Zur bedeutungsorientierten Auflösung von Wortmehrdeutigkeiten

Vorschlag einer Methodik

Dissertation
zur Erlangung des
Doktorgrades der Naturwissenschaften

am Department Informatik
der Universität Hamburg

vorgelegt von

Ronald Winnemöller

Hamburg, den 13. Mai 2009

Genehmigt vom Fachbereich Informatik der Universität Hamburg auf Antrag von

1. Gutachter: *Prof. Dr. Christopher Habel*
Department Informatik
Universität Hamburg

2. Gutachter: *Prof. Dr.-Ing. Wolfgang Menzel*
Department Informatik
Universität Hamburg

Hamburg, den 13. Mai 2009 *Dekan Prof. Dr. Siegfried Stiehl*

Zusammenfassung

In dieser Arbeit wird die Frage behandelt, wie die Leistung bestimmter textverarbeitender Systeme verbessert werden kann. Konkretisiert wird das Thema über die Referenzproblematik der Auflösung von Wortmehrdeutigkeiten (engl. *Word Sense Disambiguation*, abgek. *WSD*), da diese Basisproblem der semantischen Textverarbeitung und gleichzeitig Gegenstand aktueller Forschungsbestrebungen ist.

Ein wesentliches Problem beim Verstehen der Sachverhalte, d.h. bei der Feststellung der Textbedeutung, besteht in der Mehrdeutigkeit vieler sprachlicher Ausdrücke. Einige dieser Unklarheiten können mit einfachen Regeln oder Annahmen aufgelöst werden, in vielen Fällen jedoch können Mehrdeutigkeiten nur durch Anwendung von Hintergrundwissen über die Welt oder von situativem Wissen aufgelöst werden. Damit stellt sich die Frage nach der Natur derartigen (semantisch-pragmatischen) Wissens: der Philosoph Ludwig Wittgenstein charakterisiert die „Familienähnlichkeit“ zwischen Begriffen, die Zuordnung zu „semantischen Kategorien“ und die Bestimmung der Wortbedeutung über ihren Gebrauch als sinnvolle Alternative zu herkömmlichen Semantiktheorien. Diese Basis führt über die sog. Prototypensemantik, in der die prototypische Auffassung der Wortbedeutung als „Grad der Repräsentativität eines Objektes“ vorherrscht — und weiter zu der in dieser Arbeit entwickelten TSR-Semantik, in der verschiedene Arten von „Zentralität“ und ein Anwendungsbereich, der durch eine Gebrauchskategorisierung bestimmt wird. Die TSR-Semantik dient hierbei nicht lediglich der theoretischen Auseinandersetzung, sondern vielmehr Zwecken der konkreten Problemlösung, d.h. zur Anwendung praktikabler, repräsentierender Objekte führt.

Schwerpunkt der Arbeit ist der vom Verfasser entwickelte Vorschlag einer Repräsentation gebrauchsorientierter Aspekte der Textbedeutung, der TSR-Ansatz. Kernobjekte des TSR-Ansatzes sind die TSR-Bäume oder einfach TSRs, die die o.a. Semantikvorstellungen realisieren und auf der Datenbasis eines Internetverzeichnisses basieren. Zudem werden verschiedene Funktionen und Operationen auf diesen Strukturen definiert, die für die eigentliche Nutzbarkeit des Ansatzes wesentlich sind.

Weil der TSR-Ansatz seine Eignung als praktikables Lösungsverfahren für die WSD-Problematik belegen soll, wurde in diesem Sinne ein prototypisches System realisiert.

Über eine Reihe von Experimenten wurde festgestellt, ob durch die TSR-Methodik im Bereich der Auflösung von Wortmehrdeutigkeiten signifikant bessere Ergebnisse erbracht werden können als über vergleichbare herkömmliche Methoden. Hierbei wurde die Verwendung von Wortvektoren als Repräsentant konventioneller Methoden gewählt, da sie eine Grundlage der meisten im SENSEVAL English Lexical Sample Wettbewerb (einem Quasi–Standardevaluationsverfahren) bewerteten Systeme darstellt. Hierfür wurden fünf Testszenarien mit unterschiedlichem Grad der Vorverarbeitung konstruiert und geprüft.

Insgesamt kann gezeigt werden, daß durch die Erarbeitung einer »aspektbasierten« Semantik semantische und pragmatische Aspekte durch Bezug zum Weltwissen genutzt und erfolgreich im WSD-Anwendungskontext eingesetzt werden können. Hierbei wird zudem eine Beobachtung verschiedener semantischer Sprachvorgänge ermöglicht, da das TSR-Verfahren im Gegensatz zu z.B. statistischen Methoden keine „black box“ ist. Über die Anwendbarkeit des Kompositionalitätsprinzipes auf beliebiger textueller Granularitätsebene durch Wort-TSRs, Chunk-TSRs, Satz-TSRs etc. kann weiter von der reinen „Wortbedeutung“ auf eine „Textbedeutung“ (im Sinne der TSR-Semantik) abstrahiert werden. Schließlich bestehen Integrationmöglichkeiten in bestehende NLP-Anwendungen über eine Transformation der TSRs in Indexvektoren.

Abstract

In this work, we will elaborate on how to improve natural language processing (NLP) systems effectiveness by improving word sense disambiguation methodology (WSD). WSD is a particularly interesting topic in this sense, because it is both a fundamental problem of NLP and object of current research efforts.

