Strafe“ darstellt und wegen der verletzten qualitativen Constraints abgezogen werden muss:

$$\begin{aligned}\boldsymbol{\theta}^{neu} = \arg\max_{\theta} \quad & \textstyle\sum_{ijk} \sum_{l=1}^{s} \quad P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}^{alt}) \\ & \qquad \times \ln[P(D_l \mid x_{ij}, pa_k(X_i)) P(x_{ij}, pa_k(X_i) \mid \boldsymbol{\theta}^{alt})] \\ & - \mathit{violation}(\boldsymbol{\theta}, \boldsymbol{C}).\end{aligned} \tag{A.2}$$

Ein Versuch, diese Funktion zu maximieren, führt zu einem nicht-linearen Gleichungssystem, für das der Autor keine analytische geschlossene Lösung gefunden hat.

Die Maximierung wird durch das Gleichsetzen der partiellen Ableitungen mit Null und dem Lösen des zugehörigen Gleichungssystems vorgenommen. Um zu gewährleisten, dass das Resultat die Forderung $\sum_j \theta_{ijk}^{neu} = 1$ erfüllt, müssen Lagrange-Multiplikatoren λ_{ik} in die Gleichungen aufgenommen werden:

$$\sum_l \frac{P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}^{alt})}{\theta_{ijk}^{neu}} + \lambda_{ik}(\sum_j \theta_{ijk}^{neu} - 1) - \frac{\partial}{\partial \theta_{ijk}^{neu}} \mathit{violation}(\boldsymbol{\theta}^{neu}, \boldsymbol{C}) = 0. \tag{A.3}$$

Nutzt man aus, das $\sum_j \theta_{ijk}^{neu} = 1$ gilt, erhält man:

$$\sum_l \frac{P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}^{alt})}{\theta_{ijk}^{neu}} + \lambda_{ik} - \frac{\partial}{\partial \theta_{ijk}^{neu}} violation(\boldsymbol{\theta}^{neu}, \boldsymbol{C}) = 0. \tag{A.4}$$

Multipliziert man nun mit θ_{ijk}^{neu}:

$$\sum_l P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}^{alt}) + \theta_{ijk}^{neu}\lambda_{ik} - \theta_{ijk}^{neu}\frac{\partial}{\partial \theta_{ijk}^{neu}} violation(\boldsymbol{\theta}^{neu}, \boldsymbol{C}) = 0. \tag{A.5}$$

Um die λ_{ik} zu eliminieren, wird ausgenutzt, dass $\sum_j \theta_{ijk}^{neu} = 1$ gilt, und nach λ_{ik} aufgelöst:

$$\lambda_{ik} = -\sum_j \sum_l P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}^{alt}) + \sum_j \theta_{ijk}^{neu}\frac{\partial}{\partial \theta_{ijk}^{neu}} violation(\boldsymbol{\theta}^{neu}, \boldsymbol{C}) \tag{A.6}$$

Eingesetzt in Gleichung A.4:

$$\begin{aligned}&\sum_l \frac{P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}^{alt})}{\theta_{ijk}^{neu}} - \sum_{j'} \sum_l P(x_{ij'}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}^{alt}) \\ &+ \sum_{j'} \theta_{ij'k}^{neu}\frac{\partial}{\partial \theta_{ij'k}^{neu}} violation(\boldsymbol{\theta}^{neu}, \boldsymbol{C}) - \frac{\partial}{\partial \theta_{ijk}^{neu}} violation(\boldsymbol{\theta}^{neu}, \boldsymbol{C}) = 0.\end{aligned} \tag{A.7}$$

Nochmal multipliziert mit θ_{ijk}^{neu}:

$$\begin{aligned}&\sum_l P(x_{ij}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}^{alt}) - \theta_{ijk}^{neu} \sum_{j'} \sum_l P(x_{ij'}, pa_k(X_i) \mid D_l, \boldsymbol{\theta}^{alt}) \\ &+ \theta_{ijk}^{neu} \sum_{j'} \theta_{ij'k}^{neu}\frac{\partial}{\partial \theta_{ij'k}^{neu}} violation(\boldsymbol{\theta}^{neu}, \boldsymbol{C}) - \theta_{ijk}^{neu}\frac{\partial}{\partial \theta_{ijk}^{neu}} violation(\boldsymbol{\theta}^{neu}, \boldsymbol{C}) = 0.\end{aligned} \tag{A.8}$$

An dieser Stelle treten die Probleme auf. Man hat hier eine Menge nicht-linearer Gleichungen (in θ_{ijk}^{neu}), die stark voneinander abhängig sind. Der Autor hat keine analytische Methode gefunden, dieses Gleichungssystem zu lösen. Möglicherweise könnte eine adäquat gewählte $violation$-Funktion dieses Problem lösen. Dies ist ein offenes Problem in der vorliegenden Arbeit.

Literaturverzeichnis

Ahman, F. & Waern, A. (2001). Modelling the interests of a news service user. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 204–206). Berlin: Springer.

