Diplomarbeit

Ein Informationsassistent zum kooperativen Verstehen von Textdokumenten

Ludger van Elst

Betreuer:
Priv. Doz. Dr. Franz Schmalhofer

Verantwortlicher Professor:
Prof. Dr. Andreas Dengel

Universität Kaiserslautern

14. November 1998

Erklärung

Hiermit erkläre ich, daß ich diese Arbeit selber verfaßt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Kaiserslautern, den 14. November 1998

Ludger van Elst

Das Lichtlein der heiligen Neugier kann dauernd verlöschen.
Albert Einstein[1]

meinen Eltern

[1]nach: *Zeiten des Staunens*, herausgegeben von Harald Schützeichel, Herder, Freiburg

Danksagung

Diese Arbeit ist im Forschungsbereich Informationsmanagement und Dokumentanalyse des Deutschen Forschungszentrums für Künstliche Intelligenz entstanden. Dem Leiter des Bereiches, Herrn Prof. Dengel, danke ich herzlich dafür, daß er die Infrastruktur zur Verfügung gestellt und gegenüber dem Fachbereich Informatik die Verantwortung für die Durchführung der Arbeit übernommen hat.
Franz Schmalhofer danke ich nicht nur für die kompetente Betreuung dieser Diplomarbeit, sondern ganz besonders auch für die unzähligen fachlichen Diskussionen und persönlichen Gespräche in den vergangenen fünf Jahren der Zusammenarbeit. Ohne seine Ideen und Anregungen wäre diese Arbeit sicher nicht in der nun vorliegenden Form entstanden.
Ich danke Alexandra Bachran und Rainer Koster, die durch ihre wertvollen Kommentare zu verschiedenen Entwürfen sehr zum Gelingen dieser Arbeit beigetragen haben.
Einem Stipendium der Bischöflichen Studienförderung Cusanuswerk verdanke ich es, daß mir ein Studium in der angestrebten Breite und Tiefe tatsächlich möglich war.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Zusammenfassung

Eine der wesentlichen Aufgaben sowohl im Internet als auch in betrieblichen Anwendungen des Wissensmanagements auf der Basis von Intranet-Technologien ist es, daß die *richtigen Personen* zur *richtigen Zeit* die *richtigen Dokumente* erhalten. Will man dieses Problem mit Informationstechnologien bearbeiten, so ist insbesondere genauer zu spezifizieren, was "richtig" in dem jeweiligen Kontext bedeutet.
Diese Arbeit beschäftigt sich in erster Linie mit der Frage, welche *Dokumente* für einen Informationskonsumenten die *richtigen* sind. Die Frage der Relevanz von Dokumenten hinsichtlich spezifischer Anfragen von Informationskonsumenten wird im allgemeinen als Problem des *information retrieval* betrachtet. Im Rahmen dieser Arbeit wird diese Betrachtungsweise etwas geweitet, indem nicht nur die Anfrage eines Benutzers und die Menge der Dokumente berücksichtigt wird, sondern die gesamte *Informationslandschaft*. Für eine in Wissensmanagement-Anwendungen typische Informationslandschaft mit ihren verschiedenen Akteuren und Informationsassistenten wird ein spezieller Assistent (*Inhaltsassistent*) entworfen, der dadurch ein besseres *information retrieval* ermöglichen soll, daß er zum einen Repräsentationen der Dokumente erstellt, die kompatibel zu denen der Benutzer sind und es zum anderen ermöglicht, die *retrieval*-Ergebnisse speziell auf die Sichtweise und Anforderung des Informationskonsumenten anzupassen.
Dazu werden insbesondere zwei neue Verfahren spezifiziert, die einerseits die Erzeugung formalerer Repräsentationen aus Textdokumenten ermöglichen (*CL-Propositionalisierung*), andererseits Suchergebnisse in einen durch den Benutzer vorgegebenen Kontext integrieren. Dadurch wird eine *Individualisierung* der Dokumentensuche erreicht.
Insgesamt wird durch die Betrachtung der gesamten Informationslandschaft nicht nur das *information retrieval* verbessert, sondern auch die Möglichkeit, ein solches Informationssystem als *Wissensmedium* und *Kommunikationsmittel* zwischen verschiedenen Benutzern zu nutzen. Für diese Aufgabe scheint die Fähigkeit, sowohl mit stark strukturierten Repräsentationen als auch mit eher an der Textoberfläche orientierten Darstellungen umgehen zu können, sehr wichtig zu sein. Der hier dargestellte Inhaltsassistent erzeugt und benutzt daher eine Reihe von mehr oder weniger formalen Dokumentrepräsentationen und gewährleistet somit die geforderte Flexibilität, wie sie für die vielfältigen Aufgaben im Informations- und Wissensmanagement notwendig ist.

Kapitel 1

Einleitung

> "Intelligent information retrieval is the next major challenge for the computational sciences."
> *Herb Simon's Proclamation* [1]

Soll diese strategische Herausforderung auf dem Gebiet der Informatik wissenschaftlich bearbeitet werden, müssen dabei sowohl konzeptuelle Ansätze als auch konkrete algorithmische Lösungen und Implementierungstechniken angesprochen werden. In dieser Arbeit wird dies für das intelligente *information retrieval* im Rahmen einer Problemstellung des Wissensmanagements durchgeführt.

1.1 Motivation

Das Gebiet der "inhaltlichen Suche in Texten" (*information* oder besser *text retrieval*) hat seit seinen Anfängen (z.B. Suche in Literaturdatenbanken in den 50er Jahren) stark an Bedeutung gewonnen (vgl. dazu [Salton 68], [van Rijsbergen 79], [Salton et al. 83], [Frakes et al. 92]).

Grundlegend für die wissenschaftlichen Arbeiten in diesem Bereich waren und sind vor allem drei Begriffe:

- *Ground Truth*: Mit diesem Begriff wird die Menge der Dokumente in der Datenbasis bezeichnet, die bezüglich einer Suche tatsächlich relevant

[1] Das Zitat von Professor Herbert Simon – Träger des Nobelpreises für Wirtschaftswissenschaften für seine Forschungen zur *bounded rationality* – ist aus einem Vortrag, der von Professor Jaime Carbonell am Deutschen Forschungszentrum für Künstliche Intelligenz am 1. April 1998 gehalten wurde [Carbonell 98].

sind. Ziel eines jeden *information retrieval*-Systems ist es also, genau diese Menge zu finden. Im allgemeinen weicht natürlich die von einem System auf eine Anfrage erzeugte Menge *Found* von der *Ground Truth* ab.

- *Precision*: Die *precision* p gibt den Anteil der relevanten Dokumente an den tatsächlich gefundenen Dokumenten wieder.

$$p = \frac{|Ground\ Truth \cap Found|}{|Found|} \tag{1.1}$$

  Ein System mit $p = 1$ liefert also auf eine Anfrage *nur* wirklich relevante Dokumente zurück.

- *Recall*: Der *recall*-Wert r gibt an, welcher Anteil der relevanten Dokumente tatsächlich gefunden wird.

$$r = \frac{|Ground\ Truth \cap Found|}{|Ground\ Truth|} \tag{1.2}$$

Die konkrete Bestimmung der *precision*- und *recall*-Werte eines Systems ist im allgemeinen sehr schwierig. Für die *precision* muß das Ergebnis einer Anfrage – eine, gemessen an der Anzahl vorhandener Dokumente, recht kleine Menge – evaluiert werden, um festzustellen, welche gefundenen Dokumente tatsächlich relevant sind. Noch problematischer ist der *recall*. Für diesen Wert wird die Mächtigkeit der *Ground Truth* benötigt, so daß eigentlich die Menge *aller* Dokumente bewertet werden müßte. Dies ist jedoch aus praktischen Gründen unmöglich.[2] Daher versucht man, Näherungswerte für $|Ground\ Truth|$ zu erhalten, indem man z.B. Stichproben betrachtet.

Schon sehr früh wurden *information retrieval*-Systeme über ihre *precision*- und *recall*-Werte miteinander verglichen. Dabei kann man durch theoretische Überlegungen wie auch empirisch leicht feststellen, daß es einen *trade-off* zwischen diesen beiden Werten gibt, d.h. ein höherer *recall* wird im allgemeinen mit einer schlechteren *precision* erkauft und umgekehrt.[3]
Seit 1992 gibt es die *Text REtrieval Conference* (TREC)[4], in der *text retrieval*-Systeme systematisch empirisch miteinander verglichen werden.

[2] Wäre dies praktisch möglich, hätte man ein (perfektes) Verfahren mit $p = r = 1$. Man wertet einfach das Ergebnis aus und bekommt so *nur* relevante und *alle* relevanten Dokumente.

[3] Im Extremfall liefert man beispielsweise auf eine Anfrage *alle* Dokumente als relevant zurück. Dann hat man einen *recall* von Eins, während die *precision* nahezu Null ist.

[4] http://trec.gist.gov

Neuere Forschungen in diesem Bereich betrachten neben dem eigentlichen *retrieval* insbesondere auch Fragestellungen bezüglich redundanter, veralteter oder unzuverlässiger Informationen, verschiedener Sprachen und Typen von Informationen bis hin zu rechtlichen Problemen. Dadurch ergeben sich Querbezüge zu den verschiedensten Forschungsbereichen der Informatik wie z.B. Sprach- und Bilderkennung, automatisches Verstehen von Dokumenten, Dokumentgenerierung und automatisches Übersetzen [Carbonell 98].

Obwohl gerade mit der Ausbreitung des *world wide web* in den 90er Jahren einige Systeme – und damit auch Konzepte – den Schritt von Forschungsprototypen in die Praxis geschafft haben (wie z.B. die Suchmaschinen MetaCrawler und Excite), basieren die meisten kommerziellen Systeme immer noch auf booleschen Anfragesprachen[5], die nur überprüfen, ob ein Wort in einem Dokument vorkommt oder nicht.

Gleichzeitig zeigt die rapide anwachsende Zahl der Dokumente im WWW aber auch weitere Probleme solcher "offenen Welten"[6] auf: Es ist unmöglich, zu einem bestimmten Zeitpunkt einen "Weltzustand" aufzuzeichnen, d.h. eine vollständige oder korrekte Repräsentation dieser Welt zu haben. Selbst Altavista als eine der leistungsfähigsten Suchmaschinen kann täglich nur etwa sechs Millionen der momentan ca. 320 Millionen Seiten im WWW indizieren. Keine der heute im Einsatz befindlichen Suchmaschinen hat mehr als ein Drittel aller vorhandenen Web-Seiten überhaupt besucht. Suchmaschinen, bei denen die Einpflege neuer Seiten in den Index sehr aufwendig sind, wie zum Beispiel Yahoo! (hier erfolgen manuelle Kategorisierungen), machen aus der Not eine Tugend und nehmen nur Seiten auf, die gewissen Qualitätsstandards genügen. Es zeichnet sich also ab, daß die "traditionellen" Techniken, die das gesamte *world wide web* als *eine* große Wissensbasis ansehen, für die es einen zentralen Index zu erstellen gilt, den Anforderungen der nächsten Zeit kaum gewachsen sein werden. Hier müssen neue Konzepte und Verfahren entwickelt werden. Dazu scheinen *agenten-orientierte Ansätze*, wie sie in letzter Zeit in vielen Bereichen der Informatik verfolgt werden [Huhns et al. 97], besonders erfolgversprechend zu sein, da sie im Kern das Potential haben, verteilte und soziale Aspekte besser zu berücksichtigen als traditionellere Methodiken.

Weiterentwicklungen auf dem Gebiet des *information retrieval* sollten dabei aber nicht nur auf dem Hintergrund des (öffentlichen) *world wide web* erfolgen, sondern auch die Erfordernisse von Intranets mit einbeziehen, wie sie von

[5]Schon zu Beginn der 60er Jahre wurde darauf hingewiesen, daß boolesche Anfragesprachen für das *text retrieval* wenig geeignet sind [Verhoeff et al. 61]. Tatsächlich erzielen diese Verfahren im Vergleich die schlechteste *retrieval*-Qualität [Salton et al. 83].

[6]Solche offenen Welten zeichnen sich durch einen hohen Grad an Parallelität aus. Es gibt keine zentrale Koordination oder verteilte Verantwortungskonzepte.

vielen Firmen betrieben werden. Die Entwicklung im Bereich der Intranets in den letzten Jahren ist ähnlich rasant wie beim Internet.

Wissen wird im modernen Informationszeitalter als eines der wichtigsten Güter von Betrieben betrachtet, so daß dem *Wissensmanagement* immer mehr Bedeutung zukommt (vgl. [Nonaka et al. 95], [Tschaitschian et al. 97]). Solches Wissen läßt sich entlang mehrerer Dimensionen beschreiben: So kann es explizites oder implizites Wissen sein, es kann individuell oder organisational sein usw. Insbesondere stehen beim Wissensmanagement die einzelnen Personen und Gruppen (Informationsanbieter, -verteiler, -konsumenten etc.) mit ihren verschiedenen Sichtweisen, Beziehungen und Zielen im Zentrum der Betrachtung.[7]

Immer mehr Wissen ist in elektronischen Archiven (z.B. auf der Basis von Intranet-Technologien) gespeichert, die für das betriebliche Wissensmanagement genutzt werden können (*organizational* oder *corporate memories*, vgl. [Kühn et al. 97]). Allerdings stellen immer mehr Mitarbeiter, die an solche *corporate memories* angeschlossen sind, fest, daß sie wichtige Informationen oft nicht bekommen, während sie mit unwichtiger Information überhäuft werden. Intelligente Werkzeuge, die den Benutzer beim Zugriff auf Informationen von Internet und Intranets unterstützen, werden daher immer wichtiger.

Eine der zentralen Aufgaben solcher Werkzeuge für das Informations- und Wissensmanagement läßt sich folgendermaßen formulieren:

> "Bringe die *richtigen* Dokumente zur *richtigen* Zeit an die *richtigen* Personen."

Diese Formulierung einer Kernaufgabe des Wissensmanagements umfaßt sowohl die Informationsverteilung von Distributoren an Informationskonsumenten (*push-oriented approach*) als auch die Suche (Nachfrage) seitens der Informationskonsumenten (*pull-oriented approach*).

Um diese Aufgabenstellung angemessen zu bearbeiten, muß insbesondere geklärt werden, was "richtig" jeweils bedeutet. Innerhalb einer *real world*-Anwendung kann hier nur ausschlaggebend sein, wie ein Benutzer in seiner Rolle als Informationskonsument ein ihm zu einem festen Zeitpunkt zugestelltes Dokument bezüglich seiner Relevanz für seine momentanen Zwecke beurteilt.[8] Ein System, das den Benutzer bei der Informationssuche unterstützt,

[7] In einer mehr mathematischen Sichtweise kann man einige Aspekte des Wissensmanagement durchaus auch im Rahmen einer spieltheoretischen Betrachtung als N-Personenspiele mit Kooperations- und Kündigungsmöglichkeiten usw. beschreiben.

[8] Somit ist die oben beschriebene *ground truth* – die Menge der "richtigen Dokumente" – eher eine abstrakte oder artifizielle Einheit.

sollte daher eine solche Relevanzbeurteilung möglichst gut annähern können. Somit besteht im einfachsten Fall so ein System im Kern aus einer Funktion R, die die Relevanz eines Dokumentes d für einen Benutzer p zu einem bestimmten Zeitpunkt t modelliert. Es zeigt sich also, daß die zu Beginn dargestellte Problemstellung des *information retrieval* auch für die Bearbeitung von Fragestellungen Wissensmanagements mit Informationstechnologien von zentraler Bedeutung ist.

1.2 Überblick

In dieser Arbeit soll ein kooperativer Informationsassistent entwickelt werden, der als intelligentes Werkzeug den Benutzer beim Zugriff auf Informationen in Intranets und im Intranet unterstützen soll. Dabei soll die Relevanz eines Dokumentes für einen Benutzer sowohl vom Inhalt des Dokumentes als auch vom Ziel des jeweiligen Benutzers abhängig sein. Es wird daher wesentlicher Bestandteil des Konzeptes sein, daß der Informationsassistent Repräsentationen aufbaut, die mit denen der Benutzer kompatibel sind.

In einem ersten Schritt wird das Problem anhand eines konkreten Szenarios ("Student sucht Studienplatz") genauer analysiert (Kapitel 2). In Kapitel 3 wird die Struktur der vorgeschlagenen Lösung auf einer recht allgemeinen und abstrakten Ebene charakterisiert. Anschließend werden die benötigten Repräsentationen und Prozesse detailliert ausgearbeitet (Kapitel 4). Die Implementierung von Modulen, die benutzt werden können, um einen konkreten Informationsassistenten für eine bestimmte Anwendung zu konfigurieren, wird in Kapitel 5 beschrieben. In der Diskussion (Kapitel 6) wird die vorgeschlagene Lösung noch einmal in einen umfassenderen Rahmen eingeordnet, um weitere Anwendungs- und Entwicklungsmöglichkeiten aufzuzeigen.

Kapitel 2

Problembeschreibung

In diesem Abschnitt wird eine prototypische Anwendung des Informations- und Wissensmanagements aus der realen Welt dargestellt. Diese wird verwendet, um die für ein Assistenzsystem relevanten Entitäten sowie einige grundsätzliche, sich aus der Struktur der Aufgabenstellung ergebende Probleme zu identifizieren. Die grundlegende Architektur der Informationslandschaft zeigt auf, daß der Inhaltsassistent an der Schnittstelle zwischen individuellen Bereichen und organisationalen Bereichen angesiedelt ist. Er stellt somit ein Hilfsmittel zur Koordination sozialer Belange in solch einer Informationslandschaft dar.

2.1 Ein Beispielszenario: Student sucht Studienplatz

Im folgenden wird in einem Beispielszenario die in Kapitel 1 beschriebene Aufgabenstellung der Informationssuche und -verteilung für einen konkreten Bereich mit seinen spezifischen Informationsproduzenten und -konsumenten instanziiert.

Studierende, die eine Universität für ihr Promotionsstudium in den USA suchten, waren bis vor wenigen Jahren vor allem auf die traditionellen Medien der Informationsverteilung angewiesen. Als Europäer mußten sie Vorabinformationen wie die Adressen und die Fakultäten amerikanischer Universitäten bei Kontaktstellen im Heimatland (Auslandsämter, Amerika-Häuser, etc.) besorgen und dann per Post Informationsmaterialien bei einzelnen möglicherweise in Frage kommenden Universitäten anfordern. Dieses zum Teil recht heterogene Material (allgemeine Informationen über die Stadt, die Wohn-

möglichkeiten, die Forschungsgebiete, Mitglieder der Fakultäten, Studiengebühren usw.) konnte der Student dann hinsichtlich seiner persönlichen Anforderungen (Interessen, Budget usw.) bewerten. Mit fortschreitendem Stand der Entscheidungsfindung änderte sich dann oft auch die Anforderung an das Informationsmaterial, so daß zusätzliche Quellen, etwa konkrete Arbeiten einzelner Professoren, herangezogen werden mußten.

Zusammen mit den Randbedingungen (etwa Bewerbungsfristen, Postlaufzeiten etc.) und den Anforderungen der Studierenden, eine "möglichst gute Uni" zu finden, sowie der Universitäten, "möglichst gute Studenten" zu bekommen, ergibt sich eine recht komplexe Aufgabenstellung, für deren gute Bearbeitung die *richtigen Dokumente* (von der Info-Broschüre bis zum wissenschaftlichen Zeitschriftenbeitrag) für die *richtigen Personen* zur *richtigen Zeit* (vom Beginn des Prozesses bis zur Entscheidungsfindung) zentral sind. Die in der Einleitung allgemein formulierte Aufgabenstellung des Informations- und Wissensmanagements findet sich in diesem Szenario also wieder.

Mit zunehmender Bedeutung des Internet wird die beschriebene Aufgabe mehr und mehr mit Hilfe der dort abgelegten Informationen gelöst. Fachbereiche der Universitäten haben ebenso eigene Seiten im WWW wie einzelne Professoren, Mitarbeiter, Projekte usw. Der Zugriff auf diese Informationen ist jedoch im allgemeinen nur recht unstrukturiert, über schlüsselwort-basierte Standardsuchmaschinen, oder mittels Browsing durch Hypertextstrukturen möglich. Da sich bei der Gestaltung von WWW-Seiten aber auch nur in geringem Maße (Quasi-)Standards herausgebildet haben, ist ein Auffinden der *richtigen* Dokumente oft jedoch schwierig. Der vorgeschlagene Informationsassistent könnte hier die Ergebnisse der Informationssuche verbessern, indem er den Kommunikationskanal zwischen Anbietern (Universitäten) und Konsumenten (Studenten) neben den "traditionellen" Medien (Telefon, Brief) sowie dem wenig strukturierten Internet-Informationsangebot um einen *proaktiven* Aspekt erweitert. "Proaktiv" bedeutet in diesem Zusammenhang, daß der Assistent schon aktiv wird, bevor eine konkrete Aufgabenstellung an ihn herangetragen wird. Dazu soll er verschiedene Repräsentationen der zur Verfügung stehenden Informationen aufbauen, die dann – je nach Problemstellung – flexibel eingesetzt werden können. Es ist daher notwendig, daß beim Entwurf des Informationsassistenten der *ganzen* Problemlösungsprozeß berücksichtigt wird.

2.2 Relevante Entitäten

Das oben beschriebene Szenario ermöglicht es, die für einen Informationsassistenten im Rahmen des Wissensmanagements relevanten Entitäten zu bestimmen. Gleichzeitig werden aber auch einige grundlegende Probleme klar. Relevante Entitäten sind im einzelnen diese:

- Eine Menge von *Dokumenten*: $D = \{d_1, d_2, \ldots\}$

  Hier tritt deutlich der Unterschied zwischen *offenen* und *geschlossenen* Welten zutage. Während es in geschlossenen Welten – etwa Firmenarchiven mit einem zentralen Administrator – zumindest theoretisch möglich ist, eine Repräsentation der vorhandenen Dokumente bezüglich Vollständigkeit und Korrektheit zu pflegen, ist dies in offenen Welten wie dem WWW völlig undenkbar. Selbst die größten Suchmaschinen decken nur einen Bruchteil des Netzinhaltes ab, und tote Verweise sind wegen der dezentralen Organisation unvermeidbar. *Matchmaker-* oder *Brokering*-Ansätze versuchen, diesen Nachteil zu vermeiden, indem Segmente des WWW als geschlossene Welten betrachtet werden. Der hier im folgenden vorgeschlagene Informationsassistent verfolgt im wesentlichen diesen Ansatz, so daß die Menge D der Dokumente in der Tat als wohldefiniert angesehen werden kann.

- Eine Menge von *Informationskonsumenten*: $K = \{k_1, k_2, \ldots\}$

  Informationskonsumenten lassen sich hinsichtlich vielfältiger Merkmale charakterisieren. Im Bereich von Unternehmensanwendungen kann man ebenso verschiedene Rollen unterscheiden (Mitarbeiter, Manager, Kunde, Partner, ...) wie im oben beschriebenen Studienplatz-Szenario (Student aus dem Ausland, Student der eigenen Universität, ...). Ein weiteres Merkmal ist auch die Kenntnisse eines Konsumenten hinsichtlich der Domäne. Diese können sich für den einzelnen Konsumenten auch im Laufe der Zeit ändern, wie oben am Beispiel des Studenten dargelegt wurde. In Informationssystemen werden Informationskonsumenten oft durch ihre Interessensgebiete repräsentiert. Solche *Konsumentenprofile* können sich auch über die Zeit hinweg entwickeln, beispielsweise mit dem Grad der Expertise oder der Rolle. Sie sind zentral für die Bestimmung der Relevanz von Dokumenten für den Konsumenten.

- Eine Menge von *Informationsanbietern*: $P = \{p_1, p_2, \ldots\}$

  Auch Informationsanbieter können bezüglich ihrer Rolle unterschieden werden. So hat der Autor eines Dokumentes möglicherweise völlig an-

dere Ziele (z.B. nur die Ablage) als ein Verteiler, der (deterministisch) sicherstellen möchte, daß alle relevanten Personen ein Dokument erhalten. Informationsanbieter sind im allgemeinen besonders geeignet eine *Primär*-Lesart eines Dokumentes anzugeben (Stichworte, Zusammenfassungen, mögliche Ablageorte, etc.). Es sollte jedoch berücksichtigt werden, daß oft viele andere Lesarten sinnvoll sind, wenn etwa das Dokument in einem anderen Kontext benutzt werden soll. Solche Angaben des Anbieters können also immer nur Anhaltspunkte sein und haben damit eher heuristischen Wert.