Many ambiguous expressions can be resolved by applying simpe heuristics or formulas – but yet many others need application of world, domain or situated knowledge. It is therefore vital to discuss the nature of such (semantic and pragmatic) knowledge: the philosopher Ludwig Wittgenstein used the notions of „family resemblance“, „semantic categories“ and the definition of word meaning through word use in order to characterize an alternative understanding of text meaning. His work, as well as the so-called „prototype semantics“ theory, which in turn is based on the prototypical properties of notions and their respective „centrality“, forms the grounds of the semantic theory, proposed in this work. This „TSR-semantics“ is subsequently used for practical application on WSD by creating a realistic system architecture based on it.

This architecture is formed around so–called TSR trees, graphical representations of „meaning aspects“ in the sense of the above mentioned understanding of semantics. The TSR trees are derived from actual internet directories by transforming their textual and structural information appropriately. Apart from the trees themselves, a number of operations based on them were defined and implemented in a concrete prototype system, thus enabling the TSR–proposal to be evaluated.

A number of experiments, most of them based on the „SENSEVAL English Lexical Sample“ Task (a quasi standard evaluation procedure for WSD) were carried out in order to verify the first claim that TSRs are more effective on WSD than conventional methods – in this case, word vectors were used as representatives of such methods. The Evaluation included testing five different scenarios and showed that the application of the TSR-methodology, and therefore the application of aspect–based semantics and pragmatics, indeed can be justified in the WSD context.

Furthermore, because the TSR-approach is not a „black box“ principle – as many statistical methodologies are – certain linguistic operations can be observed and monitored. Because of the differences to conventional semantic theories, the

principle of compositionality can be applied on TSRs, moving from mere word-meaning to general text meaning analysis.

Lastly, because of the opportunity to transform TSR trees into index vectors, TSR based systems can be integrated into exisiting third party architectures with very little effort.

Danksagung

Ich danke meiner Familie Claudia, Emily und Lea für ihren Glauben in meine Arbeit und für ihre liebevolle Unterstützung. Besonders danke ich meinem Bruder Holger für den häufigen Austausch über unsere jeweiligen Dissertationsvorhaben; ebenso danke ich meinen Eltern und anderen Geschwistern, meiner Schwiegermutter und der Familie meiner Frau für ihren Optimismus gegenüber meinem Vorhaben.

Sehr wertvoll war auch die Unterstützung meiner Betreuer Herr Habel und Herr Menzel - besonders die konstruktiven Dispute und ihre hilfreichen Vorschläge bezüglich der Dissertation, aber auch meiner verschiedenen Veröffentlichungen. In diesem Zusammenhang möchte ich auch Herrn von Hahn danken, der sowohl meine Diplomarbeit als auch die Anfangszeit meiner Dissertation begleitete und unterstützte.

Nicht zuletzt möchte ich meinen Freunden und Kollegen für ihre Hilfe und begleitenden Gedanken meinen Dank aussprechen, vor allem Birko Lischke für die aufwändige Durchsicht meiner Dissertation auf offensichtliche Schwächen und Fehler, sowie Wiebke Oeltjen, Iver Jackewitz und Carsten Benecke für ähnliche Hilfestellungen bezüglich meiner Veröffentlichungen.

Inhaltsverzeichnis

Kapitel 1

Einleitung

In dieser Arbeit wird die Frage behandelt, wie die Leistung bestimmter textverarbeitender Systeme verbessert werden kann. Betrachtet werden vor allem solche Verfahren, die sich auf die Verarbeitung natürlichsprachlicher Texte beziehen.

Beispiele hierfür sind Systeme, die automatisch Texte von einer Sprache in eine andere übersetzen, die Kurzzusammenfassungen von Textdokumenten erstellen oder die gezielt nach bestimmten Informationen suchen, beispielsweise nach Firmennamen von internationalen Konzernbildungen. Außerdem soll nicht die globale Vielfalt der verschiedenen hierbei verwendeten Algorithmen, Implementierungen und Theorien untersucht werden, sondern ein spezifisches – wenn auch zentrales – Teilgebiet des Bereiches.

Viele Systeme aus dem Bereich der Verarbeitung natürlichsprachlicher Texte sind in Form von Prozesspipelines realisiert, also als modulare Systeme, deren Komponenten weitgehend funktional voneinander abgrenzbar sind und idealerweise Daten empfangen, verarbeiten und an einen nachfolgenden Modul übergeben, ohne notwendigerweise die Funktionsweise der anderen beteiligten Komponenten zu kennen.

Eine solche Prozesspipeline für die Verarbeitung natürlicher Sprache (engl. *natural language processing*, abgekürzt *NLP*), in diesem Fall für eine einfache Suchmaschine, ist exemplarisch - und stark vereinfacht - in Abbildung 1.1 auf der nächsten Seite dargestellt.

Weitere Beispiele finden sich in der einführenden Untersuchung von Prozessen zur Informationsgewinnung aus Texten von Ralph Grishman (vgl. [Grishman, 1997]), in der Dissertation von Hamish Cunningham, in welcher dieser das universelle Framework GATE zur Verarbeitung von natürlichsprachlichen Texten beschreibt (vgl. [Cunningham, 2000]), sowie in neueren Arbeiten zur Komposition von NLP-Prozesspipelines unter Nutzung moderner Webtechnologien (z.B. vgl. [Halpin, 2006] und [Chute et al., 2006]).