Albrecht, D. W., Zukerman, I. & Nicholson, A. E. (1998). Bayesian models for keyhole plan recognition in an adventure game. *User Modeling and User-Adapted Interaction*, *8*, 5–47.

Alspector, J., Kolcz, A. & Karunanithi, N. (1997). Feature-based and clique-based user models for movie selection: A comparative study. *User Modeling and User-Adapted Interaction*, *7*(4), 279–304.

Balabanovic, M. (1998). Exploring versus exploiting when learning user models for text recommendation. *User Modeling and User-Adapted Interaction*, *8*(1).

Bangsø, O., Langseth, H. & Nielsen, T. D. (2001). Structural learning in object oriented domains. In J. Kolen & I. Russell (Hrsg.), *Proceedings of the 14th International FLAIRS Conference.*

Bauer, E., Koller, D. & Singer, Y. (1997). Update rules for parameter estimation in Bayesian networks. In D. Geiger & P. P. Shenoy (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference* (S. 3–13). San Francisco: Morgan Kaufmann.

Bauer, M. (1996). *Ein evidenztheoretischer Ansatz zur Planerkennung.* Dissertation, Universität des Saarlandes.

Bayes, T. (1763). An essay towards solving a problem in the doctrines of chances. *Philosophical Transactions*, *3*, 370–418. (Reprinted in Biometrica, 45:296-315, 1958)

Beierle, C. & Kern-Isberner, G. (2000). *Methoden wissensbasierter Systeme. Grundlagen, Algorithmen, Anwendungen.* Vieweg-Verlag.

Berger, J. O. (1985). *Statistical Decision Theory and Bayesian Analysis.* New York: Springer.

Berthold, A. (1998). *Repräsentation und Verarbeitung sprachlicher Indikatoren für kognitive Ressourcenbeschränkungen.* Diplomarbeit, Lehrstuhl Wahlster, Fachrichtung Informatik, Universität des Saarlandes, Saarbrücken.

Berthold, A. & Jameson, A. (1999). Interpreting symptoms of cognitive load in speech input. In J. Kay (Hrsg.), *UM99, User Modeling: Proceedings of the Seventh International Conference* (S. 235–244). Wien: Springer.

Billsus, D. & Pazzani, M. J. (1999). A hybrid user model for news story classification. In J. Kay (Hrsg.), *UM99, User Modeling: Proceedings of the Seventh International Conference* (S. 99–108). Wien: Springer.

Binder, J., Koller, D., Russell, S. & Kanazawa, K. (1997). Adaptive probabilistic networks with hidden variables. *Machine Learning, 29*, 213–244.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition.* Oxford: Clarendon Press.

Bohnenberger, T., Brandherm, B., Großmann-Hutter, B., Heckmann, D. & Wittig, F. (2002). Empirically grounded decision-theoretic adaptation to situation-dependent resource limitations. *Künstliche Intelligenz, 16*(3), 10–16.

Bohnenberger, T. & Jameson, A. (2001). When policies are better than plans: Decision-theoretic planning of recommendation sequences. In J. Lester (Hrsg.), *IUI 2001: International Conference on Intelligent User Interfaces* (S. 21–24). New York: ACM.

Borth, M. (2002). Learning form multiple Bayesian networks for the revision and refinement of expert systems. In J. Köhler & G. Lakemeyer (Hrsg.), *Proceedings of the 25th German Conference on Artificial Intelligence (KI2002).*

Bouckaert, R. R. (1995). *Bayesian Belief Networks: From Construction to Inference.* Dissertation, Universität Utrecht.

Bradley, K., Rafter, R. & Smyth, B. (2000). Case-based user profiling for content personalisation. In P. Brusilovsky, O. Stock & C. Strapparava (Hrsg.), *Adaptive hypermedia and adaptive web-based systems: Proceedings of AH 2000* (S. 62–72). Berlin: Springer.

Brandherm, B. (2000). *Rollup-Verfahren für komplexe dynamische Bayessche Netze.* Diplomarbeit, Lehrstuhl Wahlster, Fachrichtung Informatik, Universität des Saarlandes, Saarbrücken.

Breese, J., Heckerman, D. & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In G. F. Cooper & S. Moral (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Fourteenth Conference* (S. 43–52). San Francisco: Morgan Kaufmann.

Brusilovsky, P. (2001). Adaptive hypermedia. *User Modeling and User-Adapted Interaction, 11*, 87–110.

Bunt, A. & Conati, C. (2001). Modeling exploratory behaviour. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 219–221). Berlin: Springer.

Bunt, A., Conati, C., Huggett, M. & Muldner, K. (2001). On improving the effectiveness of open learning environments through tailored support for exploration. In J. Moore, C. Redfield & W. Johnson (Hrsg.), *Proceedings of the 10th Interational Conference on Artificial Intelligence in Education.* San Antonio, Texas.

Buntine, W. (1991). Theory refinement on Bayesian networks. In B. D'Ambrosio, P. Smets & P. P. Bonissone (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Seventh Conference* (S. 52–60). San Mateo, CA: Morgan Kaufmann.