- eine Menge von *künstlichen intelligenten Akteuren*: $A = \{a_1, a_2, \ldots\}$

  Es hat sich als nützlich herausgestellt, solche künstlichen Akteure auf der Wissensebene[Newell 82] zu beschreiben. In den letzen Jahren wurden zum Begriff der Wissensebene einige Erweiterungen vorgeschlagen, die solche Beschreibungen um weitere Parameter ergänzte (vgl. [Schmalhofer et al. 94a]).[1] In dieser Arbeit sollen nun Zuschreibungen bezüglich folgender Parameter verwendet werden:

  - Handlungsziele: Die künstlichen Akteure handeln in sich ständig verändernden Umgebungen. Dabei *reagieren* sie nicht ausschließlich nur auf die Veränderungen ihrer Umgebung, sondern haben auch eigene langfristige Ziele, die sie zu verwirklichen suchen.
  - Wissen: Die künstlichen Akteure haben Wissen bezüglich der relevanten Bereiche ihrer Umgebung und der darin vorkommenden Objekte und Prozesse sowie bezüglich ihrer eigenen Ziele.
  - Kompetenz: Die Akteure haben ein bestimmtes Handlungspotential, also die Möglichkeit in ihrer Umgebung Aktionen ausführen zu können.
  - Kommunikationsverhalten: In der Umgebung der künstlichen Akteure können sich weitere Akteure befinden. Mit diesen können sie kommunizieren, d.h. Wissen über Fakten, Ziele, Kompetenzen usw. austauschen. Dadurch werden Absprachen möglich. Ziele können angepaßt und Aufgaben aufgeteilt werden. Das Kommu-

[1] Solche Beschreibungen auf der Wissensebene unterstellen einem Agenten rationales Verhalten und ermöglichen es insbesondere, dieses Verhalten menschlichen Benutzern verständlich zu machen. Beschreibt man etwa einen Thermostat auf der Wissensebene, so wird ihm *Wissen* über Temperaturen zugeschrieben, ferner das *Ziel*, die eingestellte Raumtemperatur herzustellen, und die *Kompetenz*, Temperaturen zu verändern. Von operationalen Details ("Wenn die Temperatur zu hoch ist, schalte die Heizung ab.") wird ebenso abstrahiert wie von Implementationsdetails ("Bimetall-Feder").

nikationsverhalten ist somit wesentlich, um künstliche Akteure als *soziale Entitäten* beschreiben zu können.

Die künstlichen Akteure können sowohl einzelnen Benutzern individuell zugeschrieben sein (*persönliche Assistenten*), als auch eher global agieren und damit soziale Aufgaben (*soziale Assistenten*) erfüllen.

2.3 Informationslandschaft

In diesem Abschnitt wird prototypisch die Architektur eines Informationssystems und der zugehörigen Informationslandschaft vorgestellt, in der ein Informationsassistent zum gemeinsamen Textverstehen eingesetzt werden kann. Solche Architekturen sind sowohl in Unternehmensumgebungen für das *corporate knowledge management* mittels Intranet-Technologien als auch für Anwendungen im Internet geeignet. Die Darstellung lehnt sich an der Konzeption eines Distributions- und Inhaltsassistenten für ein spezielles Intranet an (vgl. [Schmalhofer et al. 98b]).

Zwei wesentliche Aufgaben des *knowledge management* sind das Verteilen (die Distribution) von Dokumenten an alle relevanten Informationskonsumenten (*information push*) und das Suchen von Dokumenten (*information pull*, adhoc/kurzfristig oder regelmäßig/langfristig). Abbildung 2.1 stellt eine Architektur dar, die beide Aufgaben unterstützt.
Basis ist eine Topographie des Informationssystems mit drei Regionen. Die erste Region – in der Abbildung 2.1 im oberen Bereich gezeigt – beinhaltet die *organisationalen Bereiche* und besteht aus dem zugrunde liegenden Ablagesystem der Dokumente. Dies kann beispielsweise das Internet sein, aber auch eine firmenspezifische Informationsdatenbank als *corporate memory*, aufbauend etwa auf Intranet-Technologien, wäre hier angesiedelt. Diese Region stellt die grundlegenden Funktionen zur Identifizierung, zur Ablage und zum Zugriff auf Dokumente zur Verfügung. Die Organisation erfolgt entweder dezentral, wie im Falle des WWW, oder vielfach auch durch einen dezidierten Administrator, der – wie ein Bibliothekar – die Ordnungsstruktur des Ablagesystems wartet und neue Dokumente in das Archiv einpflegt. Letztere Form findet sich gerade in Unternehmensarchiven sehr häufig. Analog zur Organisationsform kann auch die technische Realisierung dieser Ebene entweder dezentral und massiv parallel (viele Server im Internet) oder eher zentral (eine oder wenige Datenbanken im Unternehmen) orientiert sein.

Die zweite Region des Informationssystem (vgl. Abbildung 2.1, Mitte) ist wesentlich für die *Koordination der organisationalen und sozialen Aspekte* des

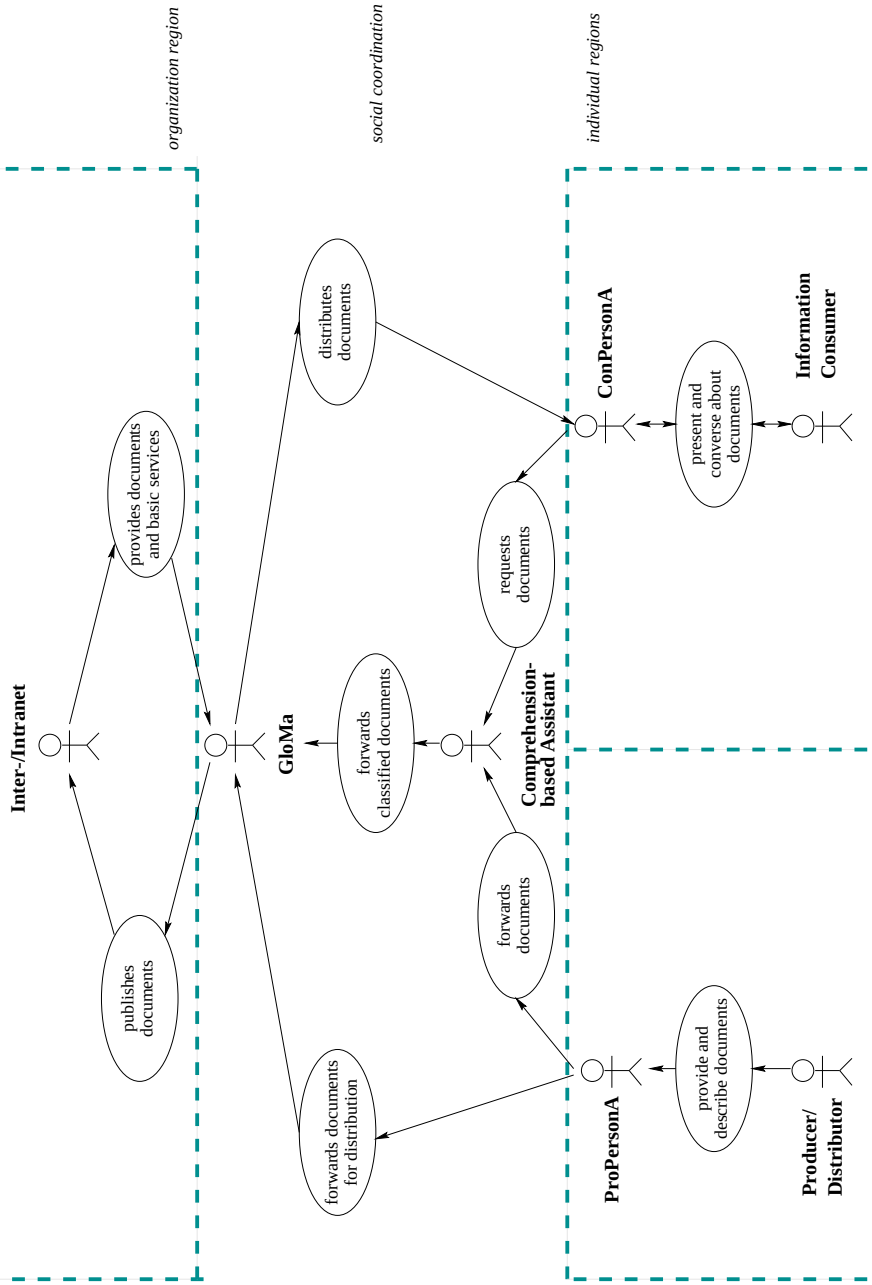

Abbildung 2.1: Integrierte Anwendungsfallmodellierung: In einer Informationslandschaft, die aus individuellen und globalen Regionen besteht, werden verschiedene Informationsassistenten (*GloMa, ConPersonA, ProPersonA, comprehension-based assistant*) eingesetzt, die individuelle und soziale Belange koordinieren.

Assistenzsystems sowie für die Koordination mit den individuellen Belangen der einzelnen Benutzer zuständig. Hier wird eine zentrale Repräsentation des Archivs erstellt und gepflegt. Diese Repräsentation besteht im Kern aus Verweisen auf die Archiv-Dokumente und sogenannten *Dokumentsurrogaten*. Diese Dokumentsurrogate werden *anstelle* der eigentlichen Dokumente zur Ermittlung der Relevanz für den Benutzer herangezogen. Grundlage zur Erstellung dieser Surrogate können sowohl die Dokumentinhalte selber sein, also die Worte, als auch Meta-Informationen über die Dokumente, etwa der Dateityp, der Ablageort, Inhaltsbeschreibungen durch den Autor etc. Da die Surrogate ein wesentlicher Faktor bei der Relevanzberechnung sind, ist die Wahl geeigneter Dokumentsurrogate oder -repräsentationen zentral für die Leistungsfähigkeit eines jeden Informationsassistenten und wird daher ein Schwerpunkt dieser Arbeit sein.
Als Koordinationsstelle für die sozialen Belange des Assistenzsystems ist diese Ebene außerdem für solche Prozeduren zuständig, die auf der Basis mehrerer oder aller Benutzer operieren. Ein solches Verfahren ist beispielsweise das *social matching* [Maes 94], das seine Relevanzberechnungen nur anhand der Benutzerprofile durchführt und völlig ohne Dokumentsurrogate auskommt. Als Koordinationsstelle für individuelle und soziale Belange ist diese Ebene – gerade hinsichtlich der Verfahren wie *social matching* – auch mit zuständig für die Wahrung der Schutzrechte und -interessen einzelner Benutzer.
Diese Ebene ist aufgrund ihrer zentralen Stellung zwischen den Benutzern und dem Archiv, aber auch zur Vereinfachung der oben angedeuteten Sicherheitsaspekte, normalerweise auf einem dezidierten Anwendungs-Server angesiedelt.

Komplementär dazu ist die dritte Region (Abbildung 2.1 unten) des Informationssystems für die *Realisierung der individuellen Belange der Benutzer* zuständig und daher hauptsächlich auf Seiten der Benutzer-Clients lokalisiert. Benutzer können sowohl Informationsanbieter wie auch -konsumenten sein. Auf dieser Ebene werden die konkreten Verteilungs- und Suchaufgaben definiert, die kurzfristiger (adhoc-Anfragen oder -Distributionen) wie auch längerfristiger Art sein können. Auch die Präsentation von Suchergebnissen oder von zugestellten Distributionen erfolgt hier. Durch individuelle Benutzerprofile, die auf dieser Ebene verwaltet werden, kann sowohl Suche als auch Präsentation den Erfordernissen des jeweiligen Benutzer angepaßt werden.

Abbildung 2.1 zeigt, wie die beiden angesprochenen Aufgaben des *knowledge management*, Informationssuche und Informationsverteilung, in einem aus den genannten Regionen bestehenden Informationssystem von verschiedenen Akteuren durchgeführt werden. Solche Akteure können sowohl menschliche Akteure als auch künstliche intelligente Agenten sein, mit jeweils spezifi-

schem Wissen, spezifischen Zielen und Kompetenzen. Zur Vereinfachung wurde der gesamte Bereich des Informationsarchives mit seinen Organisations-, Speicherungs- und Abruffunktionalitäten zu einem *Netz*-Agenten zusammengefaßt. Auch wenn diese Aufgaben insgesamt sehr komplex sein können, genügt es hier, diesen Akteuren allgemein das entsprechende Wissen und die Kompetenzen zur Durchführung zuzuschreiben.
Im individuellen Bereich werden hier zwei Benutzergruppen unterschieden. Auf der einen Seite sind das die Informationsproduzenten und -verteiler als Anbieter von Informationen, auf der anderen Seite die Informationskonsumenten als Abnehmer. Jedem Benutzer ist ein individueller künstlicher Agent zugeordnet, der ihn bei der Durchführung seiner Aufgaben unterstützt. Entsprechend gibt es zwei Typen von individuellen Agenten, einen für die Konsumenten (*ConPersonA*[2]) und einen für die Anbieter (*ProPersonA*[3]). Während die Anbieter-Agenten dem Benutzer bei der Einpflege der Dokumente oder bei der Definition von Verteilungsaufgaben helfen, sind die Konsumenten-Agenten für die Definition von Interessensprofilen und adhoc-Anfragen sowie für die Präsentation der Ergebnisse zuständig.
Die Koordination von individuellen und organisationalen Belangen wird, wie in Abbildung 2.1 dargestellt, durch zwei Agenten geleistet: Der *global manager GloMa* verwaltet eine zentrale Repräsentation des Netzes bzw. Archivs (s.o.), pflegt Dokumente ein und arbeitet konkrete Verteilungs- und Suchaufgaben ab. Ein Inhaltsassistent (*comprehension-based assistant*) erstellt die Dokumentsurrogate, auf deren Basis die Beurteilung der Relevanz von Dokumenten für einzelne Personen zu bestimmten Zeitpunkten vorgenommen wird. Was "Inhaltsassistenz" konkret bedeuten soll, wird in den folgenden Kapiteln detaillierter ausgearbeitet.

Zusammenfassend zeichnet sich die Architektur des Informationssystems für Distribution und inhaltsbasierte Suche durch folgende Eckpunkte aus:

- Individuelle und organisatorische Belange werden in einem System aus wenigen Agenten (*Oligo-Agenten-System*) koordiniert. Dieses ist an ein – möglicherweise schon unabhängig bestehendes – Informationsarchiv (Internet/ Intranet) angekoppelt.
- Ein *global manager* verwaltet eine zentrale Repräsentation des Dokumentenarchivs und aktualisiert diese kontinuierlich. Die dafür benötigten Dokumentensurrogate werden von einem *verstehensbasierten Assistenten* erzeugt.

[2] *ConPersonA* steht für "persönlicher Konsumenten-Assistent".
[3] *ProPersonA* steht für "persönlicher *Provider*-Assistent".

- Der *global manager* ist auch für die Abarbeitung konkreter Such- und Verteilungsaufgaben zuständig.
- Die individuellen Belange der Benutzer (Definition von Verteilungs- und Suchaufgaben, Präsentation der Ergebnisse) werden von individualisierten Assistenten (*ConPersonA*, *ProPersonA*) unterstützt.

Im weiteren Verlauf dieser Arbeit wird die Aufgabe der Informationsverteilung (Distribution) mit der Erstellung und Abarbeitung konkreter Verteilungsaufträge nicht mehr näher betrachtet. Für weitere Ausführungen dazu vergleiche [Schmalhofer et al. 98a] und [van Elst et al. 98].

2.4 Aufgabenstellung

Das allgemeine Wissensmanagementproblem, die richtigen Dokumente zur richtigen Zeit den richtigen Personen zur Verfügung zu stellen, findet sich in allen entscheidenden Merkmalen in dem oben beschriebenen konkreten Beispiel von Ausbildungsangebot und Ausbildungssuche instanziiert.

Aus dem Beispiel heraus wird offensichtlich, daß es hier verschiedene Akteure zu unterscheiden gibt (Informationsanbieter und Informationssucher) und daß eine Lösung nur durch einen adäquaten Interessensabgleich zustande kommen kann. In der Literatur unterscheidet man für den Interessensabgleich zwei verschiedene Ansätze, das *matchmaking* und das *brokering*. Während beim *brokering* ein dritter Akteur verantwortungsgebundene Aktionen durchführt (etwa Preise aushandelt), entstehen beim *matchmaking* nur Empfehlungen, die dann zwischen Anbieter und Sucher zu einer Vereinbarung führen.

Beim konkreten Beispiel der Abstimmung von Studienangebot und Studienwünschen handelt es sich offensichtlich um ein *matchmaking*-Problem. Für dieses *matchmaking*-Problem soll im folgenden eine Lösung erarbeitet werden, die geeignet ist, mit dem Trade-off zwischen Flexibilität und Struktur umzugehen.
Dokumente jeglicher Art, insbesondere auch Texte, haben bekanntermaßen (mehr oder weniger) wohlgeformte Strukturen, angefangen von der Befolgung der Rechtschreibregeln über Grammatikregeln [Chomsky 57], Textstrukturen [Mandler et al. 77] bis hin zu Kommunikationsprinzipien [Grice 75]. Da die Ziele von (Informations-)Anbietern und (Informations-)Konsumenten nicht notwendigerweise deckungsgleich sein müssen, sondern vielmehr naturgemäß oft sehr unterschiedlich sind (plakativ gesprochen: “auch der schlechteste Stu-

dent will an die beste Universität" und "auch die schlechteste Universität will die besten Studenten haben"), ist es im Laufe des *matchmaking*-Prozesses erforderlich, sich von den konkreten Strukturen des Anbieters und des Konsumenten zu lösen. Mit Hilfe von Abstraktionen soll dann eine Lösung gefunden werden, der beide Akteure zustimmen können (plakativ gesprochen: "jeder mehr oder wenig gute Studierende geht an eine mehr oder weniger gute Universität"). Wenn der Prozeß des *matchmaking* mit Informationstechnologie unterstützt werden soll, dann müssen auch die zugrunde liegenden Repräsentationen den Trade-off zwischen Flexibilität und Struktur widerspiegeln.

In dieser Arbeit soll für den Rahmen des Informations- und Wissensmanagements in einer Informationslandschaft, wie sie im vorausgehenden Abschnitt beschrieben wurde, ein Inhaltsassistent entworfen werden, der den Informationskonsumenten bei seiner Suche unterstützt, indem er einen inhaltsbasierten Zugriff auf Dokumente ermöglicht. Dieser Informationsassistent soll Dokumente so repräsentieren, daß die Kategorisierungen kompatibel mit denen der Benutzer sind. Mittels dieser Repräsentationen wird dann eine Beurteilung der Relevanz von Dokumenten für einen Informationskonsumenten zu einem bestimmten Zeitpunkt vorgenommen.

Kapitel 3

Lösungsansatz

Eine der zentralen Aufgaben von *information retrieval*- und *knowledge management*-Systemen, die das *matchmaking* in der beschriebenen Art unterstützen sollen, ist es, sowohl unstrukturierte Informationen, die flexibel eingesetzt und gedeutet werden können, als auch formalere Strukturen mit festerer Semantik handhaben zu können.

Zur Bearbeitung dieser Problemstellung werden domänenadäquate Abstraktionen (vgl. [Schmalhofer et al. 95a]) vorgeschlagen, die neben einer an der Oberflächenstruktur orientierten Repräsentation auch abstraktere terminologische Darstellungen sowie auf die Inhaltsdomänen bezogene Darstellungen erlauben.
So kann einerseits die Flexibilität der oberflächen-orientierten Repräsentationen genutzt werden, andererseits aber auch strukturelle und inhaltliche Information besser eingesetzt werden, um so zu angemesseneren Ergebnissen bei der Informationssuche zu kommen.

Der Inhaltsassistent repräsentiert Dokumente daher auf drei verschiedenen Ebenen, die sich vor allem hinsichtlich des Abstraktionsgrades vom Dokumenttext unterscheiden.
Während auf der ersten Ebene, der *wörtlichen* Ebene, die Dokumente noch sehr nahe an der ursprünglichen Textoberfläche repräsentiert werden, versucht die *propositionale* Ebene schon, vom genauen Wortlaut zu abstrahieren. Auf der *situativen* Ebene wird dann eine abstrakte Bedeutung des gesamten Dokumentes repräsentiert. Diese Ebene kann auch ein detaillierteres Modell der Domäne und damit unterschiedliche Interpretationsrahmen für die einzelnen Dokumente enthalten. Ein solches Modell kann insbesondere für den Experten-Nutzer sinnvoll und hilfreich sein, während ein Neuling eher mit den ersten beiden Ebenen arbeitet, die ohne tiefere Theorie auskommen.

Für den Prozeß des *matchmaking* werden Konstruktions- und Integrationsphasen vorgeschlagen, wie sie auch im Rahmen von Theorien der Wissensakquisition (vgl. [Schmalhofer 98], [Schmalhofer et al. 95b]) und des Textverstehens (als allgemeines Paradigma für Kognition [Kintsch 98]) postuliert werden.
Während die Konstruktionsphasen dabei einerseits aus der Erstellung eines (oder mehrerer) abstrakten Bezugsrahmens und andererseits aus Abstraktionen für individuelle Informationseinheiten bestehen, erfordert die Integrationsphase die Reduzierung von Redundanzen und Inkonsistenzen.[1]

Abbildung 3.1 zeigt zusammenfassend die Eckpunkte für den Entwurf des Inhaltsassistenten auf: Auf der Basis eines (von Informationsanbietern und -konsumenten gemeinsam vereinbarten) Repräsentationsrahmens werden Dokumente auf den oben skizzierten drei Ebenen dargestellt.
Zusammen mit der Anfrage des Informationskonsumenten wird eine Liste von Dokumenten erstellt, die für den Konsumenten relevant sein könnten. Sowohl die Erstellung der Drei-Ebenen-Repräsentation der Dokumente als auch die Erstellung der Liste relevanter Dokumente wird durch Konstruktionsprozesse geleistet.
Diese Liste potentiell relevanter Dokumente kann jedoch redundant sein (wenn sie etwa sehr viele ähnliche Dokumente enthalten) oder auch inkonsistent sein (Dokumente werden von einem Konstruktionsprozeß für relevant befunden, von einem anderen jedoch nicht). Integrationsprozesse versuchen daher, auf der Basis der Anforderungen des Konsumenten (das können sowohl die Anfragen als auch längerfristige Benutzermodelle und -profile sein) solche Redundanzen und Inkonsistenzen aufzulösen und eine kohärente Sicht zu erzeugen. Da sich die Anforderungen im Laufe des Prozesses leicht ändern können (zum Beispiel, wenn der Konsument ein Dokument tatsächlich gelesen hat), kann die Integrationsphase über mehrere Iterationen verfügen.

Im folgenden Kapitel soll dieser allgemeine Ansatz verfeinert werden, indem konkrete Konstruktions- und Integrationsprozesse spezifiziert werden. Diese sollten so gewählt sein, daß sie möglichst automatisch ablaufen können und sich in einer Informationslandschaft, wie sie in Abbildung 2.1 gezeigt wurde, ansiedeln lassen.

[1] Beschreibt man beispielsweise Meta-Suchmaschinen wie etwa den *Metacrawler* [Selberg et al. 97] auf dem Hintergrund von Konstruktions- und Integrationsphasen, dann kann man die Einzelanfragen bei den jeweiligen Suchmaschinen als Konstruktionsprozesse auffassen. Es werden Hypothesen generiert, welche Dokumente relevant sein könnten. Diese Einzelergebnisse werden dann aufbereitet, indem etwa doppelte Seiten entfernt und die verschiedenen Einzelrelevanzen zusammengefaßt werden. Dies kann man als Informationsintegration bezeichnen.

Dabei gibt es im Gegensatz zu den verfügbaren Techniken der Konstruktionsphase, die bereits mehr oder minder gut ausgearbeitet wurden, für die Erfordernisse der Informationsintegration bisher noch keine zufriedenstellenden Lösungen. Dies wird insbesondere auch bei den derzeit verfügbaren Standardsystemen für das *information retrieval* im Internet oder in Intranets ersichtlich, die nahezu keine Integrationskomponenten besitzen.