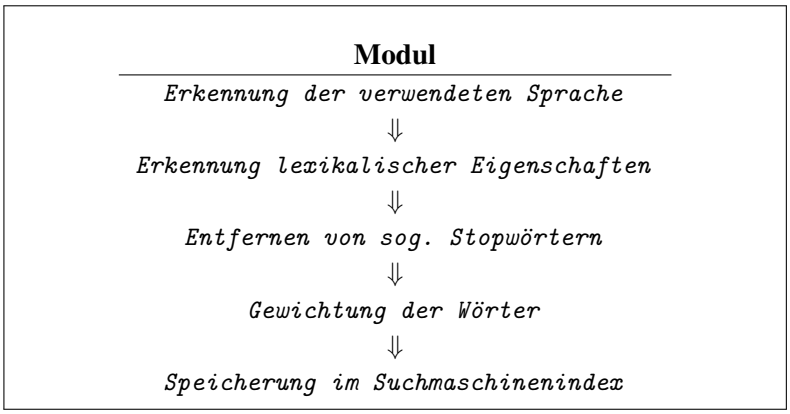

Abbildung 1.1: NLP-Pipeline „Suchmaschine“

Für eine solche Art des modularen Vorgehens wird von Cunningham (auszugsweise) wie folgt argumentiert[1]:

- Die Komponenten können erheblich leichter in anderen Anwendungen wiederverwendet werden, da der Zusatzaufwand der Integration, Dokumentation und der Datenvisualisierung entfällt.
- Modulare Systeme ermöglichen die kollaborative Zusammenarbeit über institutionale Grenzen hinweg.
- Die komparative Evaluation verschiedener Komponenten von vergleichbarer Funktionalität wird durch die leichte Auswechselbarkeit von Modulen ermöglicht.

Allerdings zeigen existierende NLP-Systeme, die auf Prozesspipelines beruhen, auch einige Nachteile, wie von Graca et al. wie folgt zusammengefasst wird (vgl. [Graca et al., 2006]):

- Es können Informationsverluste entstehen, weil einzelne Module ggf. nicht alle Informationen ihrer jeweiligen Eingabedaten nutzen - und damit auch nicht an subsequente Module weitergeben.
- Jeder Modul hat in der Regel sein eigenes Ein- und Ausgabeformat. Dies trifft in besonderem Maße zu, wenn die Module von unterschiedlichen Herstellern und aus unterschiedlichen Anwendungskontexten stammen. Eine einfache Datentransformation ist häufig nicht möglich, da sich die Repräsentationskomplexität der Formate unterscheiden kann (also einige Formate mehr Informationen aufnehmen können als andere).

[1] Diese Darstellung erfolgt in Bezug auf das GATE-Framework, ist als Anforderungsprofil jedoch verallgemeinerbar.

- Die Informationsflüsse zwischen verschiedenen Modulen gestatten oft keine Verknüpfungen zwischen Ein- und Ausgabedaten. Diese Verknüpfungen werden aber benötigt, um die in diesen Daten enthaltenen Informationen zueinander in Beziehung setzen zu können.

Aus den oben genannten Punkten ergibt sich in der Regel eine Zunahme der Gesamtfehlerrate durch Akkumulation der Fehler einzelner Module (vgl. [Halpin, 2006]), wie in Abbildung 1.2 exemplarisch dargestellt ist – wobei eine (hypothetische) Fehlerrate von 10% in vielen Modulen noch als relativ optimistisch angesehen werden kann. Schon in dem simplifizierten Modell aus der Abbildung ergibt sich damit eine kumulierte Fehlerrate von 40% – für reale Anwendungen häufig zu hoch, so daß meistens über Einschränkungen der Anforderungen an das Endergebnis oder durch Verbesserungen der Modulresultate versucht wird, dieses Problem zu beseitigen.

Modul	**Fehlerrate**
Erkennung der verwendeten Sprache	10%
⇓	
Erkennung lexikalischer Eigenschaften	20%
⇓	
Entfernen von sog. Stopwörtern	30%
⇓	
Gewichtung der Wörter	40%
⇓	
Speicherung im Suchmaschinenindex	

Abbildung 1.2: NLP-Pipeline mit zunehmender Fehlerrate

Graca et al. schlagen die folgende Liste von Anforderungen bezüglich der Repräsentation von Informationen vor, die ein Konzept für eine Basistechnologie zur Erstellung von NLP-Prozesspipelines erfüllen sollte:

1. Informationen, die ein NLP-Modul produziert, sollten auf einer einzigen Ebene gehalten werden. „Ebenen“ sind in diesem Zusammenhang beispielsweise die Textebene, die Ebene der Textstruktur (die durch eine Syntaxanalyse erzeugt wird), die Ebene der Semantik, etc.

2. Mehrdeutigkeiten (auf verschiedenen Ebenen) sollen darstellbar sein.

3. Die Darstellungsmethodik sollte Baumstrukturen sprachlicher Elemente unterstützen können.

4. Beziehungen zwischen sprachlichen Elementen sollen darstellbar sein.

5. Beziehungen zwischen Informationen auf unterschiedlichen Ebenen sollten repräsentierbar sein.

Um die Leistung von NLP-Anwendungen zu steigern, sind daher prinzipiell zwei Ansätze denkbar:

1. Leistungssteigerung einzelner Module. Dies gilt besonders für Module von zentraler Bedeutung, aber vergleichsweise geringer Leistung, da diese die Leistung des Gesamtsystems oft maßgeblich beeinflussen.