Buntine, W. (1996). A guide to the literature on learning probabilistic networks from data. *IEEE Transactions on Knowledge and Data Engineering, 8,* 195–210.

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*. (Im Druck)

Castillo, E., Gutierrez, J. M. & Hadi, A. S. (1997). *Expert Systems and Probabilistic Network Models.* Berlin: Springer.

Cheng, J., Greiner, R., Kelly, J., Bell, D. & Liu, W. (2002). Learning Bayesian networks from data: An information-theory based approach. *Artificial Intelligence, 137*(1–2), 43–90.

Chickering, D. M., Geiger, D. & Heckerman, D. (1994). *Learning Bayesian networks is NP-hard* (Tech. Rep. Nr. MSR-TR-94-17). Microsoft Research.

Chin, D. N. (1989). KNOME: Modeling what the user knows in UC. In A. Kobsa & W. Wahlster (Hrsg.), *User models in dialog systems* (S. 74–107). Berlin: Springer.

Chin, D. N. (2001). Empirical evaluation of user models and user-adapted systems. *User Modeling and User-Adapted Interaction, 11*, 181–194.

Chow, C. K. & Liu, C. N. (1968). Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory, 14*, 462–467.

Cloete, I. & Zurada, J. M. (Hrsg.). (1999). *Knowledge-Based Neurocomputing.* Cambridge, MA: MIT Press.

Conati, C. & VanLehn, K. (1999). A student model to assess self-explanation while learning from examples. In J. Kay (Hrsg.), *UM99, User Modeling: Proceedings of the Seventh International Conference* (S. 303–305). Wien: Springer.

Conati, C. & VanLehn, K. (2001). Providing adaptive support to the understanding of instructional material. In J. Lester (Hrsg.), *IUI 2001: International Conference on Intelligent User Interfaces.* New York: ACM.

Cook, R. & Kay, J. (1994). The justified user model: A viewable, explained user model. In A. Kobsa & D. Litman (Hrsg.), *UM94, User Modeling: Proceedings of the Fourth International Conference* (S. 145–150). Boston, MA: User Modeling, Inc.

Cooper, G. F. (1990). The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence, 42*, 393–405.

Cooper, G. F. & Herskovits, E. (1992). A Bayesian method for the induction of probabilistic networks from data. *Machine Learning, 9*, 309–347.

Dagum, P., Galper, A. & Horvitz, E. (1992). Dynamic network models for forecasting. In D. Dubois, M. P. Wellman, B. D'Ambrosio & P. Smets (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Eighth Conference* (S. 41–48). San Francisco: Morgan Kaufmann.

Decker, B. (2001). *Implementation von Lernverfahren für Bayes'sche Netze mit versteckten Variablen* (Tech. Rep.). Lehrstuhl Wahlster, Fachrichtung Informatik, Universität des Saarlandes, Saarbrücken. (READY-Memo 81)

DeGroot, M. H. (1970). *Optimal Statistical Decisions.* New York: McGraw-Hill.

Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, 39*, 1–38.

Druzdzel, M. J. (1996). Qualitative verbal explanations in Bayesian belief networks. *Artificial Intelligence and Simulation of Behaviour Quarterly, 94*, 43–54.

Druzdzel, M. J. & Simon, H. A. (1993). Causality in Bayesian belief networks. In D. Heckerman & A. Mamdani (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Nineth Conference* (S. 3–11). San Mateo, CA: Morgan Kaufmann.

Druzdzel, M. J. & van der Gaag, L. C. (1995). Elicitation of probabilities for belief networks: Combining qualitative and quantitative information. In P. Besnard & S. Hanks (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Eleventh Conference* (S. 141–148). San Francisco: Morgan Kaufmann.

Druzdzel, M. J. & van der Gaag, L. C. (2000). Building probabilistic networks: Where do the numbers come from? *IEEE Transactions on Knowledge and Data Engineering, 12*(4), 481–486.

Duda, R. & Hart, P. (1973). *Pattern Recognition and Scene Analysis.* John Wiley and Sons.

Elidan, G., Lotner, N., Friedman, N. & Koller, D. (2000). Discovering hidden variables: A structure-based approach. In *Proceedings of the 2000 Conference on Neural Information Processing Systems.*

Fisher, R. (1912). On an absolute criterion for fitting frequency curves. *Messenger of Mathematics, 41*, 155–160.

Fisher, R. (1922). On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society, 22*, 700–725.

Forbes, J., Huang, T., Kanazawa, K. & Russell, S. (1995). The BATmobile: Towards a Bayesian Automated Taxi. In C. S. Mellish (Hrsg.), *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (S. 1878–1885). San Mateo, CA: Morgan Kaufmann.

Friedman, N. (1997). Learning belief networks in the presence of missing values and hidden variables. In *Proceedings of the 13th International Conference on Machine Learning.*

Friedman, N. (1998). The Bayesian structural EM algorithm. In G. F. Cooper & S. Moral (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Fourteenth Conference* (S. 129–138). San Francisco: Morgan Kaufmann.

Friedman, N., Geiger, D. & Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning, 29*, 131–163.