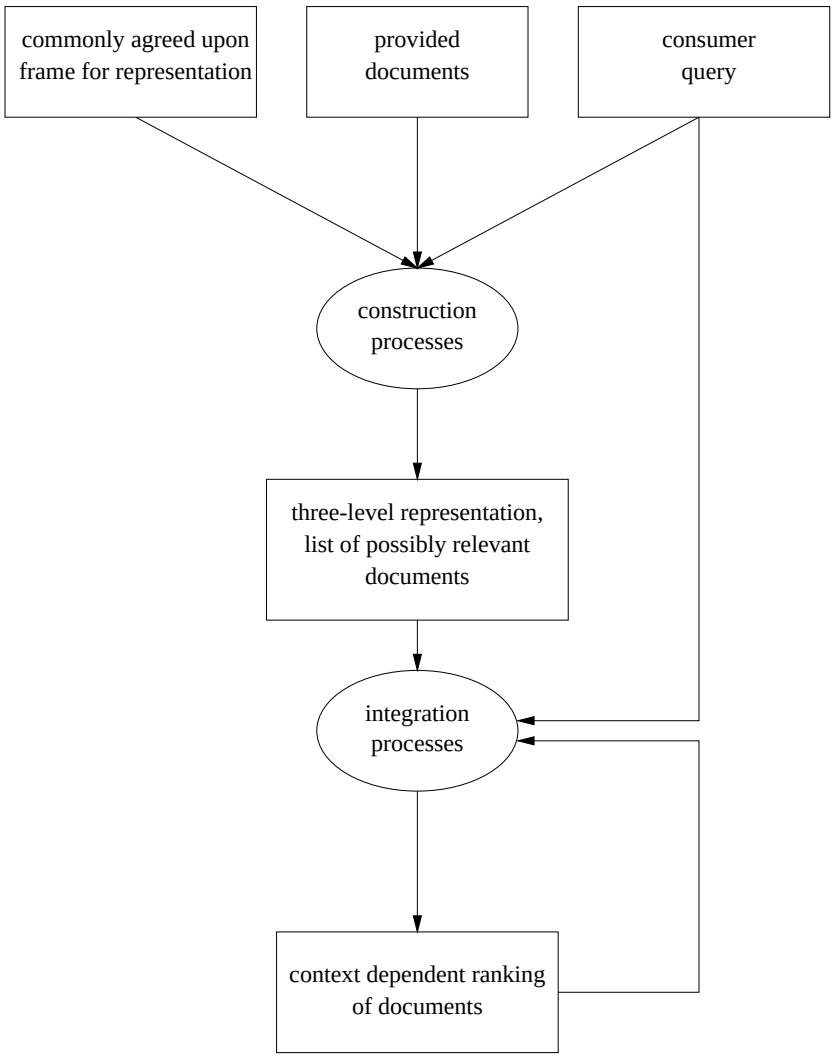

Abbildung 3.1: Grundstruktur des Inhaltsassistenten: Konstruktionsprozesse stellen Dokumente auf drei verschiedenen Ebenen auf der Basis eines festen Repräsentationsrahmens dar und erzeugen zusammen mit der Konsumentenanfrage eine Liste von möglicherweise relevanten Dokumenten. Integrationsprozesse individualisieren die Darstellung dieser Liste, indem sie sie auf den speziellen Kontext anpassen.

Kapitel 4

Inhaltsassistent

Im vorhergehenden Kapitel wurden Eckpunkte eines Inhaltsassistenten für das Informations- und Wissensmanagement spezifiziert. Der Inhaltsassistent über soll drei Ebenen der Dokumentrepräsentation verfügen. Die *Oberflächenebene* soll Dokumente hinsichtlich der vorkommenden Worte darstellen. Auf der *propositionalen Ebene* wird von den konkreten Formulierungen abstrahiert. Die *situative Ebene* soll die Bedeutung eines Dokumentes als Ganzes erfassen.
Die Prozesse des Inhaltsassistenten lassen sich entweder einer *Konstruktionsphase* zuordnen, in der die entsprechenden Repräsentationen aufgebaut werden und für den Benutzer möglicherweise relevante Dokumente ausgewählt werden, oder einer *Integrationsphase*, die das Ergebnis der Informationssuche speziell den Benutzeranforderungen anpaßt.
Im folgenden werden die Techniken der Repräsentation und der Prozesse für die beiden Phasen im einzelnen dargestellt. Es kommen dabei sowohl Standard-Verfahren des *information retrieval* als auch aktuelle Forschungsergebnisse sowie innerhalb dieser Arbeit eigenständig entwickelte Techniken zur Anwendung. Die Datenstrukturen sind so gewählt, daß sie innerhalb eines betrieblichen Informationsmanagement-Systems zum Beispiel in einer relationalen Datenbank gut verwaltet werden können.

4.1 Konstruktionsprozesse: Drei Ebenen der Repräsentation

4.1.1 Oberflächendarstellung

Repräsentation

Der Oberflächenrepräsentation liegt ein Vektorraummodell [Salton 71] zugrunde, wie es in vielen *information retrieval*-Systemen benutzt wird. In diesem Modell werden die Dokumente als Punkte in einem Vektorraum aufgefaßt, der von den Termen der Datenbasis aufgespannt wird. Als Dokumentsurrogate werden in diesem Modell gewichtete Term-Vektoren benutzt, wobei als Standard-Gewichtung die Häufigkeit verwendet wird, mit der ein Term in dem Dokument vorkommt.
Grundlegende Datenstrukturen sind somit die folgenden:

- eine Menge von *Dokumenten*:

$$D = \{d_1, d_2, \ldots d_k\}, \tag{4.1}$$

  wobei k die Anzahl der Dokumente in der Datenbasis ist.

- eine Menge von *Termen*:

$$V = \{v_1, v_2, \ldots v_l\}, \tag{4.2}$$

  d.h., es gibt l verschiedene Worte in den Dokumenten. Die v_i spannen somit einen l-dimensionalen Vektorraum auf.

- eine (gewichtete) $l \times k$ *Term-Dokument-Matrix*:

$$TDM = (tdm_{ij}), \text{ mit } 1 < i < l \text{ und } 1 < j < k, \; tdm_{ij} \in N \tag{4.3}$$

  Eine solche Term-Dokument-Matrix stellt eine Beziehung zwischen der Menge aller Dokumente und der Menge aller Terme her. Eine der einfachsten und am häufigsten benutzte Beziehung ist die Häufigkeit, mit der ein Term v_i in einem Dokument d_j vorkommt(*term frequency*). In diesem Fall gilt $tdm_{ij} = tf_{ij}$ mit

$$tf_{ij} := tf(v_i, d_j) := \text{occurences of term } v_i \text{ in document } d_j \tag{4.4}$$

  Man beachte, daß schon diese Darstellung eine ganz enorme Abstraktion von den tatsächlichen Dokumenten darstellt, da sich beispielsweise

die linguistische Struktur nicht widerspiegelt und damit die Reihenfolge der Worte in den Dokumenten keinerlei Rolle mehr spielt.
Die Verwendung der Termhäufigkeit hat jedoch einige Nachteile. Sie bezieht beispielsweise nicht die unterschiedliche Länge von Dokumenten. In Kapitel 5 werden einige Gewichtungsfunktionen vorgestellt, die im praktischen Einsatz bessere Ergebnisse liefern als die Termhäufigkeit (vgl. Tabellen 5.3 und 5.3).

Abbildung 4.1 zeigt, wie eine solche Term-Dokument-Matrix aussieht. Die Spaltenvektoren werden dann als Surrogate der jeweiligen Dokumente benutzt.

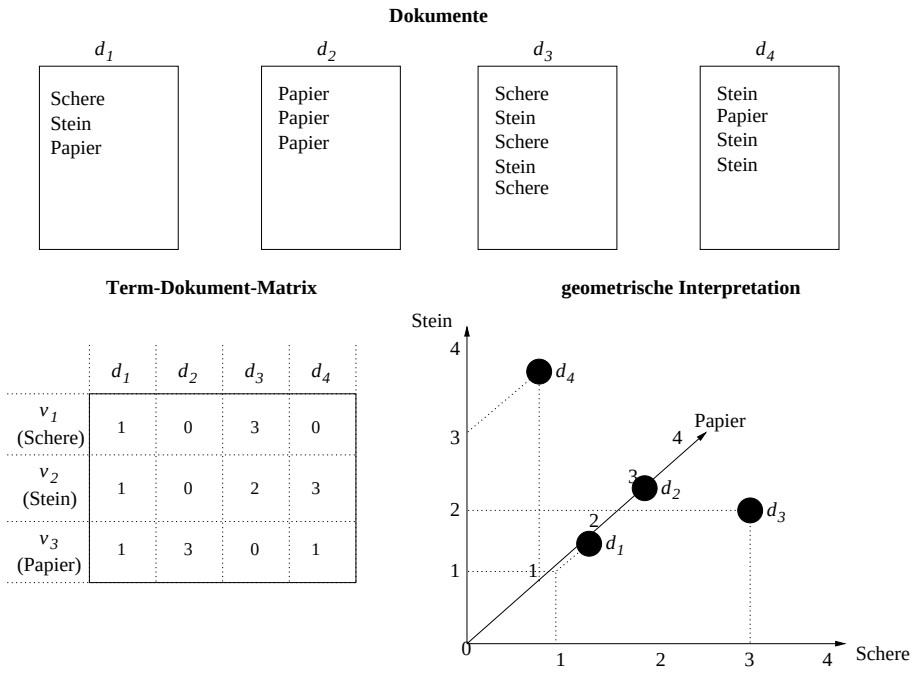

Abbildung 4.1: Ein Beispiel für die Repräsentation von Dokumenten in einer Term-Dokument-Matrix. Dargestellt ist weiterhin die geometrische Interpretation: Dokumente werden als Vektoren in einem Vektorraum aufgefaßt, den die Terme aufspannen.

Anfragen

Will man in diesem Modell zwei Dokumente miteinander vergleichen, so kann man dazu prinzipiell beliebige Vektor-Ähnlichkeitsmaße verwenden. Weit verbreitet ist es, dazu das Skalarprodukt oder das normierte Skalarprodukt (Kosinus-Maß) Sim_{cos} der beiden Dokumentvektoren zu verwenden. Dieses bewertet zwei Dokumente d_x und d_y als ähnlich, wenn der Winkel zwischen ihren Vektoren klein, der Kosinus also nahe bei 1 ist.

$$Sim_{cos}(d_x, d_y) = \frac{\sum_{i=1}^{l}(tf(v_i, d_x)tf(v_i, d_y))}{\sqrt{\sum_{i=1}^{l} tf(v_i, d_x)^2}\sqrt{\sum_{i=1}^{l} tf(v_i, d_y)^2}} \tag{4.5}$$

Stellt man nun die Anfragen der Informationskonsumenten auch als Vektoren im gleichen Raum wie die Dokumente dar, so kann die Ähnlichkeit zwischen Anfrage und Dokumenten analog zur Ähnlichkeit zwischen zwei Dokumenten berechnet und als Relevanzmaß benutzt werden. Da für die Anfrage Term-Häufigkeiten nicht zur Verfügung stehen, müssen Term-Gewichte angegeben werden, deren Wahl natürlich erheblichen Einfluß auf das Suchergebnis hat. Eine Anfrage kann also als Funktion gesehen werden, die jedem Term v_i aus V ein Gewicht $w(v_i)$ zuordnet. Damit ergibt sich die Relevanz eines Dokumentes d_x bezüglich dieser Anfrage folgendermaßen:

$$Rel_{cos}(w, d_x) = \frac{\sum_{i=1}^{l}(w(v_i)tf(v_i, d_x)}{\sqrt{\sum_{i=1}^{l} w(v_i)^2}\sqrt{\sum_{i=1}^{l} tf(v_i, d_x)^2}} \tag{4.6}$$

Als Antwort auf eine Anfrage wird dem Informationskonsumenten dann eine sortierte Liste der Dokumente mit den höchsten Relevanzwerten geliefert.
In der Praxis oft angewendete Verfeinerungen des Vektorraummodells, die zum Beispiel besser mit sehr ungleich langen Texten umgehen können, beziehen sich hauptsächlich auf die Gewichtung der Dokumentvektoren und werden in Kapitel 5 dargestellt, das sich mit der Implementierung des Informationsassistenten beschäftigt.

Vor- und Nachteile des Vektorraummodells

Zwei der Probleme dieses Verfahrens, das man als *schwaches* Verfahren (*weak method*) bezeichnen kann, da es auf jegliches tieferes Wissen verzichtet, zeigen sich bei sinngleichen und bei mehrdeutigen Worten (Synonyme und Polyseme).
Beide Probleme ergeben sich im wesentlichen prinzipiell aus der Orientierung

an der Textoberfläche, also daraus, daß lediglich Zeichenketten *ohne Semantik* betrachtet werden. Im ersten Fall werden weniger tatsächlich relevante Dokumente gefunden (niedrigerer *recall*-Wert), weil die Synonyme nicht zu einer Erhöhung der Relevanz beitragen. Die Gewichtung im Anfragevektor nicht vorkommender, aber sinngleicher Worte, ist Null. Im zweiten Fall werden zu viele irrelevante Dokumente geliefert (niedrigerer *precision*-Wert), da auch Dokumente, die die Terme der Anfrage in einer nicht gemeinten Bedeutung enthalten, als relevant bewertet werden.
Die beschriebenen Effekte senken also die Qualität des Suchergebnisses in einer Form, wie sie aus der Benutzung von Standard-Suchmaschinen des *world wide web* wohlbekannt ist.[1]
Nichtsdestotrotz sollte der Nutzen einer an der Textoberfläche orientierten Repräsentation für die Relevanzbewertung von Dokumenten innerhalb eines Informationsassistenten nicht unterschätzt werden. So ist dieses Verfahren zum einen genauso schnell (wichtig!) wie einfach (und damit für den Benutzer nachvollziehbar!). Zum anderen ist bei vielen Schritten innerhalb eines Vorgangs der Informationssuche (d.h. innerhalb des Problemlöseprozesses) die Formulierung in wörtlichen Kategorien durchaus möglich und notwendig.[2] In dem Szenario aus Abschnitt 2.1 denke man beispielsweise an einen Studenten, der unbedingt an der "Washington State University", nicht aber an der "University of Washington" studieren möchte. In diesem Fall kann die genaue wörtliche Formulierung sehr wichtig sein.

4.1.2 Propositionale Darstellung

Auf dieser Ebene der Repräsentation soll von der Textoberfläche auf Wortebene abstrahiert werden, um so die oben beschriebenen Nachteile der Oberflächendarstellung zu reduzieren.
Im ersten Teil dieses Abschnittes werden Forschungsansätze dargestellt, die sich diesem Problem widmen. Diese basieren hauptsächlich auf der Verwendung eines kontrollierten Vokabulars innerhalb der Dokumentsurrogate. Sie unterscheiden sich darin, wie dieses Vokabular gefunden wird und wie die Surrogate konkret erzeugt werden.
Der zweite Teil beschreibt, wie kontrollierte Dokumentationssprachen innerhalb des Inhaltsassistenten eingesetzt werden. Mit der *CL-Propositionalisie-*

[1] [Furnas et al. 84] haben beispielsweise gezeigt, daß zwei Personen nur zu etwa 10-20% der Zeit dasselbe Wort benutzen, um ein bestimmtes Objekt zu beschreiben

[2] Die in fast allen Suchmaschinen vorgesehene Möglichkeit, Suchbegriffe in Anführungszeichen zu klammern, trägt dieser Tatsache Rechnung.

rung[3] wird ein Verfahren vorgeschlagen, das auf der Grundlage eines kontrollierten Wortschatzes vollautomatisch Dokumentsurrogate erstellt, die von der exakten Wortwahl von Dokumenten abstrahieren und damit mehr von der Semantik erfassen.

Relevante Ansätze aus der Literatur

Komplexe semantische Netze Einer der bekanntesten Ansätze aus der Expertensystemforschung zur Darstellung von Wissen, der mit festgelegten Dokumentationssprachen arbeitet, ist Verwendung *komplexer semantischer Netze*. Solche Netze erlauben es, daß die Dokumentsurrogate aus Vokabeln eines kontrollierten Wortschatzes gebildet werden. Dieser kontrollierte Wortschatz erhöht beim *information retrieval* die *precision* (da es keine mehrdeutigen Begriffe, d.h. keine Polyseme gibt) genauso wie den *recall* (da verschiedene Textformulierungen auf eine einzige Bezeichnung abgebildet werden, es gibt keine Synonyme).

Das bekannteste Beispiel für semantische Netzwerke ist der KL-ONE Formalismus [Brachman et al. 85], in dem Taxonomien entwickelt werden können, deren Instanzen zur Darstellung von Textinhalten in semantischen Netzen dienen. Solche Netze bestehen im wesentlichen aus Instanzen von Konzepten und Rollen. Die Konzeptinstanzen können als Knoten eines Graphen angesehen werden, die Rolleninstanzen als Kanten. Ein solcher Graph spiegelt dann die Beziehungen zwischen den Objekten eines Dokumentes in einer kontrollierten Sprache wider.
Anfragen können als Graphen betrachtet werden, die Variablen enthalten. Das Beantworten von Fragen geschieht dann dadurch, daß im semantischen Netz ein Teilgraph gefunden werden muß, der mit dem Graphen der Anfrage durch Substitution unifiziert werden kann. Eine solche Teilgraph-Suche kann sehr aufwendig sein, so daß oft komplexe Heuristiken notwendig werden [Schmid et al. 96].
Ein großes Problem bei semantischen Netzen ist allerdings, daß es immer noch unklar ist, wie das Netz auf der Basis eines Textdokumentes erzeugt wird, selbst wenn bereits eine geeignete Taxonomie vorhanden sein sollte. Dazu liegt bis heute keine überzeugende Lösung vor.[4]

[3] *CL* steht für "*controlled language*".

[4] Das *CYC*-Projekt zum Verstehen natürlicher Sprache [Lenat et al. 90] basiert wesentlich auf dem Ansatz komplexer semantischer Netze. Es geht davon aus, daß zuerst eine umfangreiche Akquisitionsphase nötig ist, in der die Netze manuell von Wissensingenieuren aufgebaut werden, und daß das System dann irgendwann (geplant waren zehn Jahre ab 1984) selbständig lernen kann [Lenat 89]. Diese Erwartung hat sich bisher nicht bestätigt.

Um die Vorteile von Dokumentationssprachen mit festen Ontologien trotzdem nutzen zu können, kann man dazu übergehen, nicht die gesamten Dokumente in dieser Form zu repräsentieren sondern nur eingeschränkte Aspekte. Solche Aspekte können sich neben den Inhalten von Dokumenten auch auf Meta-Informationen (etwa ihre Position in hierarchisch strukturierten Wissensspeichern) beziehen.
Projekte wie der *Meta Content Framework* [Guha 97] oder der *Resource Description Framework* des W3C-Konsortiums stellen Ansätze zur Standardisierung für die Notation sowohl semantisch reichhaltiger Inhaltsbeschreibungen als auch struktureller Information zur Verfügung.

Im folgenden werden Verfahren vorgestellt, mit denen Dokumentsurrogate auf der Basis von kontrollierten Dokumentationssprachen erzeugt werden können.

Informationsextraktion mit maschinellem Lernen In der Literatur der Wissensakquisition werden schon seit langem maschinelle Lernverfahren vorgeschlagen, um Fakten aus Textdokumenten zu gewinnen. Ein Ansatz zum automatischen Extrahieren von symbolischem Wissen aus Dokumenten (etwa aus dem *world wide web*) ist in [Craven et al. 98] dargestellt. Das Ziel ist hier die automatische Erzeugung und Wartung einer computerverständlichen Wissensbasis, die den Inhalt des WWW widerspiegelt. Dazu wird eine Ontologie zugrunde gelegt, die die relevanten Klassen und Rollen spezifiziert. Es werden dann Dokumente vorgelegt, die Instanzen der definierten Klassen und Rollen repräsentieren. Aus diesen Beispielen versucht das System allgemeine Verfahren zu lernen, wie aus neuen Dokumenten neue Instanzen extrahiert werden können. Solche Extraktionsverfahren werden als Regeln erster Ordnung dargestellt. Als Lernverfahren für die Erzeugung der Regeln werden Varianten des FOIL-Algorithmus [Quinlan et al. 93] benutzt, der im wesentlichen mit Hilfe einer *hill climbing*-Strategie Horn-Klauseln ohne Funktionen lernt. Die zugrunde liegende Hintergrundtheorie enthält Prädikate wie `has_word(document)` oder `link_to(document, document)`, d.h. die Hyperlink-Struktur der Dokumente wird ausgenutzt. Dadurch können Konzepte gelernt werden, die über mehrere Dokumente verteilt sind.
Mit diesem Verfahren können sowohl Dokumente als ganze klassifiziert werden als auch Informationen aus Dokumenten extrahiert werden. Die positiven Beispiele, die das maschinelle Lernverfahren für das Extrahieren von Textfragmenten aus Dokumenten bekommt, bestehen aus Dokumenten, in denen Teile als Instanzen einer Klasse markiert sind. Nicht markierte Dokumentabschnitte sind dann negative Beispiele. [Craven et al. 98] geben an, daß damit eine Genauigkeit von ungefähr 77% bei der extrahierten Information

erreicht wird. Es ist zu beachten, daß solche *fact* oder *information extraction*-Verfahren, die man sich als das Füllen von Feldern einer Datenbank vorstellen kann, normalerweise nur für die Feldnamen ein kontrolliertes Vokabular verwenden, während die Feldinhalte immer noch als Zeichenstrings betrachtet werden. Dort ergeben sich also wieder die Nachteile, wie sie schon auf der Oberflächenebene identifiziert wurden, wenn auch nicht mehr in gleichem Maße.

Benutzergruppen mit einem gemeinsamen Bezugsrahmen: Der Ontobroker-Ansatz Im Gegensatz zu den oben beschriebenen maschinellen Lernverfahren werden beim Ontobroker-Ansatz [Fensel et al. 98] die Dokumente manuell annotiert. Ontobroker stellt Sprachen zur Verfügung, um Ontologien zu repräsentieren, Web-Dokumente mit ontologischer Information zu annotieren und Anfragen zu formulieren. Ferner verfügt Ontobroker über eine Inferenzmaschine, die es möglich macht, auch auf nichtexplizites Wissen zuzugreifen sowie über einen Web-Crawler, der das Wissen im WWW sammelt.
Zentral für den Ontobroker ist seine "*newsgroup*-Metapher", die eine Gruppe von Akteuren postuliert, die ein *gemeinsames Thema* bearbeiten und eine *ähnliche Sichtweise* haben. Für diese *Ontogroups* [Fensel et al. 97] wird ein intelligenter Informations-Broker als Service zur Verfügung gestellt.
Konkret benutzt Ontobroker eine auf der Frame-Logik [Kifer et al. 95] basierende Repräsentationssprache. Die Annotation der Dokumente mittels kleiner Erweiterungen von HTML innerhalb der Dokumente. Dadurch soll die Forderung des Wissensmanagements, Oberflächeninformation, halbformales und formales Wissen eng zu verzahnen [Kühn et al. 97], besser erfüllt werden. Eine solche enge Verzahnung wurde auch schon für die Wissensakquisition bei der Expertensystementwicklung vorgeschlagen [Schmalhofer et al. 91].
Die Beispielanwendung benutzt die $(KA)^2$-Initiative [Benjamins et al. 98] als *Ontogroup*, die eine gemeinsame Ontologie zur Beschreibung von Forschungsgruppen, Themen, Produkten etc. benutzt. Die Web-Dokumente dieser Gruppe werden dann von Hand annotiert und dem System zur Verfügung gestellt. Im Gegensatz zu SHOE [Luke et al. 97], das keinen solchen zentralen Service postuliert und das beliebige (lokale) Erweiterungen der Ontologie zuläßt, kann man beim Ontobroker immer davon ausgehen, daß die Semantik für die Mitglieder der jeweiligen *Ontogroup* tatsächlich eindeutig ist, da ja ein expliziter Einigungsprozeß postuliert wurde.

Kontinuierliche Propositionalisierung

In diesem Abschnitt wird – auf der Basis der oben dargestellten Ansätze aus der Literatur – die propositionale Ebene des Inhaltsassistenten spezifiziert. Dazu wird zuerst der verwendete Propositionsbegriff genauer definiert. Anschließend wird mit der *CL-Propositionalisierung* ein Verfahren vorgeschlagen, das automatisch aus Textdokumenten solche Propositionen für die Verwendung in Dokumentsurrogaten erzeugt.