2. Verbesserung des Informationaustausches zwischen den einzelnen Modulen.

Intuitiv scheint man versucht zu sein, einfach mehr Daten in die Strukturen zu pressen, die von den Programmen verstanden werden müssen, in der Hoffnung, daß irgendwann eine hypothetisches kritisches Volumen an verfügbarem Wissen überschritten wäre und daraufhin textverarbeitende Algorithmen einfach funktionieren müßten. Leider scheinen Steigerungen der Repräsentationskomplexität von Informationen innerhalb von textverarbeitenden Systemen jedoch häufig nur marginale Verbesserungen der Leistung zu erbringen - oder gar keine, wie bereits 1992 von Lewis (Vgl. *Equal Effectiveness Paradox*, [Lewis, 1992, S. 7]) formuliert: vereinfacht gesagt, stellt Lewis hiermit die Hypothese auf, daß die Modifikation komplexer Verfahren im Bereich des Information Retrieval ab einem bestimmten Punkt keine Verbesserung der Leistung mehr bewirkt. Es scheint, daß die Aussagefähigkeit der verfügbaren Information von bestehenden Verfahren bereits annähernd maximal ausgeschöpft wird. Aus diesem Grunde entsteht die Motivation, den *Informationsgehalt* der Daten zu steigern, die diesen Systemen zugrunde liegen, um die Leistung dieser Systeme hierdurch ebenfalls steigern zu können. Konkret ausgedrückt könnte dies bedeuten, nicht nur größere Korpora und damit einfach mehr Wörter und statistische Zusammenhänge zwischen diesen zu analysieren, sondern Wertattribute an Wörter zu binden und über deren Zusammenhänge mehr Informationen zu gewinnen.

Hierzu bedarf es nicht nur einer geeigneten Basistechnologie (im Sinne eines allen Modulen zugrundegelegten Frameworks), sondern auch einer angemessenen Repräsentation der verarbeiteten Informationen. In dieser Arbeit wird die Ansicht vertreten, daß eine rein technisch orientierte Repräsentationsweise von Informationen, die insbesondere der Übermittlung von semantischen Daten dienen soll, ungenügend ist. Der Grund hierfür ist, daß häufig gerade die inhaltliche *Bedeutung* übermittelt werden soll. Für einen korrekten Transfer der relevanten Information muß daher auch eine theoretische Einigung darüber bestehen, wie diese Informationen zu interpretieren sind. Es ist folglich erforderlich, festzustellen, wie der Begriff Bedeutung interpretiert bzw. verwendet wird.

Nach James Allen (vgl. [Allen, 1995]) hat die automatische Textverarbeitung zwei Ziele: Erkenntnisse über das Wesen der Sprache gewinnen und außerdem qualitativ

bessere Verfahren zu entwickeln. Nach Ansicht von Allen stehen beide Ziele in engem Zusammenhang, so daß über eine bessere Erkenntnis der Mechanismen der Sprache auch bessere Verfahren entwickelt werden können und umgekehrt bessere Verfahren auch bessere Einsichten in sprachliche Prozesse ermöglichen können.

In dieser Arbeit wird auf die Problematik in folgender Weise eingegangen: Es wird eine theoretisch fundierte Darstellungsmethodik für bestimmte, an Vorstellungen der Textbedeutung orientierte NLP-Informationen entwickelt werden, die prinzipiell in der Lage ist, den oben vorgestellten Anforderungen zu genügen. Da ein solcher Anspruch in seiner Allgemeingültigkeit bzw. für die Gesamtheit der denkbaren NLP-Pipelines kaum zu untermauern ist, wird diese Arbeit sich auf die Funktion eines einzigen (wenn auch zentralen) Moduls beschränken und anhand dessen die Möglichkeiten der entwickelten Lösung - im Vergleich zur meistverwendeten herkömmlichen Lösung - darlegen.

Konkret dient in dieser Arbeit die Auflösung von Wortmehrdeutigkeiten (engl. *Word Sense Disambiguation*, abgek. *WSD*) als Referenzanwendung für den entwickelten Ansatz, da sie Basisproblem der semantischen Textverarbeitung und gleichzeitig Gegenstand aktueller Forschungsbestrebungen ist: Beispielsweise kann das Wort *Band* sowohl einen Gegenstand referenzieren, als auch eine Musikgruppe als auch einen Frequenzbereich, ... soll eine befragte Suchmaschine Informationen um die Katze *Jaguar* liefern - oder um den gleichnamigen Automobilhersteller? Tatsächlich stehen derartige Fragen am Anfang vieler komplexer Aufgabenstellungen der automatischen Textverarbeitung (die ausführliche Behandlung der WSD-Problematik ist Inhalt von Kapitel 2). Anhand dieser Beispiele läßt sich die zugrunde liegende Problematik recht gut beschreiben: es besteht die grundlegende Schwierigkeit, daß automatische Systeme die Bedeutung der ihnen gestellten Fragen und der von ihnen verarbeiteten Inhalte nicht kennen. Man kann sagen, sie *verstehen* diese Inhalte nicht; nicht einmal, wie diese Inhalte sprachlich korrekt *benutzt* werden - ihnen fehlt das „Weltwissen“, die relevanten Hintergrundinformationen. Dieses Verständnis ist jedoch häufig erforderlich, um bestimmte Fragestellungen - wie die oben angeführten - zufriedenstellend beantworten zu können. Hieraus läßt sich zudem das Problem ableiten, wie Inhalte des Weltwissens in einem modularen NLP-System weitergegeben und unter Berücksichtigung der Weitergabe angemessen repräsentiert werden können - hierbei handelt es sich allerdings um ein komponentenübergreifendes Problem.

Durch diese Überlegungen motiviert, lassen sich die Ziele der vorliegenden Arbeit wie folgt formulieren: Der entwickelte Ansatz soll ...