Friedman, N. & Goldszmidt, M. (1996). Discretizing continuous attributes while learning Bayesian neworks. In *Proceedings of the 13th International Conference on Machine Learning* (S. 157–165). Morgan Kaufmann.

Friedman, N. & Goldszmidt, M. (1997). Sequential update of Bayesian network structure. In D. Geiger & P. P. Shenoy (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference* (S. 165–174). San Francisco: Morgan Kaufmann.

Friedman, N., Goldszmidt, M. & Wyner, A. (1999). Data analysis with Bayesian networks: A bootstrap approach. In K. B. Laskey & H. Prade (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the 15th Conference* (S. 196–205). S.F., Cal.: Morgan Kaufmann.

Friedman, N. & Koller, D. (2002). Being Bayesian about network structure: A Bayesian approach to structure discovery in Bayesian networks. *Machine Learning*.

Fung, R. M. & Crawford, S. L. (1990). Constructor: A system for the induction of probabilistic models. In W. Dietterich, Tom; Swartout (Hrsg.), *Proceedings of the Eighth National Conference on Artificial Intelligence* (S. 762–769). MIT Press.

Geiger, D., Heckerman, D., King, H. & Meek, C. (1998). *Stratified exponetial families: Graphical models and model selection* (Tech. Rep. Nr. MSR-TR-98-31). Redmond, Washington: Microsoft Research.

Geiger, D., Heckerman, D. & Meek, C. (1996). Asymptotic model selection for directed networks with hidden variables. In E. Horvitz & F. V. Jensen (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Twelfth Conference* (S. 283–290). San Francisco: Morgan Kaufmann.

Gervas, P. (2001). Modeling literary style for semi-automatic generation of poetry. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 231–233). Berlin: Springer.

Getoor, L., Friedman, N., Koller, D. & Pfeffer, A. (2001). Learning probabilistic relational models. In S. Dzeroski & N. Lavrac (Hrsg.), *Relational Data Mining.* Springer-Verlag.

Goren-Bar, D., Kuflik, T., Lev, D. & Shoval, P. (2001). Automating personal categorization using artificial neural networks. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 188–198). Berlin: Springer.

Greiner, R., Grove, A. J. & Schuurmans, D. (1997). Learning Bayesian nets that perform well. In D. Geiger & P. P. Shenoy (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference* (S. 198–207). San Francisco: Morgan Kaufmann.

Großmann-Hutter, B., Jameson, A. & Wittig, F. (1999). Learning Bayesian networks with hidden variables for user modeling. In *Proceedings of the IJCAI 99 Workshop "Learning About Users"* (S. 29–34). Stockholm.

Heckerman, D. (1995). *A tutorial on learning with Bayesian networks* (Tech. Rep. Nr. MSR-TR-95-06). Microsoft Research. (Revised November 1996)

Heckerman, D. (1998). A tutorial on learning with Bayesian networks. In M. I. Jordan (Hrsg.), *Learning in Graphical Models.* Cambridge, MA: MIT Press.

Heckerman, D., Chickering, D. M., Meek, C., Rounthwaite, R. & Kadie, C. (2000). Dependency networks for collaborative filtering and data visualization. In C. Boutilier & M. Goldszmidt (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the 16th Conference* (S. 264–273). San Francisco, CA: Morgan Kaufmann.

Heckerman, D., Geiger, D. & Chickering, D. M. (1994). Learning Bayesian networks: The combination of knowledge and statistical data. In R. Lopez de Mantaras & D. Poole (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Tenth Conference* (S. 293–301). San Francisco: Morgan Kaufmann.

Heckerman, D., Geiger, D. & Chickering, D. M. (1995). Learning Bayesian networks: The cobination of knowledge and statistical data. *Machine Learning, 20*, 197–243.

Herlocker, J. L., Konstan, J. A. & Riedl, J. (2000). Explaining collaborative filtering recommendations. In *Proceedings of the 2000 Conference on Computer-Supported Cooperative Work.*

Hofmann, R. (2000). *Lernen der Struktur nichtlinearer Abhängigkeiten mit graphischen Modellen.* Dissertation, Technische Universität München.

Höppner, S. (2001). An adaptive user-interface-agent modeling communication ability. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 127–136). Berlin: Springer.

Horvitz, E., Breese, J., Heckerman, D., Hovel, D. & Rommelse, K. (1998). The Lumière project: Bayesian user modeling for inferring the goals and needs of software users. In G. F. Cooper & S. Moral (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Fourteenth Conference* (S. 256–265). San Francisco: Morgan Kaufmann.

Horvitz, E., Jacobs, A. & Hovel, D. (1999). Attention-sensitive alerting. In K. B. Laskey & H. Prade (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Fifteenth Conference* (S. 305–313). San Francisco: Morgan Kaufmann.

Horvitz, E., Koch, P., Kadie, C. M. & Jacobs, A. (2002). Coordinate: Probabilistic forecasting of presence and availability. In A. Darwiche & N. Friedman (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Eighteenth Conference* (S. 224-233). San Francisco: Morgan Kaufmann.