Um Vorteile nutzen zu können, die die Repräsentation von formalem Wissen (sowohl bezüglich der Nutzung von faktischem Wissen als auch der Inferenz von neuem Wissen) bietet, aber auch der Tatsache Rechnung zu tragen, daß eine volle Formalisierung beim momentanen Stand der Forschung nicht möglich erscheint, wird für den Inhaltsassistenten eine schrittweise Propositionalisierung vorgeschlagen.
Der Begriff der *Proposition* wird in vielen Wissenschaftsbereichen benutzt, etwa in der Philosophie, der Linguistik, der Kognitiven Psychologie, der Logik und der Künstlichen Intelligenz (vgl. [Fischer et al. 96]). Allen Verwendungen ist gemein, daß Propositionen abstrakte Einheiten sind, die kleinste oder elementare Aussagen beinhalten. Im Inhaltsassistenten werden Propositionen als Bausteine für die Repräsentation von Dokumentinhalten verstanden, die folgende Eigenschaften haben:

- Propositionen beinhalten *elementare Aussagen*. Daraus folgt insbesondere, daß sie keine Variablen enthalten dürfen.
- Propositionen können sich auf verschiedenartige *Dokumentsegmente* beziehen. Es kann also Propositionen geben, die sich auf den gesamten Text beziehen wie etwa klassifizierende Schlagworte, die Textart oder ähnliches, aber auch solche die kleinere Einheiten zur Grundlage haben wie zum Beispiel "Titel, Einleitung, Hauptteil, Schluß" oder einzelne Sätze oder auch Satzsegmente.
- Propositionen sind syntaktisch eine Liste mit einem ausgezeichneten ersten Element, dem *Prädikat*. Die übrigen Listenelemente heißen *Argumente*.
- *Prädikate* können nur Konzepte einer zugrunde liegenden *Ontologie* und damit einer kontrollierten Sprache sein.
- Als *Argumente* sind sowohl Konzepte der kontrollierten Sprache als auch "Zeichenketten" erlaubt.

Die hier beschriebene Verwendung des Propositionsbegriffs basiert somit auf der Verwendung in der Literatur des Textverstehens (vgl. dazu [Kintsch 98], [Schmalhofer 98], [Schmid et al. 96]). Durch die Verwendung von Ontologien werden jedoch bessere Berechenbarkeitseigenschaften erreicht (indem beispielsweise die Gleichheit leicht entscheidbar ist).

Da sich Propositionen auf verschiedenartige Dokumentsegmente beziehen können, ist eine kontinuierliche Erweiterung der propositionalen Darstellung der Dokumente erreichbar.
In einem ersten Schritt könnten als Konzepte der Ontologie generische Strukturierungsattribute benutzt werden, wie sie für die meisten Textdokumente zur Verfügung stehen. Für HTML-Dokumente wären dies Marken wie "*TITLE*", "*CITE*", "*BODY*" oder die Überschriftsmarken "*H1*" bis "*H6*". Die Argumente für Prädikate, die auf diesen Konzepten basieren, sind dann die entsprechenden Dokumentsegmente, so daß im Vergleich zur Oberflächenebene die Dokumentstruktur besser repräsentiert werden kann. Dadurch können Anfragen formuliert werden, die nach Dokumenten mit einem bestimmten Aufbau suchen. Weiterhin können sich Anfragen auf spezifische Textsegmente beziehen.
Diese generischen Konzepte können nach und nach um domänenspezifische Elemente erweitert werden, wie sie etwa im Abschnitt über das Ontobroker-System beschrieben wurden. Diese Erweiterung der Ontologie geschieht dabei in Abstimmung zwischen Informationsanbietern, Informationskonsumenten und den Administratoren des Informationsassistenten. Lokale Veränderungen und Erweiterungen, wie sie zum Beispiel in SHOE möglich sind, sind hier nicht zulässig, so daß eine von allen gemeinsam verstandene Semantik der Konzepte gewährleistet ist.

Die Propositionalisierung der Textdokumente soll automatisch vonstatten gehen. Dies ist in unterschiedlichen Inhaltsdomänen jedoch nur zu verschieden großen Anteilen möglich. Während in gut strukturierten Gebieten, etwa den *homepages* von Personen oder Forschungsprojekten, mit maschinellen Lernverfahren gute Ergebnisse zu erzielen sind, ist deren Einsatz im allgemeinen nicht immer sinnvoll, da die Qualität der Lernergebnisse oft nicht ausreicht. Vielfach kann deshalb nicht auf die manuelle oder halbautomatische Annotation von Dokumenten verzichtet werden, wie sie schon in vielen Wissensakquisitionssystemen durchgeführt wurde (siehe [Schmidt 92]). Mit fortschreitender Standardisierung von Leitlinien für die Erstellung von Dokumenten im *world wide web* (z.B. XML [W3-Consortium 98]) ist damit zu rechnen, daß viele Dokumente schon bei der Erstellung mit semantischer Information angereichert werden (ähnlich den Marken für die Textstruktur, die es in HTML schon gibt). Diese Information kann dann für die Erzeugung der

Dokument-Surrogate der propositionalen Ebene ausgelesen werden.

Um für bereits jetzt existierende Dokumente einen möglichst hohen Anteil von kontrollierter Sprache (im Gegensatz zu den "Zeichenstrings") in den Propositionen zu haben, wird neben dem maschinellen Lernverfahren, wie es oben postuliert wurde, das folgende Propositionalisierungsverfahren vorgeschlagen.

CL-Propositionalisierung Die *CL-Propositionalisierung* basiert auf der Idee, daß man mit Hilfe von Thesaurus-Techniken und Normalformbildung Propositionen erzeugen kann, die ganz oder zu großen Teilen aus Worten einer kontrollierten Konzeptsprache aufgebaut sind. Das Verfahren besteht aus einer Vorverarbeitungsphase, in der das Vokabular der kontrollierten Konzeptsprache festgelegt wird, und der eigentlichen Propositionalisierungsphase.

1. Vorverarbeitung

   (a) Es wird ein *Thesaurus* erstellt, der die in den Dokumenten vorkommenden Terme in *Äquivalenzklassen*, also Mengen von Synonymen einteilt. Dazu können allgemeine, domänenunabhängige Thesauri oder auch auf der Basis der bisher bekannten Dokumente erstellte Thesauri benutzt werden (siehe [Kühn et al. 98]).

   (b) Für jede solche Äquivalenzklasse wird ein *eindeutiger Repräsentant* ermittelt. Für die oben beschriebenen Konzepte der festgelegten (vereinbarten) Ontologie sind diese Konzept-Terme die Repräsentanten der Äquivalenzklassen, zu denen sie gehören. Enthält eine Äquivalenzklasse keinen dieser Konzept-Terme, wird ein eindeutiger Repräsentant zufällig ausgewählt[5].

   (c) Den Repräsentanten der Äquivalenzklassen werden *Stelligkeiten* zugeordnet. Diese lassen die Verwendung als Prädikate der Propositionen zu. Die Stelligkeit regelt die zulässige Zahl der Argumente.
   Eine Proposition kann auch mehrere Stelligkeiten haben. Der folgende Schritt wird dann für jede Stelligkeit durchgeführt.

   (d) Jeder Argumentstelle einer Proposition wird eine Äquivalenzklasse (in Form ihres Repräsentanten) sowie eine *Fensterbreite* zugeordnet. Damit sind *prototypische Propositionen* vollständig definiert.

[5] Als Preis für die vollständige Automatisierung führt dies zu einer – streng genommmen unerwünschten – Erweiterung der Ontologie.

Die Fensterbreite gibt an, in welchem Umfeld beim Auftreten eines Prädikates in einem Dokument nach den Argumenten gesucht werden soll.

Auf diese Weise werden Propositionsprototypen der Form erzeugt, die die folgende Form haben: $predicate_i(arg_{i1}, \ldots, arg_{in})$, wobei jeweils n die Stelligkeit von $predicate_i$ ist. Bei diesen Prototypen stammen sowohl die Prädikate $predicate_i$ als auch die Argumente arg_{ij} aus einer kontrollierten Sprache, nämlich den Repräsentanten der Äquivalenzklassen, die durch den Thesaurus erzeugt wurden.

2. Propositionalisierung eines Text-Dokumentes

   Die folgenden Schritte werden sukzessive für alle Terme t_i des Dokumentes durchgeführt:

   (a) Bestimme die Äquivalenzklasse des Terms t_i. Das Prädikat der neuen Proposition besteht aus dem eindeutigen Vertreter dieser Klasse. Führe den folgenden Schritt für alle Propositionsprototypen durch, deren Prädikat gleich dem Prädikat der neuen Proposition ist.

   (b) Versuche für alle Stellen des Prototypen, im Rahmen der Fensterbreite um den Term t_i (und innerhalb einer festgelegten Analyseeinheit, z.B. Satz oder Abschnitt) Terme zu finden, die in die gleiche Äquivalenzklasse fallen wie der Repräsentant der entsprechenden Stelle.

   (c) Wenn dies gelingt, nimm den Propositionsprototypen in das Surrogat für das Dokument auf, ansonsten fahre beim nächsten Term fort.

   Das Propositionalisierungsverfahren versucht also, für alle Terme des Dokumentes, den Term und sein Umfeld auf einen Propositionsprototypen zu *matchen*. Dabei *matchen* Prädikate oder Argumente, wenn sie der gleichen Äquivalenzklasse angehören. Da nur eindeutige Repräsentanten benutzt werden, ist ein solcher *match* von Prototypen auf ein Textsegment – wenn er gelingt – immer eindeutig. Die so erzeugten Propositionen heißen *CL-Propositionen*.

Gibt man nicht für alle Stellen in den Propositionsprototypen Repräsentanten an, so muß man die Definition für einen *match* eines Argumentes in dem obigen Algorithmus erweitern. In diesem Fall führt *jeder* Term innerhalb des Textfensters zu einem *match* an der entsprechenden Stelle.

Man beachte, daß dieses Verfahren insbesondere die Anwendung eines Thesaurus auf die Oberflächenrepräsentation (Abschnitt 4.1.1) einschließt, wenn auch die 0-stelligen Propositionsprototypen mit betrachtet werden. Dann sind nämlich alle Äquivalenzklassen auch Propositionsprototypen, und jeder Term wird auf mindestens eine Proposition *gematched*. Im Sinne einer effizienten Verarbeitung sollten jedoch nicht zu viele Propositionen definiert werden, so daß es sinnvoll erscheint, nach Schritt (1b) eine Auswahl der Propositionen zu treffen, mit denen weiter verfahren werden soll.

Dieses Verfahren liefert somit eine (exklusiv der Vorverarbeitung) vollautomatische Propositionalisierung von Textdokumenten. Wie schon auf der Oberflächenebene können die Anfragen des Informationskonsumenten als Pseudo-Dokumente betrachtet und analog zu echten Dokumenten propositionalisiert werden. Die Berechnung der Relevanz eines Dokumentes bezüglich einer Anfrage geschieht nun derart, daß versucht wird, die propositionalisierte Anfrage mit den Propositionen des Dokumentes zu *matchen*. Gelingt dies, so wird ein Dokument für relevant erachtet, ansonsten nicht. Der *match*-Test ist sehr einfach, da alle Prädikate und Argumente aus Normalformen bestehen, so daß ein einfacher Zeichenkettenvergleich genügt. Läßt man auch unbestimmte Argumentstellen in der Anfrage zu, so verallgemeinert sich das *matching*-Verfahren wie oben beschrieben.

4.1.3 Situative Darstellung

Wie in Kapitel 3 beschrieben, sollen auf dieser Ebene "Bedeutungen" ganzer Dokumente repräsentiert werden. Dabei soll von konkreten Formulierungen sowohl auf der Wort- als auch auf der Satzebene abstrahiert und so die Semantik von Dokumenten erfaßt werden.
Dazu wird das LSA-Verfahren[6] [Deerwester et al. 90] eingesetzt, das davon ausgeht, daß es eine unterliegende oder "latente" Struktur des Musters der Wortbenutzung über die Dokumente hinweg gibt, die benutzt werden kann, um Ähnlichkeiten zwischen Dokumenten zu bestimmen. LSA nähert diese Strukturen durch statistische Analyse einer Stichprobe der Dokumente an.
Im folgenden wird beschrieben, wie Dokumente mit dem LSA-Verfahren repräsentiert werden und wie diese Technik im Rahmen des vorgeschlagenen Informationsassistenten eingesetzt wird.

[6]LSA steht für *latent semantic analysis*.

Latent Semantic Analysis

Ähnlich wie das Standard-Vektorraummodell, das für die Oberflächendarstellung genutzt wurde (vgl. Abschnitt 4.1.1), werden beim LSA-Verfahren Dokumente in einem hochdimensionalen Raum repräsentiert. Im Gegensatz zum Standard-Modell, dessen Dimensionalität durch die Anzahl der vorkommenden Terme bestimmt ist (oft mehrere zehn- oder hunderttausend Dimensionen), wird hier die Dimensionalität deutlich reduziert (auf z.B. 50–1500 Dimensionen) und somit die gewünschte Abstraktion erreicht (vgl. [Wong et al. 87]). Mathematische Basis dafür ist die *singular value decomposition*, eine Technik zur Matrixzerlegung, die eng mit der Faktoranalyse verwandt ist und für deren Ausführung effiziente Verfahren zur Verfügung stehen (siehe [Cullum et al. 85], [Berry et al. 93]).

Eine der Annahmen beim Standard-Vektorraummodell ist es, daß die einzelnen Dimensionen linear unabhängig voneinander sind. Dies ist jedoch normalerweise nicht der Fall. Vielmehr kommen Worte in bestimmten Kontexten häufiger vor als in anderen.[7] Das LSA-Verfahren erstellt einen semantischen Raum, dessen Dimensionen tatsächlich linear unabhängig voneinander sind, indem es statistische Zusammenhänge analysiert und somit versucht, die "latente Semantik" von Worten zu erfassen.

Ausgangspunkt für LSA ist eine $l \times k$ Term-Dokument-Matrix TDM, wie sie schon in der Oberflächendarstellung verwendet wurde. Durch *singular value decomposition* wird diese Matrix in das Produkt dreier Matrizen T_0, S_0 und D_0 zerlegt, so daß gilt:

$$TDM = T_0 \times S_0 \times {D_0}^T, \tag{4.7}$$

wobei T_0 eine orthonormale $l \times m$-Matrix und D_0 eine orthonormale $m \times k$-Matrix ist mit $m = rang\ TDM \leq min(l, k)$.
$S_0 = diag(\sigma_1, \sigma_2, \ldots, \sigma_m)$ ist eine Diagonalmatrix mit den *singular values* von TDM. Dies sind die nicht-negativen Quadratwurzeln der verallgemeinerten Eigenwerte von $TDM^T TDM$ [Berry et al. 95]. Das Verfahren liefert die *singular values* so, daß gilt $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_m$.

Behält man von S_0 nur die n größten Werte und die korrespondierenden Vektoren in T_0 und D_0, so erhält man einen auf n Dimensionen reduzierten Raum. Für diesen Raum TDM_n gilt, daß er TDM bezüglich des quadrati-

[7] In einer Menge von Dokumenten mit Texten aus der Informatik wird beispielsweise die Verwendung des Wortes "Expertensystem" nicht unabhängig von der Verwendung des Wortes "Regel" sein.

schen Fehlers und der neuen Dimensionalität optimal annähert.

$$TDM \approx TDM_n = T_n \times S_n \times D_n^T \tag{4.8}$$

Abbildung 4.2 zeigt schematisch die Zerlegung der Original-Term-Dokument-Matrix und den Übergang auf das reduzierte Modell durch Reduzierung der Dimensionen.

Während im Standard-Vektorraummodell jede Dimension einem Term entspricht, stehen die Dimensionen des reduzierten Raumes für *abstrakte Konzepte*. Diese haben a priori keine Bezeichnungen (etwa aus der Menge der Terme), die sie für den Benutzer verständlich machen würden. Die Wahl von n, der neuen Dimensionszahl, ist natürlich wesentlich für den Grad der Abstraktion.

Will man Dokumente in dem neuen, reduzierten Raum repräsentieren, muß man zwischen den Dokumenten unterscheiden, die schon in TDM direkt als Spaltenvektoren enthalten sind, da sie beim Aufbau des semantischen Raumes berücksichtigt wurden, und neuen Dokumenten, die nicht in der ursprünglichen Term-Dokument-Matrix vorkamen:

1. *alte* Dokumente:
   Für ein Dokument d_i wird der entsprechende Spaltenvektor aus TDM_n als Dokumentsurrogat benutzt. Dieser läßt sich aus den kleineren Matrizen D_n und S_n berechnen:

   $$TDM_{ni} = DS_i \tag{4.9}$$

   Da S Diagonalmatrix ist, ist der Raum DS lediglich eine gestreckte Version von D.

2. *neue* Dokumente:
   Um ein neues Dokument in dem reduzierten Raum zu repräsentieren, muß als Dokumentsurrogat ausgehend vom seinem Termvektor X ein passender *abstrakter Konzeptvektor* P_n gefunden werden. Die Berechnung sollte so geschehen, daß er für neue Dokumente, die genaue Kopien alter Dokumente sind, derselbe Vektor erzeugt wird wie für die alten[8]. Die folgende Definition leistet genau das Gewünschte:

   $$P_n = X^T T_n \tag{4.10}$$

   Dieser Prozeß, der für ein neues Dokument mit Termvektor X eine Repräsentation im reduzierten Raum unter der oben genannten Bedingung erstellt, wird als *Einfalten von* X, (*folding-in*, [Deerwester et al. 90])

[8] Dies sollte zumindest im perfekten Modell mit $n = m$ gelten.

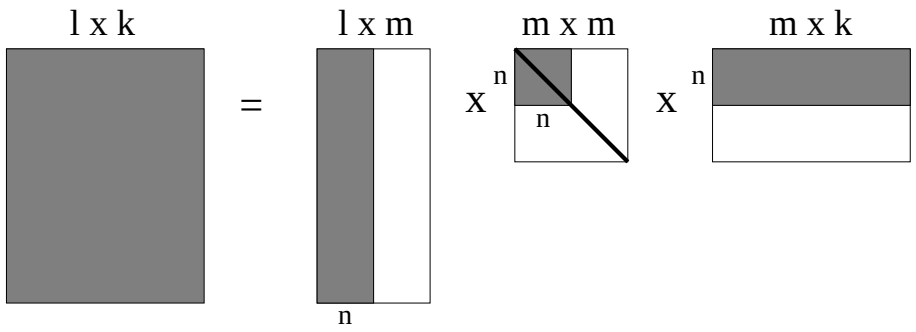

Abbildung 4.2: Schematische Darstellung der *singular value decomposition* der Term-Dokument-Matrix TDM. Indem nur die n ersten Dimensionen behalten werden (schattierter Bereich), erhält man eine Approximation TDM_n, die bezüglich des quadratischen Fehlers optimal ist.

bezeichnet. Das so berechnete Dokumentsurrogat P_n kann somit analog zu einem Spaltenvektor TDM_{ni} eines alten Dokumentes verwendet werden.

Man erhält damit sowohl für alte als auch für neue Dokumente als Surrogate Vektoren der Länge n, die man als *abstrakte Konzeptvektoren* der Dokumente im reduzierten Raum auffassen kann. Es könne also auch Dokumente repräsentiert werden können, die in der ursprünglichen Term-Dokument-Matrix, die zur Analyse bei der *singular value decomposition* benutzt wurde, nicht enthalten waren. Daher muß zur latenten semantischen Analyse nicht

der volle Raum der Dokumente verwendet werden, sondern es kann auch eine große Stichprobe Grundlage des Verfahrens sein.

Will man im neuen, reduzierten Raum zwei *Dokumente vergleichen*, kann man dazu – wie im Standard-Vektorraummodell – den Winkel zwischen den entsprechenden Vektoren, also wiederum das Kosinus-Maß (siehe Gleichung 4.5) verwenden.
Anfragen von Informationskonsumenten (etwa in Form von Schlüsselworten oder Beispieldokumenten) kann man auch als neue Dokumente (oder *Pseudo-Dokumente*) im reduzierten Raum repräsentieren, so daß als Relevanzmaß eines Dokumentes bezüglich der Anfrage wieder das Kosinusmaß Rel_{cos} (siehe Gleichung 4.6) benutzt werden kann.

Abbildung 4.3 zeigt noch einmal zusammenfassend, wie die Bestimmung der Relevanz eines Dokumentes bezüglich einer Informations-Konsumenten-Anfrage im Rahmen des LSA-Verfahrens von der Generierung des abstrakten Raumes über das Einfalten von Dokumenten und Anfragen bis zur Ähnlichkeitsberechnung in technischer Hinsicht abläuft.
Im Zentrum der situativen Ebene stehen die abstrakten Räume. Diese werden auf Basis einer Term-Dokument-Matrix von Beispieldokumenten mittels *singular value decomposition* erzeugt. Durch Reduzierung der Dimensionszahl erhält man einen abstrakteren, aber immer noch hochdimensionalen Raum, der durch drei Matrizen D_n, S_n und T_n repräsentiert wird. Jeder solche abstrakte Raum definiert einen Bezugsrahmen für die situative Darstellung von Dokumenten auf der situativen Ebene.
Wie im oberen Teil von Abbildung 4.3 dargestellt ist, können sowohl alte als auch neue Dokumente in dem abstrakten Raum dargestellt werden (einfalten, *folding-in*). Analog können auch Konsumentenanfragen (in Form von Schlüsselworten oder in Form von Beispieldokumenten, die als relevant betrachtet werden) in den abstrakten Raum eingefaltet werden. Somit erhält man abstrakte Dokumentvektoren bzw. abstrakte Anfragevektoren. Für jeden abstrakten Anfragevektor können nun die ähnlichsten Dokumente durch ein Ähnlichkeitsmaß auf Elementen des abstrakten Raumes (z.B. das Kosinus-Maß, Gleichung 4.5) bestimmt werden. Diese Ähnlichkeit wird schließlich als Indikator für die Relevanz interpretiert.

Einsatz abstrakter Räume im Inhaltsassistenten

Wenn man die abstrakte Bedeutung eines Textdokumentes ausdrücken will, dann ist eine solche Abstraktion nur hinsichtlich eines *Interpretationsrahmens* sinnvoll. Anders ausgedrückt: Die Bedeutung eines Textes gibt es nicht

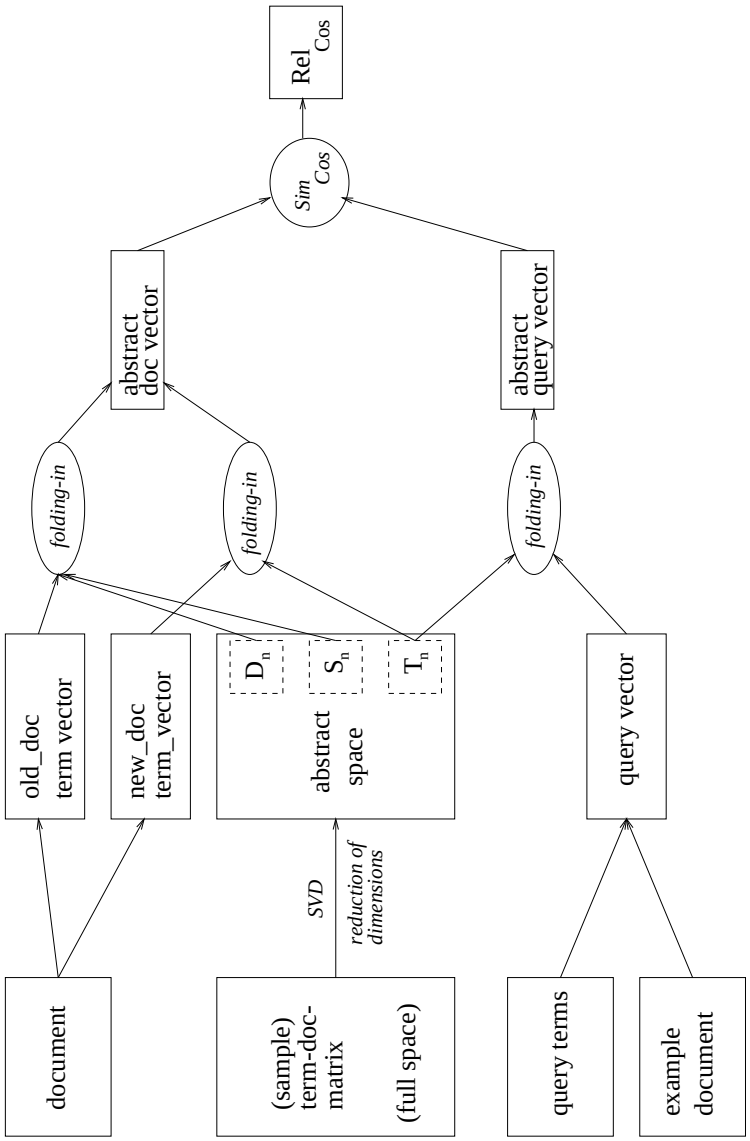

Abbildung 4.3: Einsatz des LSA-Verfahrens zur Bestimmung der Relevanz von Dokumenten für Konsumenten-Anfragen. Der Grad der Dimensionsreduktion bestimmt die Abstraktionsstärke.

an sich sondern nur vor dem Hintergrund einer "Theorie".
Wie oben dargestellt, benutzt das LSA-Verfahren als Interpretationsrahmen einen abstrakten Raum, der auf der Grundlage aller Dokumente oder einer Stichprobe aufgebaut wurde. Dieser Raum sollte so aufgebaut sein, daß er die latenten Strukturen über die Texte hinweg erfaßt. Mit der Wahl der Stich-

probe wird somit ein bestimmter Repräsentationsrahmen festgelegt. Benutzt man etwa eine Stichprobe von Texten einer bestimmten Kategorie oder Klasse von Dokumenten, dann werden *alle* Dokumente hinsichtlich dieses Rahmen interpretiert.