- ... Aspekte der Semantik und der Pragmatik für die Auflösung von Wortmehrdeutigkeiten nutzbar zu machen. Dieser Ansatz soll Hintergrundwissen (insbesondere Weltwissen) nutzbar machen, um semantische Mehrdeutigkeiten auflösen zu können.

- ... die wesentlichen Vorteile bisheriger Verfahrensparadigmen beibehalten. Das

Phänomen des o.a. „Equal Effectiveness Paradox“ legt eine alternative Darstellungsweise semantischer und pragmatischer Aspekte nahe - es soll also hierfür eine robuste und leistungsfähige Repräsentationsmethodik entworfen und umgesetzt werden.

... auf einer klaren theoretisch-philosophischen Basis fundieren. Diese Basis soll das spätere, praktische Verfahren vorbereiten und die Interpretation der empirischen Untersuchungen argumentativ unterstützen.

... Einsichten in sprachliche Zusammenhänge ermöglichen (im Sinne einer Funktionstransparenz).

... darauf angelegt sein, sowohl das spezielle Problem der Verbesserung von WSD-Leistung als auch das allgemeine Problem der Verbesserung der Leistung von NLP-Modulen allgemein praktisch zu handhaben. Die Leistungsfähigkeit des entwickelten Ansatzes soll experimentell überprüft werden, hierzu soll ein geeigneter Prototyp entwickelt werden. Gegenstand der Evaluation soll ein relevantes und aktuelles Problem der automatischen Textverarbeitung sein - aus diesem Grunde bietet sich die oben angeführte Auflösung von Wortmehrdeutigkeiten als Aufgabenstellung an.

Wegen der bedeutungsorientierten Ausrichtung der entwickelten Methodik soll diese die englische Bezeichnung *Text Sense Representation* Ansatz, kurz *TSR*-Ansatz erhalten. Wenn also den folgenden Kapiteln auf den „TSR-Ansatz“ Bezug genommen wird, ist die in der vorliegenden Arbeit entwickelte und als zentrales Thema behandelte Methodik (wie auch deren zugrundeliegender Theorie) gemeint.

Hierzu ist die Dissertation im weiteren Text wie folgt gegliedert:

Zur Auflösung von Wortmehrdeutigkeiten (**Kapitel 2 auf der nächsten Seite**): Vorstellung wichtiger Paradigmen, Ansätzen und Verfahren im Bereich WSD (Stand der Forschung)

Theorie und Datengrundlagen (**Kapitel 3 auf Seite 63**): Entwicklung einer theoretischen Basis, die durch die in der Einleitung dargelegten Aspekte und Zielvorstellungen motiviert ist und die eine adäquate Grundlage für eine realistische Umsetzung bietet

Realisierungssvorschlag (**Kapitel 4 auf Seite 95**): Vorstellung eines Ansatzes, der der Theorie entspricht, praktisch durchführbar ist und für die Lösung der in der Einleitung skizzierten Problematik geeignet ist

Evaluation des Ansatzes (**Kapitel 5 auf Seite 139**): Aufzeigen der Leistungsfähigkeit des Ansatzes in Bezug auf das WSD-Problem

Schluß (**Kapitel 6 auf Seite 153**): Fazit und Ausblick

Kapitel 2

Zur Auflösung von Wortmehrdeutigkeiten

‘Twas brillig, and the slithy toves
Did gyre and gimble in the wabe:
All mimsy were the borogoves,
And the mome raths outgrabe.

"Beware the Jabberwock, my son!
The jaws that bite, the claws that catch!
Beware the Jubjub bird, and shun
The frumious Bandersnatch!"

He took his vorpal sword in hand:
Long time the manxome foe he sought –
So rested he by the Tumtum tree,
And stood awhile in thought.

And, as in uffish thought he stood,
The Jabberwock, with eyes of flame,
Came whiffling through the tulgey wood,
And burbled as it came!

One, two! One, two! And through and through
The vorpal blade went snicker-snack!
He left it dead, and with its head
He went galumphing back.

"And, has thou slain the Jabberwock?
Come to my arms, my beamish boy!
O frabjous day! Callooh! Callay!’
He chortled in his joy.

‘Twas brillig, and the slithy toves

Did gyre and gimble in the wabe;
All mimsy were the borogoves,
And the mome raths outgrabe.

[Lewis Carroll, *Jabberwocky. Through the Looking-Glass and What Alice Found There*, 1872.]

„Verstehen Sie dieses Gedicht?“

Intuitiv dürfte den meisten Lesern mit einem guten Verständnis der englischen Sprache der Inhalt des Gedichtes ungefähr klar sein.

Was ist damit gemeint? Ist uns der Inhalt eines Textes „klar“, wenn wir wissen, „worum es geht“? ... und was bedeutet: „worum es geht“? Gerade „*Jabberwocky* spielt mit unserer Intuition von Verständnis; wir wissen beispielsweise, was ein Vogel ist – da wir Sperling, Drosseln, Elstern und Krähen kennen, und viele mehr – aber den (erdachten) „Jubjub“-Vogel kennen wir nicht.

Dennoch haben wir eine vage Vorstellung dieses Dinges, wir visualisieren einen Vogel aus der Erinnerung. Damit verstehen wir den ungefähren Sinn der betreffenden Strophe, auch wenn wir die genaue Bedeutung von Jubjub-Vogel nicht kennen (können).

In vielen Fällen scheint uns die undeutliche Kenntnis einer beschriebenen Situation zu genügen, um sie „verstehen“ zu können. Nach John Searle ist jeder Mensch ein Fachmann darin, beurteilen zu können, wann er etwas verstanden hat (vgl. [Searle, 1997]).