Horvitz, E. & Paek, T. (1999). A computational architecture for conversation. In J. Kay (Hrsg.), *UM99, User Modeling: Proceedings of the Seventh International Conference* (S. 201–210). Wien: Springer.

Horvitz, E. & Paek, T. (2001). Harnessing models of users' goals to mediate clarification dialog in spoken language systems. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 3–13). Berlin: Springer.

HUGIN Expert A/S. (2000). *HUGIN API Manual.* Aalborg, Dänemark. (http://www.hugin.com)

Jacobs, N. & Blockeel, H. (2001). The learning shell: Automated macro construction. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 34–43). Berlin: Springer.

Jameson, A. (1996). Numerical uncertainty management in user and student modeling: An overview of systems and issues. *User Modeling and User-Adapted Interaction, 5,* 193–251.

Jameson, A. (2002). Adaptive interfaces and agents. In J. A. Jacko & A. Sears (Hrsg.), *Handbook of Human-Computer Interaction in Interactive Systems.* Mahwah, NJ: Erlbaum. (Im Druck)

Jameson, A., Großmann-Hutter, B., March, L., Rummer, R., Bohnenberger, T. & Wittig, F. (2001). When actions have consequences: Empirically based decision making for intelligent user interfaces. *Knowledge-Based Systems, 14,* 75–92.

Jameson, A., Konstan, J. & Riedl, J. (2002). *AI techniques for personalized recommendation.* Tutorial Notes AAAI 2002. (Available from http://www.dfki.de/∼jameson/)

Jameson, A., Wahlster, W., Bohnenberger, T., Brandherm, B., Großmann-Hutter, B. & Wittig, F. (2001). READY: Lernen, Modellierung und Entscheidung für situierte Interaktion. In J. Siekmann (Hrsg.), *Fortsetzungsantrag Sonderforschungsbereich "Ressourcenadaptive kognitive Prozesse" (SFB 378).* Saarbrücken: Universität des Saarlandes.

Jameson, A. & Wittig, F. (2001). Leveraging data about users in general in the learning of individual user models. In B. Nebel (Hrsg.), *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence* (S. 1185–1192). San Francisco, CA: Morgan Kaufmann.

Jensen, F., Jensen, F. V. & Dittmer, S. L. (1994). From influence diagrams to junction trees. In R. Lopez de Mantaras & D. Poole (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Tenth Conference* (S. 367–373). San Francisco: Morgan Kaufmann.

Jensen, F. V. (1996). *An Introduction to Bayesian Networks.* New York: Springer.

Jensen, F. V. (2001). *Bayesian Networks and Decision Graphs.* New York: Springer.

Jordan, M. I. (Hrsg.). (1998). *Learning in Graphical Models.* MIT Press.

Kahneman, D., Slovic, P. & Tversky, A. (1982). *Judgement under Uncertainty: Heuristics and Biases.* Cambridge: Cambridge University Press.

Kay, J. & McCreath, E. (2001). Automatic induction of rules for e-mail classification. In R. Schäfer, M. E. Müller & S. A. Macskassy (Hrsg.), *Proceedings of the UM2001-Workshop on "Machine Learning for User Modeling"* (S. 59–66). Sonthofen.

Kiefer, J. (2002). *Auswirkungen von Ablenkung durch gehörte Sprache und eigene Handlungen auf die Sprachproduktion.* Diplomarbeit, Fachbereich Psychologie, Universität des Saarlandes, Saarbrücken.

Kjærulff, U. (1995). dHugin: A computational system for dynamic time-sliced Bayesian networks. *International Journal of Forecasting, 11,* 89–111.

Kobsa, A. (2001a). Generic user modeling systems. *User Modeling and User-Adapted Interaction, 11*, 49–63.

Kobsa, A. (2001b). Tailoring privacy to users' needs. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 303–313). Berlin: Springer.

Kobsa, A., Koenemann, J. & Pohl, W. (2001). Personalized hypermedia presentation techniques for improving online customer relationships. *The Knowledge Engineering Review*.

Koller, D. & Pfeffer, A. (1997). Object-oriented Bayesian networks. In D. Geiger & P. P. Shenoy (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference* (S. 302–313). San Francisco: Morgan Kaufmann.

Koller, D. & Pfeffer, A. (1998). Probabilistic frame-based systems. In *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-98)* (S. 580–587). Madison, Wisconsin.

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R. & Riedl, J. (1997). GroupLens: Applying collaborative filtering to Usenet news. *Communications of the ACM, 40*(3), 77–87.

Koychev, I. (2001). Learning about the user in the presence of hidden context. In R. Schäfer, M. E. Müller & S. A. Macskassy (Hrsg.), *Proceedings of the UM2001-Workshop on "Machine Learning for User Modeling"* (S. 49–58). Sonthofen.

Kozlov, A. V. & Koller, D. (1997). Nonuniform dynamic discretization in hybrid networks. In D. Geiger & P. P. Shenoy (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the 13th Conference* (S. 314–325). San Francisco: Morgan Kaufmann.

Lam, W. & Bacchus, F. (1993). Learning Bayesian belief networks: An approach based on the MDL principle. *Computational Intelligence, 10*, 269–293.