Eine Einschränkung auf *einen* Interpretationsrahmen für die Darstellung der Bedeutung von Dokumenten ist jedoch einer so offenen und dynamischen Aufgabe, wie sie in Kapitel 1.1 für das Wissensmanagement formuliert wurde, nicht angemessen. So kann etwa der Begriff "Neuronales Netz" innerhalb einer biologischen Dokumentensammlung eine völlig andere Interpretation haben als in einer Informatikbibliothek. Der hier vorgeschlagene Informationsassistent soll daher auf der situativen Ebene *mehrere abstrakte Räume* handhaben können. Jeder solche semantische Raum wird auf der Basis von Dokumente einer bestimmten Kategorie erzeugt. Dies ermöglicht den Aufbau von *Domänenmodellen* hinsichtlich verschiedener inhaltlicher *Aspekte* (oder Kategorien). Ein Domänenmodell beinhaltet somit die Strukturierung der Gesamtdomäne in einzelne Kategorien, die jeweils einen eigenen semantischen Raum erzeugen.

Als weiteren Freiheitsgrad, neben der Wahl der Stichprobe für die latente semantische Analyse, kann der *Abstraktionsgrad*, also die Stärke der Dimensionsreduzierung, betrachtet werden. In einem sehr abstrakten Raum können viele Dokumente kaum noch voneinander unterschieden werden (z.B. innerhalb der Menge aller Publikationen über maschinelles Lernen). Dies entspricht der Sichtweise eines Neulings auf diesem Gebiet, für den das angemessenste Dokument ein Lehrbuch wäre. Für den Experten ist jedoch ein Lehrbuch wenig hilfreich. Er kann zum Beispiel zwischen den Papieren unterscheiden, die sich mit symbolischen Verfahren befassen und denen, die konnektionistische Ansätze verfolgen. Für diesen Experten ist also ein semantischer Raum mit einer "höheren Auflösung" sinnvoller als ein sehr allgemeiner Raum.
Neben den verschiedenen Räumen für die Kategorien des Domänenmodells soll die situative Ebene für die einzelnen Kategorien deshalb auch verschiedene Abstraktionsgrade handhaben können, indem auf der Basis des gleichen Dokumentsamples eine unterschiedlich starke Dimensionsreduktion durchgeführt wird.

Die Dokumentsuche kann somit auf den einzelnen Benutzer besser angepaßt werden. Experten-Nutzer[9] können explizit angeben, auf welchen Rahmen sich

[9]Experten-Nutzer sind die Informationskonsumenten, die von der Inhaltsdomäne ein mentales Modell haben, das zum Modell der situativen Repräsentation des Informationsassistenten kompatibel ist.

ihre Anfragen beziehen. Das kann zu Beginn der Informationssuche geschehen, aber auch im weiteren Verlauf, wenn die ersten Ergebnisse verfeinert werden. Dies erlaubt auch eine Anpassung an die fortgeschrittene Expertise von Benutzern, die ursprünglich als "Domänen-Neulinge" die Informationssuche begannen. Auch der "richtige" Abstraktionsgrad für die gerade zu bearbeitende Aufgabe steht normalerweise nicht schon zu Beginn fest, so daß oftmals ein Wechsel während der Verfeinerung der Suche sinnvoll ist.

Eine relativ einfache Möglichkeit für den Aufbau der Domänenmodelle ist es, existierende Kategorisierungsstandards als Basis zu nehmen. Solche Standards sind beispielsweise die Dezimalklassifikation, wie sie in Bibliotheken benutzt wird, oder auch die Strukturierung der Dokumente in hierarchisch organisierten Ablagesystemen (etwa Verzeichnisbäumen o.ä.). Auch maschinelle Lernverfahren zur Bildung von Kategorien und zur Klassifikation großer Textmengen ([Nigam et al. 98]) können hier zum Einsatz kommen.
Das Domänenmodell für die situative Darstellung der Dokumente ergibt sich dann, indem für alle Kategorien mittels LSA ein eigener abstrakter Raum erzeugt wird. Dazu werden entweder alle Dokumente einer Kategorie oder auch nur Stichproben für das LSA-Verfahren herangezogen. Da LSA versucht, Strukturen über die gesamten Dokumente der Kategorie hinweg zu finden und dazu auch noch ein Abstraktionsschritt vorgenommen wird, ist dieses Vorgehen relativ tolerant bezüglich einzelner Dokumente, die einer falschen Kategorie zugeordnet wurden. Sie beeinflussen den Aufbau des jeweiligen Raumes nur unwesentlich.

Insgesamt ergibt sich für die Darstellung der Dokumente auf der situativen Ebene die in Abbildung 4.4 gezeigte Struktur.
Der allgemeinste Interpretationsrahmen ist der Raum, der auf der Basis *aller* Dokumente (oder einer Stichprobe davon) erstellt wurde und die niedrigste Dimensionenzahl aufweist. Durch Erhöhung der Dimensionenzahl erhält man speziellere Räume[10], die sich aber immer noch auf alle Dokumente beziehen. Die einzelnen Unterkategorien haben eigene semantische Räume, zu deren Erzeugung die entsprechend klassifizierten Dokumente herangezogen werden. Auch hier kann man durch unterschiedliche Reduzierung der Dimensionenzahl wieder für jede Kategorie Räume auf verschiedenen Abstraktionsniveaus erhalten.

Jedes Dokument der Dokumentenbasis kann nun gemäß der Beschreibung im vorhergehenden Abschnitt in einen (den allgemeinsten) oder mehrere Räume des Domänenmodells eingefaltet werden. Als Dokumentsurrogat erhält man

[10] Als Spezialfall würde sich – ohne Dimensionsreduktion – der volle Raum der Oberflächendarstellung ergeben.

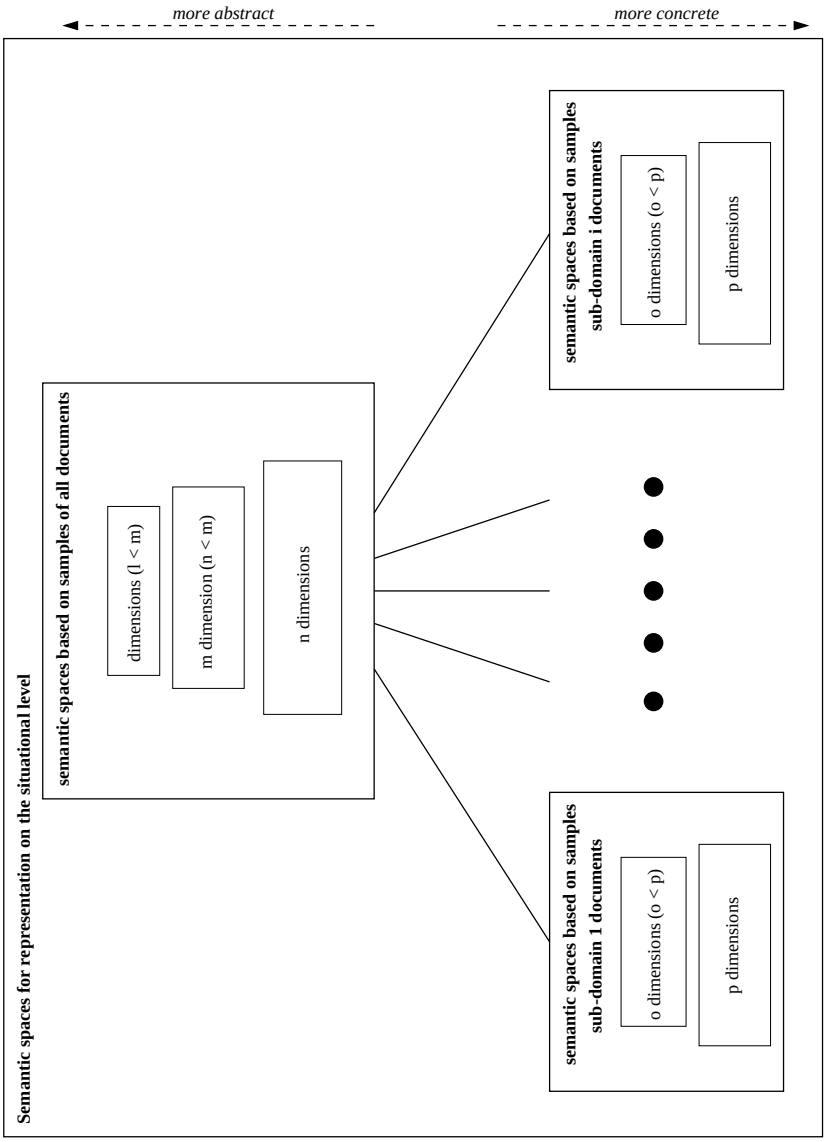

Abbildung 4.4: Struktur eines Domänenmodells auf der situativen Ebene mit mehreren semantischen Räumen. Die Inhaltsdomäne wird in verschiedene Teildomänen (Kategorien) eingeteilt, für die ein je eigener semantischer Raum existiert.

somit einen oder mehrere abstrakte Konzeptvektoren, die zur Relevanzbestimmung im jeweiligen Raum herangezogen werden können.
Aus Gründen der Effizienz sollte ein Dokument nicht in allzu vielen abstrakten Räumen repräsentiert werden. Ansonsten wird das Abspeichern der Sur-

rogate sowie die Berechnung der relevantesten Dokumente sehr teuer. Aus dem gleichen Grund sollte das gesamte Domänenmodell recht sparsam gehalten werden. Es könnte sich etwa auf einen Raum, zu dessen Erzeugung alle Dokumente benutzt wurden, mit zwei oder drei Abstraktionsniveaus und eine Ebene von Kategorien beschränken, wie es in Abbildung 4.4 angedeutet ist.

4.2 Integrationsprozesse: Reihenfolge der Präsentation

Die meisten Standard-Suchmaschinen für das *information retrieval* erzeugen eine nach der Relevanz (d.h. also meist der Ähnlichkeit zur Anfrage) sortierte Liste von Dokumenten. Dabei werden Probleme der Redundanz der Suchergebnisse jedoch meist nicht berücksichtigt. Da als Relevanzmaß die Ähnlichkeit von Dokumenten und Anfragen verwendet wird, erscheinen alle Dokumente, die zu einem sehr relevanten Dokument äußerst ähnlich sind, auch weit oben auf der Ergebnisliste. Ihr Informationsgewinn für den Konsumenten ist jedoch sehr klein. Weiter hinten in der Liste aufgeführte Dokumente, die möglicherweise ganz neue Aspekte hinsichtlich der Anfrage beinhalten, werden eventuell vom Konsumenten gar nicht mehr zur Kenntnis genommen. Der Informationsassistent soll daher in einer, dem *retrieval* nachgeschalteten Phase, die Dokumentliste so umsortieren, daß möglichst viel relevante Information in den weit vorne liegenden Dokumenten enthalten ist (*Integrationsphase*).
Im folgenden werden einige Verfahren zur Bestimmung der Präsentationsreihenfolge (*re-ranking*) vorgestellt. Diese sollten in eine Prozedur eingebettet sein, in der die für den Informationskonsumenten beste Reihenfolge in *Kooperation mit dem Konsumenten* gefunden werden kann (*conversation about documents*). Dadurch wird implizit oder explizit auch ein Benutzerfeedback erhoben, daß zur Verbesserung bei der Spezifikation zukünftiger Anfragen eingesetzt werden kann.

4.2.1 In der Literatur beschriebene Verfahren

"Maximal Marginal Relevance"

Die *maximal marginal relevance*-Metrik (MMR-Metrik) wurde von J. Carbonell und Kollegen im Rahmen des TIPSTER-Projektes entwickelt. Die

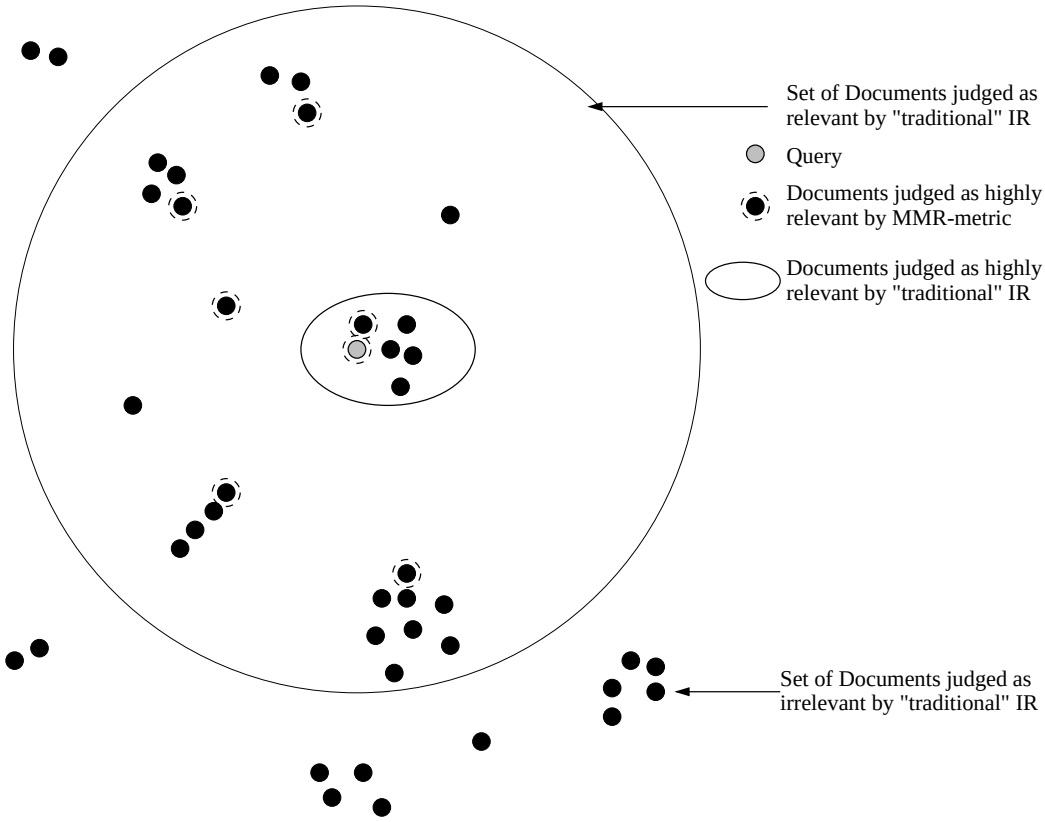

Abbildung 4.5: Prinzip der Bestimmung der Präsentationsreihenfolge mit der *maximal marginal relevance*-Metrik. Die "traditionelle" IR-Metrik könnte beispielsweise das Kosinus-Maß sein. (Bearbeitet nach [Carbonell 98])

Darstellung hier lehnt sich an [Carbonell 98] an.
Ausgangspunkt ist die Grundannahme, daß für gute *retrieval*-Ergebnisse nicht nur die Relevanz eines Dokumentes betrachtet werden sollte, sondern auch der Informationszugewinn. Um in der Liste der relevanten Dokumente, die den Konsumenten präsentiert wird, möglichst wenig redundante Dokumente weit vorne zu erhalten, wird die Reihenfolge der Dokumente mit der MMR-Metrik bestimmt. Diese versucht, in Clustern von ähnlichen Dokumenten jeweils nur wenige Dokumente als sehr relevant zu klassifizieren, die übrigen Dokumente des Clusters jedoch als weniger relevant zu bewerten. Die Annahme dabei ist also, daß Cluster von ähnlichen Dokumenten verschiedene Aspekte abdecken, so daß der Informationsgewinn größer ist, wenn Dokumente aus verschiedenen Clustern als hoch relevant betrachtet werden. Abbildung 4.5 veranschaulicht die Idee der *maximal marginal relevance*.

Die MMR-Metrik ist folgendermaßen definiert:

$$MMR(d_i, IR) = \underset{\substack{d_i \in IR \backslash D_R}}{argmax} \left[\lambda sim(d_i, q) - \left((1-\lambda) \max_{d_j \in D_R} (sim(d_i, d_j)) \right) \right] \tag{4.11}$$

Dabei ist

$q =$	query,
$IR =$	set of top k relevant documents of "traditional IR",
$D_R =$	set of already MMR ranked documents,
$\lambda \in$	$[0, 1]$,
$sim =$	Sim_{Cos}, (vgl. Gleichung 4.5).

Mit dieser Metrik kann also sukzessive die Reihenfolge der Präsentation der Dokumente ermittelt werden. Das jeweils nächste Dokument ist abhängig von der Anfrage *und* von den bereits von MMR als relevant betrachteten Dokumenten. Der Parameter λ bestimmt dabei, wie nah im Umfeld der Anfrage man sich dabei bewegen möchte. Er kann vom Informationskonsumenten verändert werden, so daß es möglich ist, näher auf das Umfeld der Anfrage zu fokussieren oder das Umfeld mehr auszuweiten.

"Connectionist Constraint Satisfaction"

In der Literatur des Textverstehens wird schon seit langem diskutiert, wie sich die Integration von neu gelesenen Textteilen mit dem vorher Gelesenen auf der Basis von Vorwissen sowie das Ziehen von Inferenzen modellieren läßt. Solche Modelle basieren oft auf Repräsentationen mit Propositionen und semantischen Netzen, wobei diese Begriffe meist etwas weiter gefaßt werden als oben beschrieben. Als Technik für diese Integration werden häufig *constraint satisfaction*-Prozesse eingesetzt, die sowohl symbolisch als auch konnektionistisch sein können ([Kintsch 98], [Schmalhofer 98], [Mross et al. 92], [Norvig 89]). Solche Verfahren sind insbesondere auch geeignet, um das Entstehen eines lokalen Diskursfokus innerhalb eines globalen Bezugsrahmens zu modellieren [van Elst et al. 97].
Dieses Problem in der Forschung zum Textverstehen ist strukturell ähnlich zum oben beschriebenen Problem der Integration der potentiell relevanten Dokumente zu einem den Bedürfnissen des Informationskonsumenten angepaßten Ganzen (also einer Art *Diskursfokus*). In beiden Fällen geht es insbesondere darum, Inkonsistenzen und Redundanzen zu entfernen und zu einer

kohärenten Sicht zu kommen.
Allerdings werden die Prozesse in der Textverstehensforschung oft nicht so detailliert spezifiziert, daß eine vollautomatische Abarbeitung möglich ist. Daher können die Ergebnisse dieses Forschungszweigs nicht einfach für den Inhaltsassistenten übernommen werden. Trotzdem dienen einige der dort beschriebenen Verfahren als Grundlage für die Integrationsprozesse des Inhaltsassistenten.

4.2.2 Der *ConRel*-Integrationsprozeß

In der bisherigen Betrachtung wurden die durch die Konstruktionsprozesse als relevant eingestuften Dokumente im wesentlichen als flach, d.h. als sequentielle Liste gesehen. Dies ist jedoch eine starke Vereinfachung, da diese Dokumente nicht nur zum nächsten mehr oder weniger relevanten Dokument in Beziehung stehen, sondern vielmehr jedes Dokument zu den anderen eine mehr oder weniger starke inhaltliche Verbindung aufweist. Diese Verbindungen gilt es für die Integration zu nutzen.

Die in dieser Arbeit entwickelte *ConRel*-Integration[11] faßt daher die in der Konstruktionsphase gefundenen Dokumente als ein *Wissensnetz* auf, in dem jedes Dokument mit einer Anzahl von anderen Dokumenten in direkter Verbindung stehen kann. Aufgabe ist dann, den Kontext, der durch den konkreten Informationskonsumenten mit seiner Anfrage gegeben ist, in dieses Wissensnetz zu integrieren. Umgekehrt formuliert: Der Kontext wird verwendet, um im Wissensnetz einen (Diskurs-)Fokus festzulegen. Die Dokumente, die innerhalb dieses Fokus liegen, werden dann als sehr relevant betrachtet und dem Konsumenten zuerst präsentiert.

Im folgenden werden die Eckpunkte der *ConRel*-Integration beschrieben. Auf diese Weise wird ein konnektionistisches Modell spezifiziert, das die Präsentationsreihenfolge der relevanten Dokumente festlegt.

- Dokumente werden im Wissensnetz durch *Aktivierungswerte* ("Grad der Zugehörigkeit" zum Fokus) sowie durch *Verbindungswerte* zu den anderen Dokumenten repräsentiert. Der Aktivierungswert heißt auch *Knotenstärke*, der Wert einer Verbindung zu einem anderen Dokument heißt auch *Kantenstärke* dieser Verbindung.

- Die Konsumentenanfrage wird analog in diesem Wissensnetz dargestellt, nämlich durch Knoten mit Aktivierungswerten und Kanten zu

[11] *ConRel* steht für "Context Relevance".

den übrigen Knoten.
Somit ist ein konnektionistisches Netzwerk definiert. In diesem kann die Aktivierung der Knoten über die Kanten zu den Nachbarknoten "fließen". Sind zwei Knoten durch eine Kante mit einem großen Stärkewert verbunden, kann "viel Aktivierung fließen", ansonsten wenig.

- Durch einen *spreading activation*-Prozeß wird dieses Fließen der Aktivierung von Knoten zu ihren Nachbarn über die Kanten simuliert. Dies geschieht solange, bis sich die Aktivierungswerte der Knoten nicht mehr ändern. Diesen Zustand des Netzes nennt man *Fixpunkt*. Während des *spreading activation*-Prozesses wird der Aktivierungswert der Knoten, die den Kontext (die Anfrage) repräsentieren, immer auf einem maximalen Wert gehalten, um der Tatsache Rechnung zu tragen, daß die Anfrage immer relevant ist, also im Fokus stehen sollte (*clamping*).

- Ist der Fixpunkt gefunden, wird die Knotenstärke benutzt, um den Grad der Zugehörigkeit der Dokumente zum Fokus und damit die Position in der Präsentationsreihenfolge zu bestimmen. Je stärker die Knotenstärke eines Dokumentes ist, um so mehr liegt es im Fokus und um so höher wird seine Relevanz bewertet.

Es wird deutlich, daß dieses Verfahren nur sinnvoll funktionieren kann, wenn es gelingt, in dem Wissensnetz die semantischen Relationen zwischen den Dokumenten widerzuspiegeln. Die Frage ist also, wie die einzelnen Stärkewerte im Netz zu wählen sind. Dabei genügt es, dies für die Kantenstärken zu spezifizieren, da sich zeigt, daß in dieser Art von konnektionistischen Modellen die Anfangsaktivierungen der Knoten kaum einen Einfluß auf den Fixpunkt haben.
Die Regeln für die Wahl der Kantenstärken werden (wegen der unterschiedlichen Repräsentationen) bezüglich der einzelnen Ebenen (Oberfläche, propositionale Ebene, situative Ebene) beschrieben. Das bedeutet, daß die Kantenstärken für jedes Dokument entsprechend der Ebene festgelegt werden, aufgrund derer das Dokument in der Konstruktionsphase als relevant erachtet wurde. Das Prinzip der Regeln ist jedoch übergreifend: Zwei Dokumentknoten werden durch eine starke Kante verbunden, wenn sie sehr ähnlich sind, sonst durch eine schwache Kante.

1. *Oberflächenebene*: Wie oben beschrieben wurde, bestehen Dokumentsurrogate auf der Oberflächenebene aus sehr langen Vektoren, die als Punkte in einem hochdimensionalen Raum aufgefaßt erden können. Es kann daher das gleiche Vektorähnlichkeitsmaß benutzt werden wie schon für die Relevanzberechnung (Kosinus-Maß).

Die Verbindungsstärke der Kante zwischen zwei Knoten, die die Dokumente d_i und d_j repräsentieren, beträgt also $Sim_{Cos}(d_i, d_j)$.