Er saß neben dem Baum.

Ebenso haben wir das Gefühl, auch diesen Satz intuitiv zu verstehen - ohne zu wissen, z.B. um wen es geht, um was für eine Art Baum es sich handelt, auf welcher Seite des Baumes gesessen wird usw.

In bestimmten Situationen genügt dieses intuitive Verständnis jedoch nicht, etwa während der Beweisführung einer Gerichtsverhandlung, in welcher der Protagonist ein Zeuge ist und sein Gesichtsfeld möglicherweise vom Baum eingeschränkt wurde.

In solchen Fällen müssen wir uns weiterführende Informationen beschaffen oder Vermutungen anstellen um unser Verständnis zufriedenstellend erweitern zu können. Hierbei nutzen wir häufig unser Wissen um Sachverhalte in der Welt, oder unsere Kenntnis über die Art und Weise, wie bestimmte Sachverhalte üblicherweise beschrieben werden.

Eine wesentliche Eigenschaft des Verstehens ist, daß wir uns ein Bild der beschriebenen Sachverhalte machen können (vgl. auch [Wittgenstein, 1984a, § 4.021]), d.h. wir können uns das Beschriebene vorstellen. Man kann auch sagen: uns ist die „Bedeutung“ des Beschriebenen einigermaßen klar (Dies gilt natürlich auch für sprachliche Inhalte, die keinen beschreibenden Charakter im strengen Sinne haben, z.B. Fragen).

Ein wesentliches Problem beim Verstehen der Sachverhalte, d.h. bei der Feststellung der Textbedeutung, besteht in der Mehrdeutigkeit vieler sprachlicher Ausdrücke (hierunter wollen wir zunächst einmal Wörter, Phrasen, Sätze und generell Textfragmente verstehen, die mehrere Bedeutungen haben können).

Die Literatur kennt viele Beispiele für sehr unterschiedliche Arten von Mehrdeutigkeiten. Eines der bekanntesten Beispiele ist das Wort *Bank*: Es kann eine *Sitz-Bank* gemeint sein, ein *Bank-Geldinstitut*, ein *Bank-Gebäude* (in dem das entsprechende Geldinstitut untergebracht ist) etc.

Ein anderes Beispiel zeigt eine lexikalisch-syntaktische Mehrdeutigkeit:

George said that Henry left in his car

Eine Unklarheit betrifft hier die Tatsache *wessen Auto* gemeint ist - das von George oder das von Henry. Außerdem ist unklar, ob George sagte, daß *Henry mit seinem Auto weg fuhr* oder ob George *im Auto saß*, während er erklärte *daß Henry den Ort verließ* (vgl. [Allen, 1995, Kapitel 6]). Derartige Probleme sind nicht etwa Ausnahmefälle - tatsächlich weisen die meisten sprachlichen Äußerungen Mehrdeutigkeiten auf, die teilweise schwierig zu erkennen sind, da sich Menschen offenbar in den meisten Fällen der Unklarheit (und ihrer Auflösung) häufig nicht einmal bewußt sind.

Einige dieser Unklarheiten können mit einfachen Regeln oder Annahmen aufgelöst werden: beispielsweise wird üblicherweise angenommen, daß sich Anaphern (Rückverweise) auf das letzte, syntaktisch passende, Objekt beziehen - es dürfte sich demnach im obigen Beispiel um Henrys Auto handeln. In vielen Fällen jedoch können Mehrdeutigkeiten nur durch Anwendung von Hintergrundwissen über die Welt oder die Anwendung von Sprache oder von situativem Wissen aufgelöst werden. Beispielsweise können wir nur durch Kenntnis der relevanten Fakten entscheiden, daß es sich in dem Satz

Ich mag Filme und Pferderennen. Sehr gut gefällt mir „Dark Star“.

bei dem Begriff *Dark Star* um einen Film handelt, obwohl insbesondere die oben erwähnte Regel für Anaphern eigentlich eher auf einen Pferdenamen hindeuten würde.

Für Computersysteme sind solche Aufgaben - der Erkennung und Auflösung von Mehrdeutigkeiten (engl. *word sense disambiguation*, Abk. *WSD*) - sehr viel

schwieriger zu lösen, da in der Regel das benötigte Hintergrundwissen eben nicht verfügbar ist[1].

Dieses Kapitel soll zunächst einen Einblick in die WSD-Problematik geben, relevante Begriffe definieren und einen kurzen Überblick über historische Entwicklungen vermitteln. Anschließend werden wichtige Arbeiten der aktuellen WSD-Forschung sowie eine weitgehend anerkannte Evaluationsmethodik für WSD-Verfahren vorgestellt.

2.1 Grundlagen der WSD-Problematik

Ein für die erfolgreiche Verarbeitung von Texten wichtiges Teilproblem ist die korrekte Auflösung von Wort-Mehrdeutigkeiten, auch als Wort-Disambiguierung bezeichnet.