Lam, W. & Bacchus, F. (1994). Using new data to refine a Bayesian network. In R. Lopez de Mantaras & D. Poole (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Tenth Conference* (S. 383–390). San Francisco: Morgan Kaufmann.

Langley, P. (1997). Machine learning for adaptive interfaces. In G. Brewka, C. Habel & B. Nebel (Hrsg.), *KI-97: Advances in Artificial Intelligence* (S. 53–62). Berlin: Springer.

Langley, P. (1999). User modeling in adaptive interfaces. In J. Kay (Hrsg.), *UM99, User Modeling: Proceedings of the Seventh International Conference.* Wien: Springer.

Langseth, H. & Bangsø, O. (2000). *Parameter learning in object oriented Bayesian networks* (Tech. Rep. Nr. CIT-87.2-00-HLOB-001). Department of Computer Science.

Laskey, K. B. & Mahoney, S. M. (1997). Network fragments: Representing knowledge for constructing probabilistic models. In D. Geiger & P. P. Shenoy (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference* (S. 334–341). San Francisco: Morgan Kaufmann.

Lau, T. & Horvitz, E. (1999). Patterns of search: Analyzing and modeling Web query dynamics. In J. Kay (Hrsg.), *UM99, User Modeling: Proceedings of the Seventh International Conference* (S. 119–128). Wien: Springer.

Lauritzen, S. L. & Spiegelhalter, D. J. (1988). Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Scociety, Series B (Methodological), 50*(2), 157–224.

Madigan, D. & Raftery, A. E. (1994). Model selection and accounting for model uncertainty in graphical models using Occam's window. *Journal of the American Statistical Association, 89*, 1535–1546.

Madigan, D. & York, J. (1995). Bayesian graphical models for discrete data. *International Statistical Review, 63*, 215–232.

Mahoney, S. M. & Laskey, K. B. (1996). Network engineering for complex belief networks. In E. Horvitz & F. V. Jensen (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Twelfth Conference* (S. 389–396). San Francisco: Morgan Kaufmann.

Mahoney, S. M. & Laskey, K. B. (1998). Constructing situation specific belief networks. In G. F. Cooper & S. Morales (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Fourteenth Conference* (S. 370-378). San Francisco: Morgan Kaufmann.

March, L. (1999). *Ressourcenadaptive Instruktionen in einem Hotline-Szenario*. Diplomarbeit, Fachbereich Psychologie, Universität des Saarlandes, Saarbrücken.

Mitchell, T., Caruana, R., Freitag, D., McDermott, J. & Zabowski, D. (1994). Experience with a learning personal assistant. *Communications of the ACM, 37*(7), 81–91.

Mitchell, T. M. (1997). *Machine Learning.* Boston: McGraw-Hill.

Moore, A. & Lee, M. S. (1998). Cached sufficient statistics for efficient machine learning with large datasets. *Journal of Artificial Intelligence Research, 8*, 67–91.

Morgan, M. G. & Henrion, M. (1990). *Uncertainty, a Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis.* Cambridge: Cambridge University Press.

Muggleton, S. (1991). Inductive logic programming. *New Generation Computing, 8*, 295–318.

Müller, C. (2001). *Symptome von Zeitdruck und kognitiver Belastung in gesprochener Sprache: eine experimentelle Untersuchung.* Diplomarbeit, Fachrichtung Computerlinguistik, Universität des Saarlandes, Saarbrücken.

Müller, C., Großmann-Hutter, B., Jameson, A., Rummer, R. & Wittig, F. (2001). Recognizing time pressure and cognitive load on the basis of speech: An experimental study. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference.* Berlin: Springer.

Müller, M. E. (2002). *Inducing Coneptual User Models.* Dissertation, Fachbereich Sprach- und Literaturwissenschaften, Universität Osnabrück.

Murphy, K. & Mian, S. (1999). *Modelling gene expression data using dynamic Bayesian networks* (Tech. Rep.). Computer Science Division, University of California.

Murphy, K. P. (2001). *Learning Bayes net structure from sparse data sets* (Tech. Rep.). Computer Science Division, UC Berkeley.

Neapolitan, R. E. (1990). *Probabilistic Reasoning in Expert Systems: Theory and Algorithms.* New York: Wiley.

Nicholson, A., Boneh, T., Wilkin, T., Stacey, K., Sonenberg, L. & Steinle, V. (2001). A case study in knowledge discovery and elicitation in an intelligent tutoring application. In J. Breese & D. Koller (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Seventeenth Conference* (S. 386–394). San Francisco: Morgan Kaufmann.

Nicholson, A. E. (1996). Fall diagnosis using dynamic belief networks. In N. Foo & R. Goebel (Hrsg.), *Proceedings of the Fourth Rim International Conference on Artificial Intelligence (PRICAI-96)* (Bd. 1114, S. 206–217). Berlin: Springer.

Nicholson, A. E. & Brady, J. M. (1994). Dynamic belief networks for discrete monitoring. *IEEE Transactions on Systems, Man, and Cybernetics*, *24*, 1593–1610.