2. *Propositionale Ebene*: Hier bestehen die Dokumentsurrogate aus einer Liste von (CL)-Propositionen. Jeder solche Proposition wird durch einen Knoten im Wissensnetz vertreten. Für jedes Argument arg_ij einer Proposition $prop_i$ wird geprüft, ob sich eine Proposition $prop_k$ im Netz befindet, deren Prädikat gleich arg_ij ist, d.h. ob eine Proposition $prop_k$ von einer Proposition $prop_i$ umschlossen wird. Ist das der Fall, so werden die beiden entsprechenden Knoten mit einer starken Kante verbunden. Ansonsten beträgt die Kantenstärke Null, d.h. die beiden Knoten sind nicht verbunden. [12]

3. *Situative Ebene*: Wie auf der Oberflächenebene bestehen die Dokumentsurrogate der situativen Ebene aus Vektoren in einem hochdimensionalen Vektorraum (mit allerdings niedrigerer Dimensionalität). Die Festlegung der Kantenstärken kann also völlig analog erfolgen.

4. *Konsumentenanfragen*: Die Konsumentenanfragen werden, wie oben dargestellt, als (Pseudo-)Dokumente aufgefaßt. Jeden Teil der Anfrage wird daher auf der Ebene als (Pseudo-)Dokument repräsentiert, auf die er sich bezieht. Entsprechend erfolgt die Festlegung der Verbindungen zu den anderen Knoten im Wissensnetz wie für die jeweilige Ebene beschrieben.

Die Struktur des Wissensnetzes kann in einer Adjazenzmatrix $A = (a_{ij})$ gespeichert werden, in der der Eintrag a_{ij} die Kantenstärke zwischen Knoten i und Knoten j angibt, $1 \leq i, j \leq k =$ Anzahl der Knoten.
Die Aktivierungswerte können in einem Vektor der Länge k gespeichert werden. Der *spreading activation*-Prozeß wird durch wiederholte Multiplikation der Adjazenzmatrix mit dem Aktivierungsvektor simuliert, bis der Fixpunkt erreicht ist (siehe Abschnitt 5.6 im Kapitel "Implementierung").

Die *Festlegung der Präsentationsreihenfolge* geschieht im Anschluß an die Fixpunkt-Berechnung nach der Höhe der Aktivierungswerte der Dokumentknoten innerhalb der einzelnen Ebenen.
Auf der propositionalen Ebene kann jedes Dokument durch *mehrere* Knoten

[12] Es sind hier noch stärke Umschließungs- oder Teiltermordnungen auf Propositionen (etwa bzgl. der Argumente untereinander) denkbar. Aus Effizienzgründen wird darauf aber verzichtet. Der Wert für eine "starke Kante" muß a priori konstant gewählt werden. Der absolute Wert ist dabei nebensächlich, da auf dieser Ebene nur zwischen "starken" und "keinen" Verbindungen unterschieden wird.

im Wissensnetz vertreten sein, von denen sich jeder Knoten auf ein Dokumentsegment bezieht. Daher gibt es für die Präsentation zwei Möglichkeiten: 1) Die Werte der zu einem Dokument gehörenden Knoten wird addiert. Die Gesamtsumme bestimmt die Position in der Präsentation. 2) Es wird eine Art "Hyper-Dokument" erstellt, daß es den einzelnen hoch relevanten Dokumentsegmenten besteht. Dieses "Hyper-Dokument" stellt dann eine Art Zusammenfassung der auf der propositionalen Ebene relevanten Dokumente dar.

Mögliche Erweiterungen Bei der *ConRel*-Integration, wie sie oben beschrieben wurde, werden die einzelnen Repräsentationsebenen separat behandelt. Es handelt sich also im Prinzip um *drei* Wissensnetze, in die die auf die jeweilige Ebene bezogenen Teilen der Anfrage integriert werden. Es wäre aber auch denkbar, die drei Ebenen als *ein* Wissensnetz zu betrachten und die gesamte Anfrage in dieses Netz zu integrieren. Dazu müßte nur noch zusätzlich spezifiziert werden, wie Knoten verschiedener Ebenen miteinander verbunden werden sollen.
Im wesentlichen gibt es dafür zwei Alternativen: 1) Es wird wieder ein Ähnlichkeitsmaß auf den Knoten verwendet. Dazu können die (CL)-Propositionen (als Liste von Worten einer kontrollierten Sprache) wie die Dokumentsurrogate auf der Oberflächenebene und der situativen Ebene in einem hochdimensionalen semantischen Raum dargestellt werden, so daß wieder Vektorähnlichkeitsmaße benutzt werden können. 2) Die Kantenstärken werden (auf der Basis des gerade Beschriebenen) vom Informationskonsumenten mitbestimmt (etwa innerhalb einer grafischen Benutzeroberfläche). Der Konsument könnte somit steuern, auf welcher Ebene (mehr konkret oder mehr abstrakt) sich die Vorschläge des Inhaltsassistenten eher bewegen sollen.

4.2.3 Diskussion der Integrationsverfahren

Die bisher in der Literatur bekannten Verfahren, die die Reihenfolge, in der Dokumente präsentiert werden, nach dem Informationsgehalt wählen und Redundanzen vermeiden sollen, basieren auf dem Prinzip des *Scatter/Gather-Browsing*(Cutting92). Dieses zerlegt in der *Scatter*-Phase die Menge der Dokumente in eine vorgegebene Anzahl von Clustern und zeigt dann dem Konsumenten für jeden Cluster einen Repräsentanten an. Der Konsument wählt dann in der *Gather*-Phase relevante Cluster aus. Diese werden zusammengemischt und bilden die neue Dokumentenmenge, auf die das Verfahren von neuem angewendet werden kann. Die Cluster werden jeweils nach Ähnlichkeit der Dokumente gebildet. Wesentlich ist hier, daß immer die *Gesamtheit*

der Dokumente betrachtet wird (nämlich beim *clustering*).

Sowohl das *maximal marginal relevance*-Verfahren als auch die *ConRel*-Integration betrachten auch – wie das *Scatter/Gather-Browsing* aber im Gegensatz zu traditionellen *information retrieval*-Systemen – die Menge der potentiell relevanten Dokumente eher als in ihrer Topographie (d.h. in der Ebene, vgl. Abbildung 4.5) statt als flache Liste. Es wird also eine Struktur auf allen diesen Dokumenten betrachtet. Insofern sind diese Verfahren dem *Scatter/Gather-Browsing* verwandt[13].
Im Gegensatz zum *Scatter/Gather-Browsing* kommen aber sowohl das *maximal marginal relevance*-Verfahren als auch die *ConRel*-Integration ohne *explizite* Berechnung der Cluster aus. Diese beiden Verfahren unterscheiden sich insofern, daß beim *maximal marginal relevance*-Verfahren das Ergebnis inkrementell erzeugt wird (welches Dokument als nächstes gewählt wird hängt von den davor gewählten ab), während bei der *ConRel*-Integration das Ergebnis "auf einmal" erzeugt wird (d.h., bevor der Fixpunkt feststeht, gibt es keine Zwischenergebnisse, die benutzbar wären).

4.3 Überblick über die Architektur

Abbildung 4.6 zeigt noch einmal zusammenfassend die Architektur des kooperativen Informationsassistenten. Durch Konstruktions- und Integrationsprozesse wird ein gemeinsames Verständnis zwischen Informationsanbietern, Informationskonsumenten und Inhaltsassistent erzeugt.

In einem ersten Schritt werden die globale Interessen (z.B. Anforderungen des Archivsystems) und individuelle Interessen (der Anbieter und Konsumenten) koordiniert, indem aufgrund von Beispieldokumenten Räume zur Repräsentation von Dokumenten auf drei Ebenen (Oberflächenebene, propositionale Ebene, situative Ebene) erzeugt werden.
Aufgrund von Anfragen (adhoc-Anfragen oder langfristige Dokumentverteilungsaufträge) wird eine Menge von möglicherweise relevanten Dokumenten konstruiert. Die Bestimmung der Präsentationsreihenfolge geschieht kontextabhängig, d.h. auch in enger Kooperation mit dem Konsumenten. Dieser Schritt kann mehrfach ausgeführt werden, so daß sich das Ergebnis mehr und mehr den Bedürfnissen des Konsumenten anpaßt.
Die dabei gewonnenen Erkenntnisse können zur Verbesserung zukünftiger

[13] Dies spiegelt sich auch darin wider, daß die Verfahren zur Berechnung von Clustern wie auch die *ConRel*-Integration auf der Basis einer Adjazenzmatrix operieren, deren Elemente Ähnlichkeiten zwischen Dokumenten darstellen.

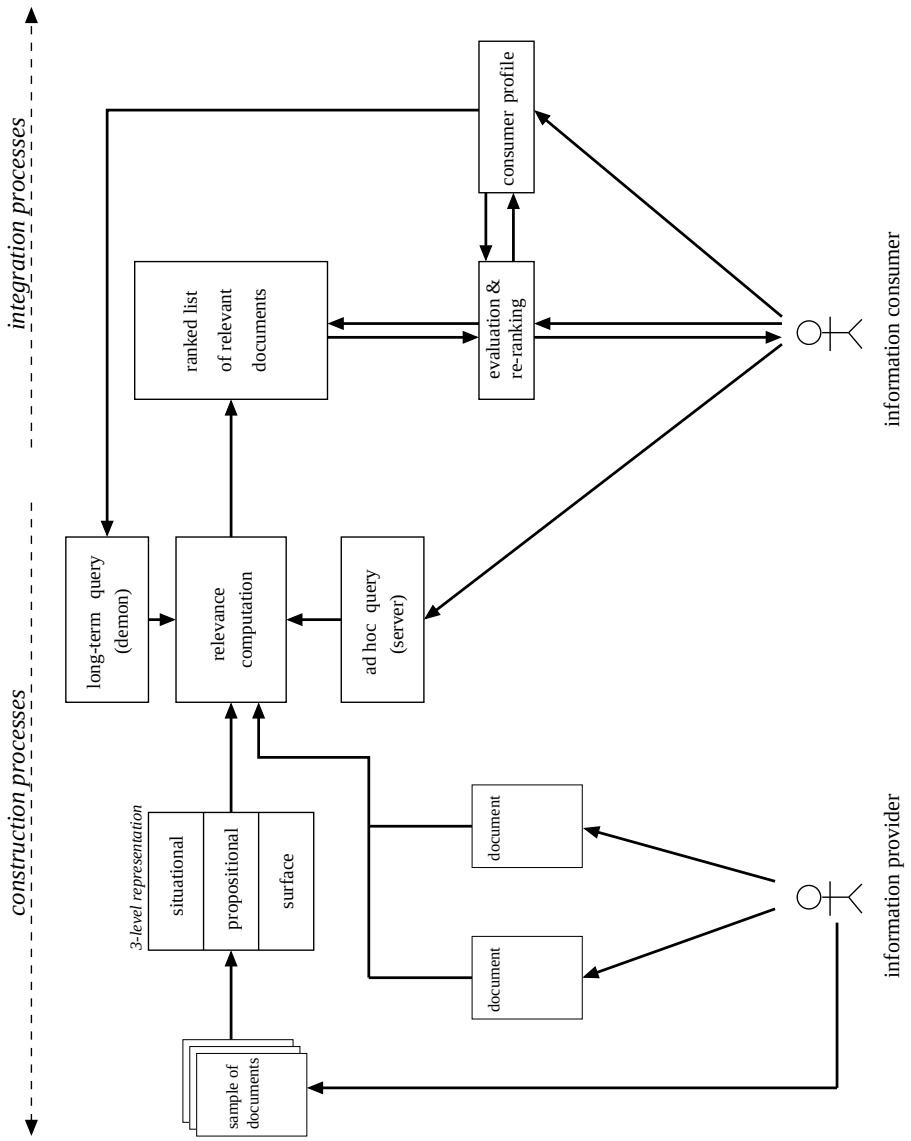

Abbildung 4.6: Statische Sicht auf die Architektur des Informationsassistenten

Anfragen eingesetzt werden. Das Benutzerprofil des Informationskonsumenten bleibt jedoch immer unter seiner direkten Kontrolle.

Im Rahmen der Forschungen zur Mensch-Computer-Interaktion wurde schon vor längerer Zeit erkannt, daß bei Computersystemen, die als Kommunikationsmedium dienen, solche Benutzeroberflächen besonders nützlich sind, die eine direkte Manipulation erlauben und die Effekte der Manipulationen so-

fort darstellen (vgl. z.B. [Malone et al. 89]). Die Informationssysteme, wie sie momentan etwa als Suchmaschinen im WWW im Einsatz sind, entsprechen dieser Forderung in keiner Weise. Vielmehr kehren sie mit ihren textbasierten Eingabefeldern, die eine mehr oder minder komplizierte Syntax zulassen, eher wieder in den Bereich kommandozeilenorientierter Benutzerschnittstellen zurück. Dies führt dazu, daß die Informationskonsumenten nur einen Bruchteil der Leistungsfähigkeit solcher Systeme tatsächlich nutzen.

Bei neueren Forschungen zu kooperativen Informationsassistenten wird daher immer mehr versucht, das WYSIWYG-Prinzip (*"What you see is what you get"*), das in anderen Bereichen wie dem *desktop publishing* längst Standard ist, in den Benutzerschnittstellen zu verwirklichen. So kann für das Navigieren in großen Ontologien die *hyperbolische Navigation* [Lamping et al. 96] verwendet werden. Diese Navigationsform verwendet eine Art "Fokus und Kontext"-Schema (*fisheye*-Schema), bei dem die Objekte im Fokus sehr genau, in einer hohen Auflösung, betrachtet werden können, der Kontext aber auch noch sichtbar bleibt, wenn auch nicht mehr so detailliert. Dadurch wird es ermöglicht, daß Objekte, die am Rand liegen, sehr schnell zum Fokus gemacht und genauer betrachtet werden können. Eine solche hyperbolische Navigation wird beispielsweise beim Ontobroker [Fensel et al. 98] eingesetzt.

Wie für die in dieser Arbeit betrachtete Informationslandschaft eine Benutzerschnittstelle aussehen könnte, die den oben aufgezeigten Forderungen Rechnung trägt, zeigt Abbildung 4.7.
Innerhalb dieser Schnittstelle kann der Integrationsprozess, wie er in Abschnitt 4.2.2 beschrieben wurde, durchgeführt werden. Im linken und im unteren Bereich der Abbildung wird der individuelle Kontext festgelegt. Eine *view* steht zum Beispiel für eine Gruppe von Informationskonsumenten mit bestimmten Privilegien. Die *personal preferences* repräsentieren eine Anfrage durch den Informationskonsumenten. Solche Anfragen bestehen aus *topics* (z.B. "*Group memory*"), die durch Schlüsselworte oder auch Beispieldokumente festgelegt sein können. Jedem *topic* ist ein Gewicht zugeordnet, das durch einen Schieberegler eingestellt wird. Für die hier gezeigte Anfrage ist das Thema *Knowledge sharing* sehr relevant, während das Thema *Creativity* nicht so wichtig ist. Im unteren Teil können die relevanten semantischen Räume ausgewählt werden. Weiterhin kann qualifiziert werden, ob die Dokumente sich eher "nahe am Zentrum" der Anfrage befinden sollen oder durchaus auch Randaspekte gewünscht sind (vgl. λ-Parameter beim *maximal marginal relevance*-Verfahren).

Auf der rechten Seite wird dann der Effekt der eingestellten Parameter sofort sichtbar, indem die Liste der relevanten Dokumente durch die *Con-*

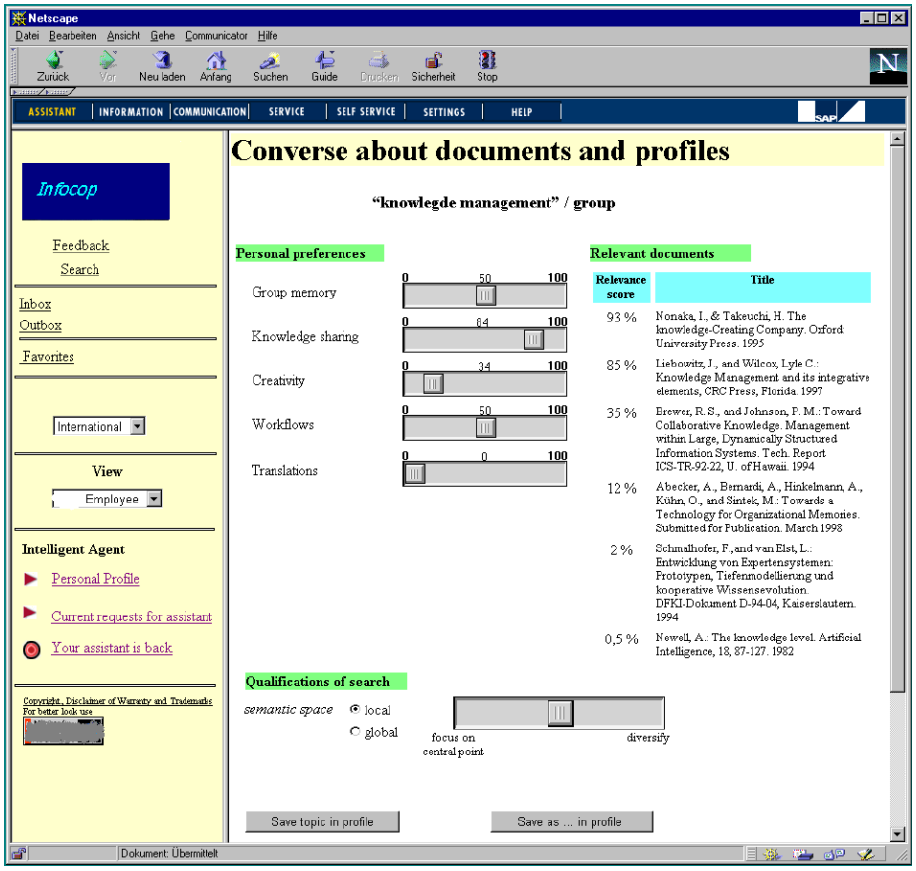

Abbildung 4.7: Benutzerschnittstelle zum Integrationsprozess. Über Schieberegler und Schalter kann die Sicht auf die Informationslandschaft leicht verändert werden. Die Effekte werden sofort sichtbar, indem sich die Reihenfolge in der Liste der relevanten Dokumente sofort ändert. (nach [Schmalhofer et al. 98a])

Rel-Integration entsprechend umsortiert wird[14]. Der Informationskonsument kann die hoch relevanten Dokumente evaluieren und über die Benutzerschnittstelle einen leicht veränderten Kontext festlegen. Den Effekt sieht wiederum sofort in der umsortierten Liste. Soll sich der Kontext stärker verändern (z.B. neue *topics*), so wird eine neue Konstruktionsphase nötig, für die eine ähnlich konzipierte Benutzerschnittstelle zur Verfügung gestellt wird.

[14]Da dazu *kein* globales Wissen über alle Dokumente oder andere Benutzer nötig ist, kann diese Integration lokal auf dem *client* des Informationskonsumenten berechnet werden.

4.4 Kooperative Wissensevolution

Intelligente Informationsassistenten wie der in dieser Arbeit beschriebene Inhaltsassistent können als Expertensysteme der dritten Generation angesehen werden. Während die Expertensysteme der ersten Generation (z.B. MYCIN [Shortliffe 76]) hauptsächlich prototypische (experimentelle) Entwicklungen waren, wurden in der zweiten Generation oft modellbasierte Entwicklungsmethoden eingesetzt (etwa die KADS-Methodologie [Schreiber et al. 93]). Expertensysteme der dritten Generation zeichnen sich dagegen dadurch aus, daß sie weniger "intelligente, autonome Programme" darstellen, als vielmehr "Wissensmedien" sein sollen [Stefik 86]. Als Mittel der Kommunikation zwischen menschlichen Benutzern und als Assistenzsysteme sind sie in hohem Maße kooperativ. Des weiteren versuchen sie, schon im Ansatz die Forderung nach Wiederverwendbarkeit von Elementen – allgemeiner also die Wartungsproblematik – zu berücksichtigen (*sharing and reuse effort* [Neches et al. 91])[15].

Die im Moment in der Entwicklung befindlichen Informationsassistenten integrieren die Wartungsproblematik besser in den Entwicklungsprozeß, als es bei früheren Expertensystem der Fall war. So gibt es schon lange Standardisierungsbemühungen wie etwa (z.B. HTML oder in neuerer Zeit XML [W3-Consortium 98]). Zwar gibt es immer noch Derivate und Uneinigkeiten über Details, wie sie schon früher dem *sharing and reuse-effort* entgegenstanden [Neches et al. 91], aber die meisten Systeme sehen Übersetzer zu Standards vor (wie etwa der Ontobroker [Fensel et al. 98]).

Ein wichtiger Ansatz für die Wartungsproblematik ist die *kooperative Wissensevolution* [Fischer et al. 94]. Nach dieser Sichtweise werden Computersystem und Benutzer als Partner bei der Problemlösung und beim Lernen in einer gemeinsamen Umgebung betrachtet. Dies geschieht innerhalb eines *life cycle*-Modells, daß aus drei Phasen besteht.
In einer ersten Phase (*seeding*) wird der Kern des Systems entwickelt. Dabei einigen sich die verschiedenen Benutzer auf einen gemeinsamen Rahmen, in dem das im System repräsentierte Wissen interpretiert wird, sowie auf gemeinsam akzeptierte Verfahren.
In der Phase des evolutionären Wachstums (*evolutionary growth*) wird das System benutzt. Dabei können dem System neue Wissenseinheiten hinzugefügt werden, es können Inkonsistenzen und Redundanzen entstehen oder bestimmte Abläufe können mit der Zeit nicht mehr für alle Benutzer akzeptabel sein, da auch sie durch die Verwendung des Systems lernen.

[15] Für eine Vertiefung dieser "geschichtlichen" Sichtweise auf Expertensysteme vergleiche [Schmalhofer et al. 94b].

Daher schließt sich in periodischen Zeiträumen eine dritte Phase, das *reseeding* an, in der der Bezugsrahmen von Repräsentationen und Prozessen neu festgelegt wird. Dabei können Redundanzen oder Inkonsistenzen entfernt werden, und es entsteht eine neue, den veränderten Bedingungen angepaßte Generation des Systems.
Durch dieses Wechselspiel von Wissensnutzung, Wissenswachstum und Reorganisation werden insbesondere auch die gemeinsamen Verantwortlichkeiten der beteiligten Akteure explizit in den Prozeß mit einbezogen werden.

In diesem Rahmen lassen sich die in dieser Arbeit spezifizierten Konstruktions- und Integrationsprozesse des Inhaltsassistenten den drei Phasen der kooperativen Wissensevolution zuordnen.

1. *seeding-Phase*:

   In dieser Phase wird der Rahmen für die Repräsentation von Dokumenten auf den drei Ebenen festgelegt. Das bedeutet auf der Oberflächenebene zum Beispiel, daß die konkreten Parameter für das allgemeine Standard-Vektorraum festgelegt werden. Dazu gehören die Gewichtungschemata, die Benutzung von Stoppwortlisten, die bestimmte Worte von der Verarbeitung ausschließen, und das Ähnlichkeitsmaß im Vektorraum (vgl. Kapitel 5).
   Auf der propositionalen Ebene werden die kontrollierte Sprache, der Thesaurus und die Propositionsprototypen für die CL-Propositionalisierung definiert.
   Das *seeding* auf der situativen Ebene beinhaltet drei Schritte. 1) Es wird die Struktur des Domänenmodells festgelegt. Dazu muß insbesondere spezifiziert werden, welche Teildomänen es gibt. Jedem semantischen Raum auf der situativen Ebene wird ein Abstraktionsgrad zugeordnet, der die Stärke der Dimensionsreduktion angibt. 2) Dann werden jeder Teildomäne Beispieldokumente zugeordnet. 3) Aufgrund der Beispieldokumente und des Abstraktionsgrades werden die abstrakten Räume mittels *latent semantic analysis* erzeugt.

   Durch dieses Vorgehen wird der Raum aller Dokumente auf eine bestimmte Weise strukturiert. Diese Struktur soll als Rahmen für eine Nutzungsphase im wesentlichen fest bleiben. Daher ist es sinnvoll, daß dieser Rahmen im Einverständnis der verschiedenen Benutzergruppen (Informationsanbietern, Informationskonsumenten, Administratoren) zustande kommt, da ansonsten nur eine geringe Qualität des *matchmaking*-Prozesses (vgl. Abschnitt 2.4) zu erwarten ist.[16]

[16] Natürlich ist dazu keine Absprache unter *allen einzelnen* beteiligten Individuen nötig

Weiterhin müssen auf der Anwendungsebene seitens der Konsumenten Profile und konkrete Anforderungen an den Inhaltsassistenten (z.B. *long-term queries*) spezifiziert werden. Informationsanbieter stellen außerdem die ersten Dokumente zur Verfügung.