Unter der „Ambiguität“ (Mehrdeutigkeit) von Wörtern versteht man die Tatsache, daß ein Wort unter Umständen mehrere Bedeutungen haben kann. In der Regel unterscheidet man hierbei zwischen dem Fall der Homonymie – zwei bedeutungsunähnliche Wörter teilen sich dieselbe Form, z.B. „Bank“ im Sinne von „Sitzbank“ versus „Bank“ im Sinne von „Kreditinstitut“ – und dem der Polysemie – zwei Wörter sind sich bedeutungsähnlich bzw. zwei Varianten einer Kernbedeutung, z.B. denotiert „Fußball“ sowohl das bekannte Spiel als auch das Spielobjekt. In anderen Fällen kann ein Wort auf verschiedene Weise interpretiert werden, es ist „vage“. Es können auch mehrere Wörter dieselbe oder eine sehr ähnliche Bedeutung haben (Synonymie). Als „Auflösung von Ambiguitäten“ bezeichnet man im Umfeld der Computerlinguistik i.A. die Anwendung eines Verfahrens zur eindeutigen Zuordnung einer Bedeutungsvariante (engl. „word sense“) zu einem Wort und ggf. umgekehrt. In der natürlichen Sprache gibt es viele weitere Formen von Ambiguitäten, die an dieser Stelle aber nicht relevant sind und auch nicht betrachtet werden (beispielsweise syntaktische Mehrdeutigkeiten).

Die Auflösung von Mehrdeutigkeiten ist für Menschen häufig recht einfach – in der Regel erfolgt sie sogar unterhalb der Ebene bewußter Entscheidungen; für Computerprogramme dagegen handelt es sich um ein außerordentlich schwieriges Problem, da das benötigte Hintergrundwissen meist nur sehr lückenhaft oder gar nicht zur Verfügung steht. Wie von A. Kilgariff 1997 aufgezeigt, sind spezielle Fragestellungen der Mehrdeutigkeit aber auch für Menschen schwierig (vgl. [Kilgarriff, 1997a]), insbesondere im lexikographischen Umfeld. Die genaue und zeitgemäße Definition von Wortbedeutungen ist häufig schwierig, weil Wortbedeutungen selten starr festlegbar sind, sondern sich - häufig abhängig vom Kontext[2] - verändern

[1] Ide und Veronis bezeichnen das WSD-Problem sogar als „AI-complete“, d.h. als ein Problem, zu dessen Lösung zunächst alle schwierigen Probleme des Bereiches der künstlichen Intelligenz gelöst werden müssen, z.B. die Repräsentation von Allgemeinwissen, etc. (vgl. [Ide & Veronis, 1998])

[2] Kilgariff versteht unter „Kontext“ sowohl den textuellen Kontext, als auch den zeitbezogenen

können. Weitere Problemfelder ergeben sich, ebenfalls nach Kilgariff (vgl. [Kilgarriff, 1997b]) aus folgenden Punkten:

- Verschiedene Lexika geben unterschiedliche Bedeutungsvarianten bzw. Definitionen für bestimmte Wörter an.
- Es gibt keine theoretische Grundlage für das Konzept der „Bedeutungsvarianten“.
- Bedeutungsvarianten sind Konzepte von und in Wörterbüchern. Diese folgen jedoch u.U. sehr unterschiedlichen Zielen und Kriterien der Bestimmung von Bedeutungsvarianten, da Lexika und Wörterbücher ihrem Wesen nach soziale Artefakte sind und die spezifische Definition der Bedeutungsvarianten sich daher in der Regel an der Zielrichtung des Wörterbuches orientiert.

Obwohl diese Probleme bekannt sind, werden sie nach Kilgariff offenbar weitgehend ignoriert bzw. verschwinden, sobald innerhalb eines Anwendungskontextes eine Einigung auf eine klar definierte Menge von Bedeutungsvarianten besteht. Towell und Vorhees wiesen weiterhin auf die Problematik der Identifizierung von Synonymen hin (dies ist bedeutsam, da Synonyme die Gleichartigkeit von Konzepten verschleiern können, vgl. [Towell & Voorhees, 1998]).

„Wofür ist word sense disambiguation gut?“ (engl. „What is word sense disambiguation good for?“) fragt 1997 Kilgariff in [Kilgarriff, 1997b] - er bezeichnet WSD als relevant für Information Retrieval (als Alternative zu NLP Techniken), Machine Translation (als fundamentales Basisproblem, vgl. auch die Untersuchung von Carpuat et al hierzu [Carpuat & Wu, 2007]) und für Fragen der Lexikographie oder von Fragen des Natural Language Understanding (im Sinne von Allen, vgl. [Allen, 1995]).

WSD wurde zuerst als eine wesentliche Aufgabe im Umfeld der maschinellen Übersetzung von Texten in andere Sprachen erkannt - bereits 1949 wies Weaver auf die Problematik hin (vgl. Weaver in [Hutchins, 1999]); 1960 wurde WSD im Rahmen einer Machbarkeitsstudie zur maschinellen Übersetzung durch Bar-Hillel sogar als prinzipiell unlösbar dargestellt (vgl. [Bar-Hillel, 1960]).

Zu beachten ist, daß WSD in der Regel nicht als Anwendung einem Endbenutzer zur Verfügung steht, sondern eher ein wichtiges Element in übergeordneten anwendungsorientierten Systemen darstellt. Beispiele hierfür stellen die maschinelle Übersetzung dar, wie auch das Information Retrieval (z.B. das Finden von Webseiten, die einen Begriff in einer bestimmten Bedeutungsvariante enthalten), das Question Answering, d.h. die automatische Beantwortung von Fragen etc. (vgl. [Ide & Veronis, 1998]).

---

sozio-kulturellen Kontext, in welchem der Text steht. Damit kann sich die genaue Bedeutung eines Wortes sowohl in der Zeit verändern, als auch z.B. in verschiedenen gesellschaftlichen Schichten unterschiedliche Bedeutungsnuancen realisieren.