Olesen, K. G., Lauritzen, S. L. & Jensen, F. V. (1992). aHUGIN: A system creating adaptive causal probabilistic networks. In D. Dubois, M. P. Wellman, B. D'Ambrosio & P. Smets (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Eighth Conference* (S. 223–229). San Mateo: Morgan Kaufmann.

Ortiz, L. E. & Kaelbling, L. P. (1999). Accelerating EM: An empirical study. In K. B. Laskey & H. Prade (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Fifteenth Conference* (S. 512–521). San Francisco: Morgan Kaufmann.

Orwant, J. (1995). Heterogeneous learning in the Doppelgänger user modeling system. *User Modeling and User-Adapted Interaction*, *4*(2), 107–130.

Paek, T. & Horvitz, E. (2000). Conversation as action under uncertainty. In C. Boutilier & M. Goldszmidt (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Sixteenth Conference.* San Francisco: Morgan Kaufmann.

Paliouras, G., Karkaletsis, V., Papatheodorou, C. & Spyropoulos, C. D. (1999). Exploiting learning techniques for the acquisition of user stereotypes and communities. In J. Kay (Hrsg.), *UM99, User Modeling: Proceedings of the Seventh International Conference* (S. 169–178). Wien: Springer.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* San Mateo, CA: Morgan Kaufmann.

Pfeffer, A., Koller, D., Milch, B. & Takusagawa, K. T. (1999). SPOOK: A system for probabilistic object-oriented knowledge representation. In K. B. Laskey & S. M. Mahoney (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Fifteenth Conference* (S. 541–550). San Francisco: Morgan Kaufmann.

Pohl, W. & Nick, A. (1999). Machine learning and knowledge-based user modeling in the LaboUr approach. In J. Kay (Hrsg.), *UM99, User Modeling: Proceedings of the Seventh International Conference* (S. 179–188). Wien: Springer.

Pohl, W., Schwab, I. & Koychev, I. (1999). Learning about the user: A general approach and its application. In *Proceedings of the IJCAI 99 Workshop "Learning About Users"*. Stockholm.

Press, W. H. (1992). *Numerical Recipes in C.* Cambridge, England: Cambridge University Press.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning, 1*(1), 81–106.

Quinlan, J. R. (1993). *C4. 5: Programs for Machine Learning.* San Mateo, CA: Morgan Kaufmann.

Rich, E. (1979). User modeling via stereotypes. *Cognitive Science, 3*, 329–354.

Rich, E. (1989). Stereotypes and user modeling. In A. Kobsa & W. Wahlster (Hrsg.), *User Models in Dialog Systems* (S. 35–51). Berlin: Springer.

Robinson, R. W. (1977). Counting unlabeled acyclic digraphs. In C. H. C. Little (Hrsg.), *Lecture Notes in Mathematics 622: Combinatorial Mathematics V.* Springer.

Roure, J. & Sangüesa, R. (1999). *Incremental methods for Bayesian network learning* (Tech. Rep. Nr. LSI-99-42-R). Software Department at the Technical University of Catalonia.

Russell, S., Binder, J., Koller, D. & Kanazawa, K. (1995). Local learning in probabilistic networks with hidden variables. In C. S. Mellish (Hrsg.), *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (S. 1146–1152). San Mateo, CA: Morgan Kaufmann.

Russell, S. J. & Norvig, P. (1995). *Artificial Intelligence: A Modern Approach.* Englewood Cliffs, NJ: Prentice-Hall.

Schäfer, R. (1998). *Benutzermodellierung mit dynamischen Bayes'schen Netzen als Grundlage adaptiver Dialogsysteme.* Dissertation, Lehrstuhl Wahlster, Fachrichtung Informatik, Universität des Saarlandes, Saarbrücken.

Schäfer, R. & Weyrath, T. (1997). Assessing temporally variable user properties with dynamic Bayesian networks. In A. Jameson, C. Paris & C. Tasso (Hrsg.), *User modeling: Proceedings of the Sixth International Conference, UM97* (S. 377–388). Wien: Springer.

Schwab, I. & Kobsa, A. (2002). Adaptivity through unobstrusive learning. *Künstliche Intelligenz, 16*(3), 5–9.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals in Statistics, 6*, 461-464.

Segal, R. B. & Kephart, J. O. (2000). Incremental learning in SwiftFile. In P. Langley (Hrsg.), *Machine Learning: Proceedings of the 2000 International Conference.* San Francisco: Morgan Kaufmann.

Semeraro, G., Ferilli, S., Fanizzi, N. & Abbattist, F. (2001). Learning interaction models in a digital library service. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 44–53). Berlin: Springer.

Shachter, R. D. (1986). Evaluating influence diagrams. *Operations Research, 34*, 871–882.

Spiegelhalter, D. J. & Lauritzen, S. L. (1990). Sequential updating of conditional probabilities on directed graphical structures. *Networks, 20*, 579–605.

Spirtes, P., Glymour, C. & Scheines, R. (1990). Causality from probability. In *Proceedings of Advanced Computing for the Social Sciences.* Williamsburgh, VA.