2. *Evolutionäres Wachstum*:

   Nun kann eine Nutzungsphase des Informationsassistenten beginnen. Anbieter fügen dem System weitere Dokumente hinzu. Informationen werden abgerufen oder den Konsumenten vorgeschlagen. Diese können Rückmeldung über ihre Zufriedenheit mit den präsentierten Dokumenten geben und über die *ConRel*-Integration individuelle Präferenzen – z.B. für einzelne Themenbereiche oder den Grad der Abstraktion, der für sie angemessen ist – erarbeiten. Dadurch ändert sich dann das persönliche Benutzerprofil.
   Der Repräsentationsrahmen bleibt aber in dieser Zeit im wesentlichen unverändert. Das gilt insbesondere für das Domänenmodell auf der situativen Ebene und die Ontologie auf der propositionalen Ebene.

3. *Re-seeding*:

   Durch die fortwährende Nutzung können Inkonsistenzen oder Redundanzen entstehen. So können etwa *long-term queries* nicht mehr mit den persönlichen Konsumentenprofilen zusammenpassen. Auch kann die Domänenmodellierung oder die Propositionalisierung der neuen Menge von Dokumenten nicht mehr angemessen sein. Daher wird der Repräsentationsrahmen periodisch neu "verhandelt". Das bedeutet, daß in der Hauptsache die beim *seeding* beschriebenen Schritte zur Erzeugung der semantischen Räume wieder auszuführen sind. Dabei sollten die Erkenntnisse, die während Nutzung des Assistenten gewonnen wurden, mit in die Konstruktion des neuen Rahmens einfließen. Dies kann durch direkte Kommunikation zwischen den Benutzergruppen geschehen. Es sind aber auch maschinelle Lernverfahren denkbar, die Konsumentenanfragen und -profile oder Kategorisierungen durch Informationsanbieter analysieren – vorausgesetzt, die Nutzer wünschen und erlauben dies explizit.
   Insgesamt entsteht so ein *zweite Generation*, die den veränderten Bedingungen besser angepaßt ist, und es kann sich eine neue Nutzungsphase anschließen.

---

und möglich, da oft nicht jeder potentielle Anbieter und Konsument zu diesem Zeitpunkt feststeht. Auch können erhebliche Informationsasymmetrien etwa zwischen einem Domänenexperten, die Dokumente anbietet und einem Domänennovizen, der Informationen sucht, bestehen.

4.5 Zusammenfassung

In diesem Kapitel wurde ein Inhaltsassistent beschrieben, der das *matchmaking* zwischen Informationsanbieter und Informationskonsumenten unterstützen soll. Dazu wurden Konstruktionsprozesse dargestellt, die Dokument-Repräsentationen auf drei Ebenen in verschiedenen semantischen Räumen erzeugen können. Diese drei Ebenen (Oberflächenebene, propositionale Ebene, situative Ebene) unterscheiden sich insbesondere darin, wie groß die Abstraktion der erzeugten Dokumentsurrogate von den eigentlichen Dokumenten ist. Es wurde beschrieben, wie auf der Basis dieser Repräsentationen für Informationskonsumenten relevante Dokumente gefunden werden können und wie die Präsentation speziell auf die Anforderungen des Konsumenten angepaßt werden kann (Integrationsprozesse). Die so entstandene Architektur des Inhaltsassistenten wurde in ein *life cycle*-Modell (kooperative Wissensevolution) eingebettet, um so eine bessere Handhabung der Wartungsproblematik zu ermöglichen.
Im folgenden Kapitel wird auf der technischen Ebene die Implementierung der in diesem Kapitel beschriebenen Repräsentationen und Prozesse dargelegt.

Kapitel 5

Implementierung

Dieses Kapitel gibt einen Überblick über die im Rahmen dieser Arbeit durchgeführten Implementierungen. Es wurden C++-Module erstellt, die benutzt werden können, um einen konkreten Informationsassistenten zu erstellen, der im Rahmen der in Kapitel 2 (Abbildung 2.1) eingesetzt werden kann.
Die Struktur der Module ist weitgehend angelehnt an die einzelnen Repräsentationsebenen, wie sie in Kapitel 4 dargestellt wurden. Zusätzlich stehen noch Klassen zur Verwaltung von Dokumenten und Dokumentensammlungen zur Verfügung, die die Schnittstelle zum jeweiligen Dokumentenarchiv darstellen und entsprechend angepaßt werden müssen.[1]

5.1 *Documents*

In den *Documents*-Klassen ist die Verwaltung von Dokumenten, Sammlungen von Dokumenten sowie die Transformation von Dokumenten in die Multimenge ihrer Worte realisiert. Außerdem gibt es eine Klasse *PseudoDocuments*, die für die Verwaltung von Anfragen zuständig ist.

[1] Die Definition von konkreten Informationssuchaufträgen und ihre Abarbeitung, ad hoc oder im Rahmen längerfristiger Verteilungsaufträge, zum Beispiel innerhalb eines *Client/Server*-Systems war nicht Gegenstand dieser Arbeit. Ihre Implementierung sollte im Rahmen des jeweils verwendeten Informationsmanagementssystems erfolgen, da sie stark von den dort vorhandenen Zustellungs- und Abarbeitungsmechanismen abhängig ist.

Documents

Eine *Documents*-Variable besteht aus dem Namen des Dokuments, einer Identifikationsnummer sowie einem Handle für das reale Dokument. Der Name dient als eindeutige Adresse für das Dokument. Dies kann beispielsweise der vollständige Dateiname oder eine Adresse im WWW in Form eines URL (*Uniform Resource Locator*) sein. Mit den Methoden `open` und `close` wird der Handle an das konkrete Dokument gebunden.

5.1.1 DocumentList

Mittels der *DocumentList*-Klasse können mehrere Dokumente in einer einheitlichen Struktur gehandhabt werden. Der Zugriff auf die Dokumente kann über einen Iterator erfolgen. Es wird jeweils die Identifikationsnummer und der Name des aktuellen Dokuments zurückgeliefert. Weiterhin kann über die Identifikationsnummer der zugehörige Dokumentenname gefunden werden (`string DocName(int id)`). Dies erlaubt weitergehenden Klassen, Dokumente nur noch mit ihren Identifikationsnummern zu bezeichnen. Es stehen Methoden zum Laden und Abspeichern von Dokumentlisten zur Verfügung.

5.1.2 Document2Words

In der *Document2Words*-Klasse können Dokumente als Multimenge ihrer Worte (*bag of words*) aufgefaßt werden. Solche Implementierungen (wie zum Beispiel die libbow-Bibliothek [McCallum 97]) sind Basis vieler wortbasierter *information retrieval*-Systeme. Wesentlich ist die `CountWords`-Methode, die auf der Basis eines Lexers (vgl. Abschnitt 5.7.2) das Dokument in seine Worte zerlegt und für diese Worte in einer Variablen der Klasse *Word2Wordcount* die entsprechenden Häufigkeiten zählt. Die Klasse *Word2Wordcount* verwaltet solche Abbildungen von Worten auf ihre Häufigkeitswerte.
Ist eine Stoppwortliste (vgl. Abschnitt 5.2.1) definiert, können die dort spezifizierten Worte von der Zählung ausgenommen werden.

5.1.3 PseudoDocuments

Anfragen in Form von Schlüsselwortlisten werden als Pseudo-Dokumente bezeichnet. Da es für diese Pseudo-Dokumente keine Häufigkeitswerte gibt, werden statt dessen Gewichte benutzt. Variablen der `PseudoDocuments`-Klasse handhaben Listen von Schlüsselworten und zugehörigen Gewichten.

Sie können damit als Vektoren im hochdimensionalen Raum, der von allen Worten aufgespannt wird, betrachtet werden (siehe Abschnitt 4.1.1). Die `ComputeSVDRep`-Methode realisiert das Einfalten in einen abstrakten Raum mit niedrigerer Dimensionalität (siehe Abschnitt 2). Entsprechend erhält die Methode den Raum, auf den sich dieser Abstraktionsschritt beziehen soll, als Parameter.
Das folgende Programmsegment implementiert damit die Definition für den abstrakten Anfragevektor ($P_n = X^T T_n$, Gleichung 4.10), wobei

- `X` der gewichtete Vektor der Anfrage,
- `svd.Dimensionen` die Dimensionszahl des reduzierten Raumes und
- `svd.T` die T_n-Matrix aus der SVD-Zerlegung $TDM_n = T_n \times S_n \times D_n^T$ mit reduzierter Dimensionszahl ist.

```
ComputeSVDRep (SVDspace &svd)
{
  for (red_dim = 1; red_dim <= svd.Dimensionen; red_dim++)
  {
    summe = 0;
    for (dim = 1; dim <= number_of_terms; dim++)
      summe += X[dim] * svd.T[dim][red_dim];
    P[red_dim] = summe;
  }
}
```

Der Vektor `P` enthält zum Schluß den abstrakten (oder eingefalteten) Anfragevektor, der für die Relevanzberechnung im abstrakten Raum benutzt werden kann. Aus Effizienzgründen wird für den konkreten Anfragevektor nicht die volle Abbildung der Terme auf ihre Gewichte gespeichert, sondern nur die Gewichte, die ungleich Null sind. Die Komplexität reduziert sich damit auf $O($`(*svd).Dimensionen *` *Anzahl Terme in der Anfrage*$)$.

5.2 *Terms*

5.2.1 StopTerms

Stoppworte sind solche Terme, die beim Erzeugen der Term-Dokument-Matrix unberücksichtigt bleiben sollen. Dies kann aus zwei Gründen sinnvoll sein:

- Die Terme tragen wenig Bedeutung (z.B. Artikel).
- Die Terme sind statistisch nicht relevant, weil sie in nahezu jedem Dokument vorkommen (z.B. das Geschäftsfeld einer Firma in ihren Pressetexten oder Floskeln in Briefen). Solche Terme diskriminieren nicht zwischen den Dokumenten.

Um die Speicher- und Berechnungseffizienz zu erhöhen, wird daher oft auf die Weiterverarbeitung solcher Terme verzichtet.
Da Stoppworte des zweiten Typs oft von der konkreten Anwendung abhängen, ist hier die Handhabung verschiedener domänenspezifischer Stoppwortlisten vorgesehen.
Während im allgemeinen beim Einsatz des LSA-Verfahrens vorgeschlagen wird, die Terme nicht weiter vorzuverarbeiten (z.B. Reduktion auf die Stammform), wurden Stoppwortlisten des ersten Typs bei fast allen Evaluationen benutzt ([Foltz et al. 92], [Dumais 95]).
In der vorliegenden Implementierung ist eine 571 Einträge große Stoppwortliste mit englischen Artikeln, Pronomen etc. enthalten (`english.stop`).

5.2.2 TermList

Die Klasse *TermList* realisiert Funktionen zur Verwaltung von Zuordnungen von Worten auf Identifikationsnummern (`ID`) sowie für den effizienten Zugriff darauf. Somit können darauf aufbauende Module Worte sparsam über ihre `ID` speichern und referenzieren.

5.3 *verbatim_level*

Hier werden die Funktionalitäten zur Repräsentation von Dokumenten auf der Oberflächenebene, wie sie in Abschnitt 4.1.1 spezifiziert wurden, realisiert.

Term-Dokument-Matrizen sind das zentrale Element der Oberflächenrepräsentation. Sie stellen die Dokumente als Vektoren in einem hochdimensionalen Raum dar, der von den Termen aufgespannt wird. Da die meisten Terme normalerweise nur in wenigen Dokumenten vorkommen, sind die Term-Dokument-Matrizen oft nur dünn besetzt, so daß der Einsatz entsprechend optimierter Speicherungs-, Zugriffs- und Manipulationsverfahren angebracht ist.
Die Klasse *term_doc_sparse* stellt Methoden zur Verfügung, um Dokumente

mit oder ohne Berücksichtigung von Stoppworten zu einer Term-Dokument-Matrix hinzuzufügen. Um Ähnlichkeiten von Dokumenten in diesem Raum zu bestimmen, gibt es die beiden Methoden `DocumentsCosinus` (Parameter: zwei Dokumentindizes) sowie `DocPseudoDocCosinus` (Parameter: ein Pseudo-Dokument und ein Dokumentindex). Die beiden Methoden berechnen den Winkel zwischen zwei Dokumentvektoren (Gleichung 4.5) bzw. einem Dokumentvektor und einem Pseudodokumentvektor (also einer Anfrage, Gleichung 4.6). Dieser Winkel kann dann als Relevanzmaß interpretiert werden.

Die in Abschnitt 4.1.1 beschriebene Gewichtung der Terme mit ihren Häufigkeiten hat einige praktische Nachteile wie zum Beispiel, daß Vergleiche sehr unterschiedlich langer Texte nicht adäquat durchgeführt werden. Daher wird die Gewichtung der Einträge in der Term-Dokument-Matrix meist etwas allgemeiner formuliert als in Gleichung 4.3. Dazu führt man eine *lokale Gewichtsfunktion* $L(v_i, d_j)$ ein, die den Anteil des Gewichtes von Term v_i *innerhalb* des Dokumentes d_j im Gesamtgewicht bestimmt (das könnte die schon beschriebene Termhäufigkeit tf_{ij} sein), sowie eine *globale Gewichtsfunktion* $G(v_i)$, die einen Anteil für das Gewicht des Terms v_i über die Menge aller Dokumente hinweg bestimmt [Salton et al. 88]. Die Einträge der Term-Dokument-Matrix werden daher folgendermaßen definiert:

$$tdm_{ij} = L(v_i, d_j) * G(v_i) \tag{5.1}$$

Die Tabellen 5.3 und 5.3 zeigen lokale und globale Gewichtsfunktionen, die in praktischen Systemen oft angewendet werden [Dumais 91].

Die bisherigen Untersuchungen – etwa in den *Text Retrieval Conferences* TREC – haben gezeigt, daß sich mit solchen Gewichtungen die *retrieval*-Ergebnisse deutlich verbessern lassen ([Harman 95], [Voorhees et al. 97]).

5.4 *propositional_level*

Eine der Hauptaufgaben auf der propositionalen Ebene ist es, *Normalformen* von Worten zu finden. Dazu wird ein Thesaurus eingesetzt, der Äquivalenzklassen auf Worten definiert. Im Rahmen dieser Arbeit stand der im FAKT-Projekt entwickelte Thesaurusgenerator TREX [Kühn et al. 98] zur Verfügung. TREX verwendet für die Thesauruserzeugung ähnliche Abstraktionstechniken, wie sie in dieser Arbeit auf der situativen Ebene benutzt wurden (*latent semantic analysis* [Deerwester et al. 90]). Die Auswahl des eindeutigen Repräsentanten für jede Äquivalenzklasse geschieht nach den in Abschnitt 4.1.2 beschriebenen Regeln.

Tabelle 5.1: Lokale Gewichtungsfunktionen

Binary	$L(v_i, d_j) = \begin{cases} 0 & : \quad tf(v_i, d_j) = 0 \\ 1 & : \quad tf(v_i, d_j) \geq 1 \end{cases}$
Term-frequency	$L(v_i, d_j) = tf(v_i, d_j)$
Log	$L(v_i, d_j) = log_2(tf(v_i, d_j) + 1)$

definition:
$tf(v_i, d_j)$ = frequency of term v_i in document d_j

Tabelle 5.2: Globale Gewichtungsfunktionen

Idf	$G(v_i) = \log_2 \left(\frac{ndocs}{df(v_i)} \right)$
Entropy	$G(v_i) = 1 + \sum_j \frac{p_{ij} \log_2(p_{ij})}{\log_2(ndocs)}$ with $p_{ij} = \frac{tf(v_i, d_j)}{gf_i}$

definitions:
$tf(v_i, d_j)$ = frequency of term v_i in document d_j,
$gf(v_i)$ = frequency of term v_i in all documents,
$df(v_i)$ = number of documents in which term v_i appears,
$ndocs$ = number of documents

5.5 *situational_level*

Wie in Abschnitt 4.1.3 beschrieben wurde, können auf der situativen Ebene *mehrere* abstrakte semantische Räume definiert werden. Die Struktur dieser Räume wird als Domänenmodell bezeichnet.
Zur Verwaltung eines solchen abstrakten semantischen Raumes kann eine Variable der *SVDspace*-Klasse verwendet werden. Es stehen Methoden zur Verfügung, um alte Dokumente, neue Dokumente und Pseudo-Dokumente in einem solchen semantischen Raum mittels des Kosinus-Maßes miteinander zu vergleichen (`DocumentsCosinus` und `DocumentPseudoDocumentCosinus`).

5.6 *integration*

Implementierungen eines Algorithmus zur Simulation des *spreading activation*-Prozesses für die *ConRel*-Integration wurde schon an anderer Stelle ausführlich beschrieben ([Mross et al. 92], [van Elst et al. 97]). Daher wird das Verfahren hier nur in Grundzügen dargestellt.

Wie schon in Abschnitt 4.2.2 gezeigt wurde, kann die Struktur des Wissensnetzes in einer Adjazenzmatrix $A = (a_{ij})$ gespeichert werden, wobei a_{ij} die Verbindungsstärke zwischen Knoten i und Knoten j angibt. Da nur ungerichtete Verbindungen angenommen werden, ist die Matrix symmetrisch: $A_{ij} = A_{ji}$.
Die Aktivierungsstärken der einzelnen Knoten werden in einem Vektor $X = (x_i)$ gespeichert.

Der *spreading activation*-Prozeß $\Phi(X)$ läßt sich durch die Komposition dreier Teilabbildungen beschreiben ($\Phi(X) = \Phi_3(\Phi_2(\Phi_1(X)))$). Dazu sei n die Anzahl der Knoten im Netz. A ist also eine $n \times n$-Matrix und X ein n-Vektor.

1. $\Phi_1(X) = A \times X$
2. $\Phi_2(X) = (max(x_1, 0), \ldots, max(x_n, 0))$
3. $\Phi_3(X) = \left(\frac{x_1}{MAX}, \ldots, \frac{x_n}{MAX}\right)$, wobei MAX das Maximum der x_i ist.

Die Multiplikation der Verbindungsmatrix mit dem Aktivierungsvektor in Φ_1 simuliert den eigentlichen Fluß der Aktivierung (parallel für alle Knoten). Wegen Φ_2 können keine negativen Aktivierungswerte vorkommen. Φ_3 schließlich *normalisiert* den Aktivierungsvektor, so daß nur noch Werte zwischen Null und Eins vorkommen können. Sind nicht alle Aktivierungswerte Null, so ist Φ also wohldefiniert.

Iterativ wird nun der Aktivierungsfluß durch das Wissensnetz simuliert:

$\Phi^m(X) = \Phi(\Phi^{m-1}(X)) \forall m \geq 1,\ \Phi^0(X) = X.$

Dieses wird solange durchgeführt, bis nahezu keine Veränderung im Aktivierungsvektor mehr auftritt (die durchschnittliche Änderung der Aktivierungsstärken unter eine feste Schwelle, z.B. 0.0001, absinkt). Das bedeutet, daß $\Phi^m(X) \approx \Phi^{m-1}(X)$ ist. In diesem Falle fließt nahezu keine Aktivierung mehr im Netz. Es wurde ein Gleichgewicht im Wissensnetz gefunden (thermodynamisch: ein Energieminimum; mathematisch: eine Näherung für einen Fixpunkt von Φ). Die Bedingungen, unter denen es für den beschriebenen Netzwerktyp einen eindeutigen Fixpunkt gibt, der mit dieser Prozedur gefunden wird, wurden in [Rodenhausen 92] untersucht. Das vorgeschlagene Verfahren zum Aufbau des Wissensnetzes berücksichtigt diese Bedingungen, so daß die Integration der potentiell relevanten Dokumente mit dem speziellen Kontext (der Konsumentenanfrage) in jedem Falle zu einem definierten Ergebnis führt.

5.7 *utilities*

Die *utilities* fassen einige Hilfsmittel zusammen, die von den übrigen Klassen benötigt werden. Die beiden wichtigsten Klassen sind *las2* und *Lexer*.

5.7.1 *las2*

Die *las2*-Klasse führt die Zerlegung (*singular value decomposition*) der Term-Dokument-Matrizen in drei Teilmatrizen gemäß Gleichung 4.7 durch. Dazu wird die SVDPACKC-Implementierung [Berry et al. 93] benutzt, an der im Rahmen des FAKT-Projektes [Kühn et al. 98] einige kleine Anpassungen vorgenommen wurden. Das Verfahren kann mittels einer Konfigurationsdatei parameterisiert werden.

5.7.2 *Lexer*

Innerhalb eines Lexers wird definiert, welche Einheiten eines Textdokumentes als *Worte* oder *Terme* betrachtet werden. Der Lexer bietet damit die Möglichkeit, ein Dokument sukzessive in *token* zu zerlegen.
Im Rahmen dieser Arbeit wurde ein *Html_and_WhitespaceLexer* implementiert. Dieser betrachtet zum einen HTML-Markierungen als Token. Das bedeutet, daß alle Zeichen, die innerhalb eines < >-Klammerpaares stehen, zu

einem Token zusammengefaßt werden. Außerhalb von HTML-Markierungen werden solche Zeichenketten als Worte betrachtet, die durch Leerzeichen oder Satzzeichen voneinander abgegrenzt sind.

5.8 Praktische Einsatzmöglichkeiten

Eine genaue Evaluation der Implementierung gehört zwar nicht zur Aufgabenstellung dieser Arbeit, trotzdem wurden Teile der oben beschriebenen Programme auf konkrete Datensätze angewendet.

Wie in Kapitel 2 gezeigt wurde, ist für den *matchmaking*-Prozeß zwischen Studienplatzsuchern (Studenten) und Studienplatzanbietern (Universitäten) die Bearbeitung der Wissensmanagement-Problematik (*richtige Dokumente* zur *richtigen Zeit* an die *richtigen Personen*) wesentlich.
Als Datenmaterial wurde der *"4 Universities"*-Satz ausgewählt, der in der Arbeitsgruppe von Prof. Tom Mitchell im *WebKB*-Projekt unter anderem zum Test von Klassifikationsverfahren eingesetzt wird (vgl. [Craven et al. 98], [Nigam et al. 98]). Diese Materialien stehen im WWW zur Verfügung.[2] Enthalten sind 8282 WWW-Seiten von Informatik-Fachbereichen verschiedener amerikanischer Universitäten. Diese liegen nach sieben Kategorien klassifiziert vor: 1. *student* (1641 Seiten), 2. *faculty* (1124 Seiten), 3. *staff* (137 Seiten), 4. *department* (182 Seiten), 5. *course* (930 Seiten), 6. *project* (504 Seiten) und 7. *other* (3764).
Diese Kategorisierung wurde zur Erzeugung eines Domänenmodells benutzt, wie es in Abschnitt 4.1.3 vorgeschlagen wurde. In [Craven et al. 98] wird eine Ontologie vorgestellt, die als Ansatzpunkt für die Darstellung der WWW-Seiten mit einer kontrollierten Konzeptsprache benutzt werden kann. Diese Ontologie enthält Konzepte wie zum Beispiel "*Department.Of.Person*", "*Advisor.Of.Student*" oder "*Members.Of.Project*".
Für die Seiten des *"4 Universities"*-Datenmaterials wurden sowohl Oberflächenrepräsentationen als auch verschiedene abstrakte semantische Räume erzeugt. Die Durchführung einiger Anfragen an einzelne semantische Räume läßt vermuten, daß in der Tat adäquatere Resultate erzeugt werden, als es bei einer reinen oberflächen-orientierten Suche der Fall ist.

[2] http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data

Kapitel 6

Diskussion

Zusammenfassende Charakterisierung des Inhaltsassistenten

In dieser Arbeit wurde ein Informationsassistent zum gemeinsamen Verstehen von Textdokumenten (Inhaltsassistent) entwickelt. Die Hauptaufgabe des Inhaltsassistenten ist es, Repräsentationen zu erzeugen, die mit denen der Benutzers kompatibel sind. Da verschiedene Benutzer unterschiedliche Zielsetzungen und Kenntnisstände haben können, ist es besonders wichtig, flexibel mit den Repräsentationen umgehen zu können.[1]
Diese Fähigkeit wurde erreicht, indem a) mehrere Repräsentationsebenen für Textdokumente spezifiziert wurden und b) für die Arbeit des Inhaltsassistenten zwei Phasen festgelegt wurden. Während die Konstruktionsphase noch recht allgemein eine Liste von möglicherweise relevanten Dokumenten erzeugt, wird in der Integrationsphase versucht, das Ergebnis der Informationssuche den speziellen Anforderungen des Informationskonsumenten anzupassen.