Spirtes, P., Glymour, C. & Scheines, R. (1991). An algoprithm for fast recovery of sparse causal graphs. *Social Science Computer Review, 9*, 62–72.

Steck, H. (2000). On the use of skeletons when learning in bayesian networks. In C. Boutilier & M. Goldszmidt (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the 16th Conference* (S. 558–565). San Francisco, CA: Morgan Kaufmann.

Suzuki, J. (1993). A construction of Bayesian networks from databases based on an MDL principle. In D. Heckerman & A. Mamdani (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Ninth Conference* (S. 266–273). San Mateo: Morgan Kaufmann.

Teach, R. L. & Shortliffe, E. H. (1984). An analysis of physicians' attitudes. In B. G. Buchanan & E. H. Shortliffe (Hrsg.), *Rule-based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project* (S. 635–652). Reading, MA: Addison-Wesley.

Tong, S. & Koller, D. (2000). Active learning for parameter estimation in Bayesian networks. In *Proceedings of the 2000 Conference on Neural Information Processing Systems.*

van der Gaag, L. C., Renoij, S., Witteman, C. L. M. & Aleman, B. M. P. (1999). How to elicit many probabilities. In K. B. Laskey & S. M. Mahoney (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Fifteenth Conference* (S. 647–654). San Francisco: Morgan Kaufmann.

von Winterfeldt, D. & Edwards, W. (1986). *Decision analysis and behavioral research.* Cambridge, England: Cambridge University Press.

Wahlster, W. (1981). *Natürlichsprachliche Argumentation in Dialogsystemen.* Informatik-Fachberichte 48, Berlin: Springer.

Wahlster, W. (Hrsg.). (2000). *Verbmobil: Foundations of Speech-to-Speech Translation.* Berlin: Springer.

Wahlster, W. & Kobsa, A. (1989). User models in dialog systems. In A. Kobsa & W. Wahlster (Hrsg.), *User Models in Dialog Systems* (S. 4–34). Berlin: Springer.

Waszkiewicz, P., Cunningham, P. & Byrne, C. (1999). Case-based user profiling in a personal travel assistant. In J. Kay (Hrsg.), *UM99, User Modeling: Proceedings of the Seventh International Conference* (S. 323–325). Wien: Springer.

Webb, G., Pazzani, M. J. & Billsus, D. (2001). Machine learning for user modeling. *User Modeling and User-Adapted Interaction, 11*, 19–29.

Weibelzahl, S. (2001). Evaluation of adaptive systems. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 292–294). Berlin: Springer.

Weibelzahl, S. & Weber, G. (2002). Advantages, oppotunities and limits of empirical evaluations: Evaluating adaptive systems. *Künstliche Intelligenz, 16*(3), 17–20.

Wellman, M. P. (1990). Fundamental concepts of qualitative probabilistic networks. *Artificial Intelligence, 44*, 257–303.

Widmer, G. & Kubat, M. (1996). Learning in the presence of concept drift and hidden contexts. *Machine Learning, 23*(1), 69–101.

Wittig, F. (1999). Learning Bayesian networks with hidden variables for user modeling. In J. Kay (Hrsg.), *UM99, User Modeling: Proceedings of the Seventh International Conference.* Wien: Springer.

Wittig, F. (2001a). Some issues in the learning of accurate, interpretable user models from sparse data. In R. Schäfer, M. E. Müller & S. A. Macskassy (Hrsg.), *Proceedings of the UM2001-Workshop on "Machine Learning for User Modeling"* (S. 11–21). Sonthofen.

Wittig, F. (2001b). Empirisch basierte Benutzermodellierung mit Bayes'schen Netzen: Strukturelle Aspekte. In N. Henze (Hrsg.), *ABIS2001: GI-Workshop "Adaptivität und Benutzermodellierung"*. Dortmund.

Wittig, F. (2002). Zum maschinellen Lernen in benutzeradaptiven Systemen am Beispiel Bayes'scher Netze. In N. Henze (Hrsg.), *ABIS2002: GI-Workshop "Adaptivität und Benutzermodellierung"*. Hannover.

Wittig, F. & Jameson, A. (2000). Exploiting qualitative knowledge in the learning of conditional probabilities of Bayesian networks. In C. Boutilier & M. Goldszmidt (Hrsg.), *Uncertainty in Artificial Intelligence: Proceedings of the Sixteenth Conference* (S. 644–652). San Francisco: Morgan Kaufmann.

Zadeh, L. A. (1996). Fuzzy logic = computing with words. *IEEE Transactions on Fuzzy Systems, 4*, 103–111.

Zukerman, I. (2001). An integrated approach for generating arguments and rebuttals and understanding rejoinders. In J. Vassileva, P. Gmytrasiewicz & M. Bauer (Hrsg.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (S. 84–94). Berlin: Springer.

Zukerman, I. & Albrecht, D. W. (2001). Predictive statistical models for user modeling. *User Modeling and User-Adapted Interaction, 11*, 5–18.

Index

C

D

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

V

W

Z