Die Beschreibung des Inhaltsassistenten erfolgte in drei Schritten. Mit dem Lösungsansatz (Kapitel 3) wurden die Eckpfeiler des vorgeschlagenen Assistenten recht allgemein beschrieben. Es wurden drei Ebenen der Repräsentation mit ihren verschiedenen Zielsetzungen sowie die Einteilung in Konstruktionsprozesse und Integrationsprozesse mit ihren verschiedenen Funktionen dargestellt. In Kapitel 4 wurden dann die einzelnen Techniken genauer spezifiziert. Dazu kamen sowohl bekannte Standard-Verfahren als auch aktuelle Forschungsergebnisse und im Rahmen dieser Arbeit entwickelte Algorithmen

[1] Dies gilt umso mehr, als gerade zwischen Informationsanbietern und Informationskonsumenten recht große Asymmetrien bezüglich ihrer Sichtweise auf die Inhaltsdomäne bestehen.

zum Einsatz. Konkretere Implementierungsdetails wurden in Kapitel 5 beschrieben.
Durch diese Struktur ist möglich, auf den einzelnen Ebenen Verbesserungen einzuführen, wenn die Forschung neuere Erkenntnisse und Möglichkeiten bringt, ohne den allgemeinen Ansatz aufzugeben. Somit ist gewährleistet, daß in der Zukunft Neuentwicklungen – etwa auf dem Gebiet der Wissensrepräsentation oder der semantischen Analyse – so gut wie möglich in die bestehende Konzeption integriert werden können.

Wissensmanagement – eine interdisziplinäre Aufgabe

In der Einleitung wurde ein zentrales Problem des Wissensmanagements charakterisiert als die Aufgabe, den *richtigen Personen* die *richtigen Dokumente* zur *richtigen Zeit* zur Verfügung zu stellen.
In dieser Arbeit wurde mittels Methoden der Informationstechnik genauer untersucht, wie man zu den *richtigen Dokumenten* kommen kann. Dazu wurde mit dem Inhaltsassistenten ein geeignetes Werkzeug entwickelt, das auf Erkenntnissen der Künstlichen Intelligenz-Forschung und der Kognitionswissenschaften basiert, um gemeinsames Textverstehen von menschlichen und artifiziellen Akteuren zu ermöglichen.
Gemäß der beschriebenen Wissensmanagement-Problematik ist noch zu fragen, wer die *richtigen Personen* sind, und was die *richtige Zeit* ist. Für beide Fragestellungen könnten möglicherweise Techniken eingesetzt oder weiterentwickelt werden, die schon lange Gegenstand der Forschung innerhalb der Künstlichen Intelligenz und angrenzender Wissenschaftsgebiete sind. Ergebnisse aus den Forschungen zur Benutzermodellierung (z.B. [Kobsa et al. 89]) könnten zu bestimmen helfen, wer die *richtigen Personen* sind. Hier ist insbesondere die Fragestellung interessant, wie pragmatische Aspekte in die Informationssuche und -verteilung einbezogen werden können, also die Ziele, die Informationskonsumenten bei der Suche haben. Für die Modellierung der Zeit in Wissensmanagement-Systemen könnte die Anwendbarkeit von Zeitlogiken (z.B. [Allen 84], [Allen 91]) überprüft werden.
Interessanterweise fehlt bei der Wissensmanagement-Problematik, wie sie oben definiert wurde, als physikalische Dimension der *Raum*. Das liegt wohl daran, daß einerseits die Orte der Personen als *fix* angenommen werden: Der Ort einer Person ist identisch mit seinem Computerarbeitsplatz, von dem aus er Zugriff auf das Informationssystem hat. Andererseits werden die Dokumente im Zeitalter des Internet als nahezu beliebig schnell beweglich angesehen, so daß auch für sie der Ort keine Rolle zu spielen scheint.
Zu Beginn der Vernetzung von Informationssystemen hat man postuliert,

daß durch solche Systeme die Begriffe "Zeit" und "Raum" vieles von ihrer Bedeutung verlieren würden, weil Informationen nahezu jeder Zeit an nahezu jedem Ort verfügbar seien. Wenn diese Aussage auch einen wahren Aspekt hat, so zeigt doch die Erkenntnis, daß man sehr häufig zu einem bestimmten Zeitpunkt *zuviel irrelevante* und *zu wenig wichtige* Information bekommt, daß die "Zeit" durchaus wieder wichtig ist und mit in den Entwurf von Informationssystemen einbezogen werden sollte.
Ähnliches könnte sich in Zukunft für die Orte herausstellen. Bei der Möglichkeit, in der ganzen Welt auf Dokumente zuzugreifen, könnten Orte von Dokumenten oder Orte von Personen, die Dokumente erzeugen oder suchen, wieder ein wichiges Kriterium dafür sein, für wen welches Dokument relevant sein könnte. Jedenfalls sollte diese räumliche Dimension – zumindest in der Forschung – beim Entwurf von Informations- und Wissensmanagementsystemen nicht leichtfertig aus den Augen verloren werden.[2]

Kommunikationssituation

Der Grund, warum es so wichtig ist, daß der Inhaltsassistent Repräsentationen aufbaut, die mit denen der verschiedenen Benutzer kompatibel sind, liegt wesentlich darin, daß solche Expertensysteme der dritten Generation weniger als *selbständige Problemlöser* betrachtet werden, sondern mehr als *Partner im Problemlöseprozeß*, der oft in *Zusammenarbeit mit vielen anderen Benutzern* geschieht. Daher ist es wichtig, die gesamte Kommunikationssituation in der Informationslandschaft zu betrachten.

Das Benutzen von Systemen wie dem vorgeschlagenen läßt sich als *mediated communication* auffassen. Die Kommunikationsforschung hat festgestellt, daß dabei das *mutual knowledge* Problem auftritt. Dieses basiert darauf, daß Sprecher sich in der Kommunikation bewußt sein müssen, was der Adressat weiß oder nicht weiß[Krauss et al. 90]. Es wird daher angenommen, daß ein *common ground* [Clark et al. 82] notwendig ist.

Um *mutual knowledge* zu etablieren, werden drei Mechanismen angenommen: a) direktes Wissen (Peter sagt Paul, daß der den Film "Blade Runner" gesehen hat.) b) Zugehörigkeit zu einer (sozialen) Kategorie (Als Student einer deutschen Hochschule weiß man, daß der AStA ein Studentengremium ist.) c) Dynamik der Interaktion (Wenn etwas gesagt wurde, gehört es danach zum *mutual knowledge*; Rückmeldungen durch den Adressaten werden

[2] Aktuelle Arbeiten zeigen, daß durch Einbeziehung von Ortsinformationen, wie sie etwa in URLs kodiert ist, durchaus bessere Ergebnisse bei der Informationssuche erzielt werden können. (vgl. [Shakes et al. 97])

wahrgenommen.) Oft sind Kommunikationssituationen asymmetrisch. Das gilt im eingangs beschriebenen Beispiel der Studierenden und Universitäten genauso wie in betrieblichen Anwendungen, die einen *corporate memory* etablieren oder *common work* unterstützen. Um mit dieser Asymmetrie (Experten vs. Novizen) besser umgehen zu können, kann der vorgestellte Ansatz der Repräsentation auf mehreren Ebenen helfen. Die Akteure können ihren Schwerpunkt auf der ihnen jeweils angemessen Ebene setzen und sich evtl. "hocharbeiten", zu Experten werden. Um das gemeinsame Verständnis in der gesamten Lebenszeit eines solchen Systems besser aufrecht erhalten zu können, sind Abgleichprozesse eingeplant (*cooperative knowledge evolution*), in denen dieses Verständnis zwischen den Akteuren explizit neu ausgehandelt wird.

Zu Anfang wurde festgestellt, daß es sich beim intelligenten *information retrieval* um eine strategische Herausforderung handelt, die besonders auch innerhalb der Wissensmanagement-Problematik zentral ist. Diese Arbeit hat gezeigt, daß für das *information retrieval* der Übergang von einer an der Dokumentoberfläche orientierten Repräsentation auf tiefere, semantische Wissensdarstellungen mit Blick auf den Einsatz für das Wissensmanagement erfolgversprechend ist, da er hinsichtlich der Dokumentinhalte zu einem gemeinschaftlichen Verständnis der verschiedenen Akteure innerhalb der Informationslandschaft beitragen kann.

□

"quod scripsi scripsi"
(Joh, 19:22)

Literaturverzeichnis

[Allen 84] J. F. Allen. Toward a general theory of action and time. *Artificial Intelligence*, 23(2) Seite: 123–154, 1984.

[Allen 91] J. Allen. Time and Time Again: the Many Ways to Represent Time. *International Journal of Intelligent Systems*, 6(4) Seite: 341–355, November 1991.

[Benjamins et al. 98] V. R. Benjamins, D. Fensel, A. Gomez-Perez, S. Decker, M. Erdmann, E. Motta und M. Musen. Knowledge Annotation Initiative of the Knowledge Acquisition Community KA^2. In: *Proceedings of 11th Banff Knowledge Acquisition for Knowledge-Based System Workshop (KAW98)*, Banff, Canada, 1998.

[Berry et al. 93] M. Berry, T. Do, G. O'Brien, V. Krishna und S. Varadhan. SVDPACKC (Version 1.0) User's Guide. Technischer Bericht UT-CS-93-194, Department of Computer Science, University of Tennessee, April 1993. Thu, 17 Sep 98 15:27:50 GMT.

[Berry et al. 95] M. W. Berry, S. T. Dumais und Gavin W. O'Brien. Using Linear Algebra for Intelligent Information Retrieval. *SIAM Review*, 37 Seite: 573–595, 1995.

[Brachman et al. 85] R. J. Brachman und J. G. Schmolze. An overview of the KL-ONE knowledge representation system. *Cognitive Science*, 9(2) Seite: 171–216, 1985.

[Carbonell 98] J. Carbonell. Powertools for Cyberspace Navigation: Translingual Retrieval, Summarization, Event

Tracking and Novelty Detection. *Slides of Talk presented at the DFKI, Saarbrücken, April 1, 1998*, Seite: 2–4, 1998.

[Chomsky 57] N. Chomsky. *Syntactic structures.* Janua linguarum. Mouton, The Hague, 1957.

[Clark et al. 82] H. H. Clark und T. B. Carlson. Speech Acts and Hearer's Beliefs. In: Neil V. Smith (Herausgeber), *Mutual Knowledge*, Seite: 1–37. Academic Press, New York, New York, 1982.

[Craven et al. 98] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam und S. Slattery. Learning to Extract Symbolic Knowledge from the World Wide Web. In: *Proceedings of 15th National Conference on Artificial Intelligence (AAAI-98)*, 1998.

[Cullum et al. 85] J. K. Cullum und R. A. Willoughby. *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*, Band 1 Theory. 2 Programs. Birkhäuser, Stuttgart, 1985.

[Deerwester et al. 90] S. Deerwester, S. Dumais, G. Furnas, T. Landauer und R. Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6) Seite: 391–407, 1990.

[Dumais 91] S. T. Dumais. Improving the retrieval of information from external sources. *Behavior Research Methods, Instruments and Computers*, 23(2) Seite: 229–236, 1991.

[Dumais 95] S. T. Dumais. Using LSI for information filtering: TREC-3 experiments. In: *D. Harman (Ed.), The Third Text REtrieval Conference (TREC3).* 1995.

[Fensel et al. 97] D. Fensel, M. Erdmann und R. Studer. Ontology Groups: Semantically Enriched Subnets of the WWW. In: *Proceedings of the International Workshop Intelligent Information Integration during the 21st German Annual Conference on Artificial Intelligence*, Freiburg, Germany, 1997.

[Fensel et al. 98] D. Fensel, S. Decker, M. Erdmann und R. Studer. Ontobroker: Or How to Enable Intelligent Access to the WWW. In: *Proceedings of 11th Banff Knowledge Acquisition for Knowledge-Based System Workshop (KAW98)*, Banff, Canada, 1998.

[Fischer et al. 94] G. Fischer, R. McCall, J. Ostwald, B. Reeves und F. Shipman. Seeding, Evolutionary Growth and Reseeding: Supporting the Incremental Development of Design Environments. In: *Proceedings of ACM CHI'94 Conference on Human Factors in Computing Systems*, Band 2 of *PAPER ABSTRACTS: Design Evaluation*, Seite: 220, 1994.

[Fischer et al. 96] H.-R. Fischer, S. Löbner, G. Strube und U. Furbach. Proposition. In: Gerhard Strube (Herausgeber), *Wörterbuch der Kognitionswissenschaft*, Seite: 543. Klett-Cotta, Stuttgart, 1996.

[Foltz et al. 92] P. W. Foltz und S. T. Dumais. Personalized information delivery: an analysis of information methods. *Communications of the ACM*, 35(12) Seite: 51–60, Dezember 1992.

[Frakes et al. 92] W. B. Frakes und R. Baeza-Yates (Herausgeber). *Information Retrieval: Data Structures and Algorithms.* Prentice-Hall, Englewood Cliffs, NJ 07632, USA, 1992.

[Furnas et al. 84] G. W. Furnas, T. K. Landauer, L. M. Gomez und S. T. Dumais. Statistical semantics: Analysis of the Potential Performance of Keyword Information Systems. In: *Human Factors in Informtion Systems, Thomas and Schneider(eds), Ablex Pub. Norwood NJ, ; ACM CR 8502-0134.* 1984.

[Grice 75] J. P. Grice. Logic and conversation. In: P. Cole und J. L. Morgan (Herausgeber), *Syntax and Semantics*, Band 3: Speech Acts, Seite: 41–58. Academic Press, New York, NY, 1975.

[Guha 97] R. V. Guha. Meta Content Framework: A Whitepaper. *http://www.xspace.net/hotsauce/mcf.html*, 1997.

[Harman 95] D. K. Harman (Herausgeber). *Third Text REtrieval Conference (TREC-3)*, Gaithersburg, MD, USA, 1995. National Institute for Standards and Technology.

[Huhns et al. 97] M. N. Huhns und M. P. Singh (Herausgeber). *Readings in Agents.* Morgan Kaufmann Publishers, 1997.

[Kifer et al. 95] M. Kifer, G. Lausen und W. James. Logical Foundations of Object-Oriented and Frame-Based Languages. *Journal of ACM*, Mai 1995.

[Kintsch 98] W. Kintsch. *Comprehension – A Paradigm for Cognition.* Cambridge University Press, Cambridge, UK, 1998.

[Kobsa et al. 89] A. Kobsa und W. Wahlster (Herausgeber). *User models in dialog systems.* Springer-Verlag, London, 1989. A superset of Computational linguistics 1988 14(3).

[Krauss et al. 90] R. M. Krauss und S. R. Fussell. Mutual knowledge and communicative effectiveness. In: R. E. Kraut J. Galegher und C. Egido (Herausgeber), *Intellectual Teamwork: Social Foundations of Cooperative Work*, Seite: 111–146. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1990.

[Kühn et al. 97] O. Kühn und A. Abecker. Corporate Memories for Knowledge Management in Industrial Practice: Prospects and Challenges. *J.UCS: Journal of Universal Computer Science*, 3(8) Seite: 929–954, August 1997.

[Kühn et al. 98] O. Kühn, A. Lauer, M. Malburg, T. Malik und S. Tabor. FAKT: Systemarchitektur 1.10. Interner Bericht, Deutsches Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, 1998.

[Lamping et al. 96] J. Lamping und R. Rao. Visualizing Large Trees Using the Hyperbolic Browser. In: *Proceedings of ACM CHI 96 Conference on Human Factors in Computing Systems*, Band 2 of *VIDEOS: Visualization*, Seite: 388–389, 1996.

[Lenat 89] D. B. Lenat. When Will Machines Learn? *Machine Learning*, 4 Seite: 255–257, 1989.

[Lenat et al. 90] D. B. Lenat und R. V. Guha. *Building Large Knowledge-Based Systems: Representation and Inference in the CYC Project.* Addison-Wesley, Reading, Massachusetts, 1990.

[Luke et al. 97] S. Luke, L. Spector, D. Rager und J. Hendler. Ontology-based Web Agents. In: W. Lewis Johnson und Barbara Hayes-Roth (Herausgeber), *Proceedings of the 1st International Conference on Autonomous Agents*, Seite: 59–66, New York, Februar5–8 1997. ACM Press.

[Maes 94] P. Maes. Agents that reduce work and information overload. *Communications of the ACM*, 37(7) Seite: 30–40, Juli 1994.

[Malone et al. 89] T. W. Malone, K. R. Grant, K.-Y. Lai, R. Rao und D. A. Rosenblitt. The Information Lens: An Intelligent System for Information Sharing and Coordination. In: M. H. Olson (Herausgeber), *Technological Support for Work Group Collaboration*, Seite: 65–88. Lawrence Erlbaum Associates, 1989.

[Mandler et al. 77] J. M. Mandler und N. S. Johnson. Remembrance of things parsed: Story Structure and Recall. *Cognitive Psychology*, 9 Seite: 111–51, 1977.

[McCallum 97] A. K. McCallum. Bow: A Toolkit for Statistical Language Modeling, Text Retrieval, Classification and Clustering. *http://www.cs.cmu.edu/ mccallum/bow*, 1997.

[Mross et al. 92] E. F. Mross und J. O. Roberts. The construction-integration model: A program and manual. Technischer Bericht ICS No. 921–14, Boulder, CO: Institute of Cognitive Science, University of Colorado, 1992.

[Neches et al. 91] R. Neches, R. Fikes, T. Finin, T. Gruber, R., T. Senator und W. R. Swarton. Enabling Technology for Knowledge Sharing. *AI Magazine*, Seite: 37–51, März 1991.

[Newell 82] A. Newell. The knowledge level. *Artificial Intelligence*, 18 Seite: 87–127, 1982.

[Nigam et al. 98] K. Nigam, M. McCallum, S. Thrun und T. Mitchell. Learning to Classify Text from Labeled and Unlabeled Documents. *Machine Learning*, to appear, 1998.

[Nonaka et al. 95] I. Nonaka und H. Takeuchi. *The Knowledge-Creating Company.* University Press, Oxford, 1995.

[Norvig 89] P. Norvig. Marker Passing as a Weak Method fo Text Inferencing. *Cognitive Science*, 13 Seite: 569–620, 1989.

[Quinlan et al. 93] J. R. Quinlan und R. M. Cameron-Jones. FOIL: A Midterm Report. In: Pavel B. Brazdil (Herausgeber), *Proceedings of the European Conference on Machine Learning (ECML-93)*, Band 667 of *LNAI*, Seite: 3–20, Vienna, Austria, April 1993. Springer Verlag.

[Rodenhausen 92] H. Rodenhausen. Mathematical Aspects of Kintsch's Model of Discourse Comprehension. *Psychological Review*, 99(3) Seite: 547–549, Juli 1992.

[Salton 68] G. Salton. *Automatic Information Organization and Retrieval.* McGraw-Hill, New York, 1968.

[Salton 71] G. Salton (Herausgeber). *The Smart Retrieval System Experiments in Automatic Document Processing.* Prentice Hall, 1971.

[Salton et al. 83] G. Salton und M. J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, Tokio, 1983.

[Salton et al. 88] G. Salton und C. Buckley. Term-weighted approaches to automatic text retrieval. *Inf. Proc. & Mngmt.*, 24(5) Seite: 513–523, ? 1988.

[Schmalhofer 98] F. Schmalhofer. *Constructive Knowledge Acquisition – A Computational Model and Experimental Evaluation.* Lawrence Erlbaum Associates, 1998.

[Schmalhofer et al. 91] F. Schmalhofer, O. Kühn und G. Schmidt. Integrated Knowledge Acquisition from Text, Previously Solved Cases, and Expert Memories. *Applied Artificial Intelligence*, 5 Seite: 311–337, 1991.

[Schmalhofer et al. 94a] F. Schmalhofer, J. S. Aitken und L. E. Bourne, Jr. Beyond the Knowledge Level: Descriptions of Rational Behavior for Sharing and Reuse. In: Luc Steels, Guus Schreiber und Walter van der Velde (Herausgeber), *Proceedings of the 8th European Knowledge Acquisition Workshop : A Future for Knowledge Acquisition*, Band 867 of *LNAI*, Seite: 83–103, Berlin, September 1994. Springer.

[Schmalhofer et al. 94b] F. Schmalhofer und L. van Elst. Entwicklung von Expertensystemen: Prototypen, Tiefenmodellierung und kooperative Wissensevolution. Document D-94-04, Deutsches Forschungszentrum für Künstliche Intelligenz, Deutsches Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, Germany, 1994.

[Schmalhofer et al. 95a] F. Schmalhofer, T. Reinartz und B. Tschaitschian. Unified Approach to Learning For Complex Real-World Domains. *Applied Artificial Intelligence*, 9 Seite: 127–156, 1995.

[Schmalhofer et al. 95b] F. Schmalhofer und B. Tschaitschian. Cooperative knowledge evolution for complex domains. In: G. Tecuci und Y. Kodratoff (Herausgeber), *Machine Learning and Knowledge Acquisition: Integrated Approaches*, Seite: 145–166. Academic Press, 1995.

[Schmalhofer et al. 98a] F. Schmalhofer, M. Bauer, L. van Elst und S. van Mulken. Gesamtkonzept des Distributions- und Inhaltsassistenten. Interner Bericht, Deutsches Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, 1998.

[Schmalhofer et al. 98b] F. Schmalhofer, M. Bauer, L. van Elst und S. van Mulken. Inhaltsbasierter Informationsassistent: Literaturüberblick. Interner Bericht, Deutsches Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, 1998.

[Schmid et al. 96] U. Schmid und M. Kindsmüller. *Kognitive Modellierung. Eine Einführung in die logischen und algorithmischen Grundlagen.* Spektrum, Heidelberg, 1996.

[Schmidt 92] G. Schmidt. Knowledge Acquisition from Text in a Complex Domain. In: F. J. Belli, F.; Radermacher (Herausgeber), *Proceedings of the 5th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE-92)*, Band 604 of *LNAI*, Seite: 529–538, Berlin, Juni9–12 1992. Springer.

[Schreiber et al. 93] G. Schreiber, B. J. Wielinga und J. A. Breuker. *KADS: a Principled Approach to Knowledge-Based System Development.* Academic Press, New York, USA, 1993.

[Selberg et al. 97] E. Selberg und O. Etzioni. The MetaCrawler Architecture for Resource Aggregation on the Web. *IEEE Expert*, (January–February) Seite: 11–14, 1997.

[Shakes et al. 97] J. Shakes, M. Langheinrich und O. Etzioni. Dynamic Reference Sifting: A Case Study in the Homepage Domain. In: *Proceedings of the Sixth International World Wide Web Conference*, 1997.

[Shortliffe 76] E. H. Shortliffe. *Computer-Based Medical Consultations: MYCIN.* Elsevier/North-Holland, Amsterdam, London, New York, 1976.

[Stefik 86] M. Stefik. The Next Knowledge Medium. *AI Magazine*, 7(1) Seite: 34–46, 1986.

[Tschaitschian et al. 97] B. Tschaitschian, A. Abecker und F. Schmalhofer. Information Tuning with KARAT: Capitalizing on Existing Documents. In: Enric Plaza und Richard Benjamins (Herausgeber), *Proceedings of the 10th European Workshop on Knowledge Acquisition, Modeling and Management (EKAW-97)*, Band 1319 of *LNAI*, Seite: 269–284, Berlin, 1997. Springer.

[van Elst et al. 97] L. van Elst und F. Schmalhofer. Die Persistenz von Inferenzen in einem verstehensbasierten kognitiven

Modell. *Kognitionswissenschaft*, 6(2) Seite: 86–98, 1997.

[van Elst et al. 98] L. van Elst und F. Schmalhofer. Portrait des Distributions- und Inhaltsassistenten. Interner Bericht, Deutsches Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, 1998.

[van Rijsbergen 79] C. J. van Rijsbergen. *Information Retrieval.* Butterworths, 1979.

[Verhoeff et al. 61] J. Verhoeff, W. Goffmann und Jack Belzer. Inefficiency of the Use of Boolean Functions for Information Retrieval Systems. *Communications of the ACM*, 4(12) Seite: 557–558, 594, Dezember 1961.

[Voorhees et al. 97] E. M. Voorhees und D. K. Harman (Herausgeber). *Fifth Text REtrieval Conference (TREC-5)*, Gaithersburg, MD, USA, 1997. National Institute for Standards and Technology.

[W3-Consortium 98] W3-Consortium. Extensible Markup Language (XML) 1.0. *http://www.w3.org/TR/1998/REC-xml-19980210*, 1998.

[Wong et al. 87] S. K. M. Wong, W. Ziarko, V. V. Raghavan und P. C. N. Wong. On Modeling of Information Retrieval Concepts in Vector Space. *ACM Transactions on Database Systems*, 12(2) Seite: 299–321, Juni 1987.