In der Literatur wird häufig zwischen *Word Sense Disambiguation* (Auflösung von Wort-Mehrdeutigkeiten, Abk. *WSD*) und *Word Sense Discrimination* (Unterscheidung von Wortbedeutungen) unterschieden, daher soll an dieser Stelle eine Abgrenzung beider Zielrichtungen erfolgen:

- **Word Sense Disambiguation** bezeichnet die Auswahl einer Lesart bzw. Bedeutungsvariante aus einer Menge von a priori erzeugten Bedeutungsklassen für ein Wort bzw. einen Begriff, z.B. aus Einträgen eines Lexikons, und deren Assoziation zum infrage stehenden Wort oder in Bezug auf die Übersetzung in eine andere Sprache (z.B. kann das englische Wort *chair* je nach Bedeutung in das Deutsche als „Stuhl“ oder als „Vorstand“ übersetzt werden) (vgl. Schütze 1998 [Schütze, 1998]). Die Repräsentation der Bedeutung eines Wortes erfolgt hierbei entweder in Bezug zu einem Wörterbuch (engl. *Dictionary*)[3] oder zu einer Übersetzung in eine andere Sprache (vgl. [Pedersen & Mihalcea, 2005]).

- **Word Sense Discrimination** bezeichnet die automatische Gruppierung (im Sinne von „Clustering“) von Wortinstanzen[4], wobei für jede Wortinstanz festgelegt wird, welcher Gruppierung sie zugeordnet wird (vgl. [Schütze, 1998]). Hier erfolgt die Bestimmung der Wortbedeutung durch den Bezug auf den lokalen Kontext (vgl. [Pedersen & Mihalcea, 2005]).

Ein etwas weniger rigoroser Begriff von WSD wurde 2003 von Ramakrishnan et al. vorgestellt: in [Ramakrishnan et al., 2004b] wird WSD nicht als Auswahl einer spezifischen Bedeutungsvariante aus einer Liste verstanden, sondern als Priorisierung einer Anzahl von Bedeutungsvarianten derselben Liste. Die einzelnen Varianten müssen hierbei nicht notwendigerweise orthogonal zueinander stehen, d.h. sie müssen sich nicht zwangsläufig gegenseitig ausschließen. Diese Deutung berücksichtigt die Tatsache, daß viele Wörter nicht nur einfache Homonyme voneinander sind, d.h. mit einer groben Bedeutungsunterscheidung, sondern der Unterschied zwischen ihnen viel feiner sein kann, so daß in einigen Fällen selbst für einen Menschen nicht in jeder textuellen Situation klar ist, welche Bedeutungsvariante intendiert ist. In der hier vorliegenden Arbeit wird dennoch lediglich der konventionelle Begriff von WSD eingesetzt, da die hier angegebenen Experimente ausgerichtet sind und eine Vergleichbarkeit zu anderen Verfahren gewährleistet sein soll. Prinzipiell kann der TSR-Ansatz die von Ramakrishnan et al. vorgeschlagene Begrifflichkeit von WSD aber gut unterstützen, da der paarweise Vergleich von TSRs eine Liste von Werten liefert, die in der von Ramakrishnan et al. vorgeschlagenen Weise genutzt werden kann.

[3] Als „Wörterbuch“ in diesem Sinne werden i.A. auch Thesauri oder semantische Netzwerke, z.B. WordNet verstanden.

[4] Eine „Wortinstanz“ kennzeichnet das Auftreten von Wörtern: eine Instanz des Wortes „von“ tritt beispielsweise in dieser Fußnote auf.

Es gibt eine große Vielzahl von Ansätzen, die Probleme der WSD zu lösen, eine Einteilung nach Kategorien erscheint daher sinnvoll. Eine mögliche Art und Weise, WSD bezogene Methoden zu kategorisieren, besteht in der systematischen Ausrichtung, die in den folgenden Abschnitten thematisiert wird.

2.2 Eine Charakterisierung der WSD-Forschung

In diesem Abschnitt sollen einige fundamentale Prinzipien der Wort-Disambiguierung vorgestellt, sowie eine exemplarisch aktuelle Forschungsbestrebungen und -richtungen angegeben werden.

Eine Charakterisierung des Forschungsstandes bezüglich der Auflösung von Wortmehrdeutigkeiten bis 1998 wurde bereits von Ide und Veronis in ihrem Artikel *Word Sense Disambiguation - State of the Art* vorgenommen (vgl. [Ide & Veronis, 1998]). In der hier vorliegenden Arbeit soll daher ein stärkerer Bezug auf methodische Unterschiede aktueller Arbeiten, besonders in Bezug auf den Einsatz webbasierter Verfahren genommen werden, als auf eher historische Gegebenheiten der WSD-Forschung.

Die WSD-Problematik, ebenso wie die hierauf bezogene Forschung, läßt sich unter anderem anhand folgender Faktoren charakterisieren:

- **Einsatz von Strukturbeschreibungen**: Unterscheidung nach der textuellen Beschreibungsebene (im Sinne der Semiotik und graphenorientierter Aspekte)
- **Art der primären Informationsquelle**: Unterscheidung nach dem Einsatz von primär kontextuellem Wissen im Vergleich zu primär externem Wissen
- **Einsatz von Trainingsmethoden**: Unterscheidung nach dem Grad der Überwachung, d.h. danach, ob zur Erstellung des eingesetzten Systemmodells Trainingsverfahren eingesetzt werden und, sofern dies der Fall ist, um welche Verfahren es sich handelt
- **Verknüpfung verwendeter Methoden**: Unterscheidung nach der Verwendung von „Single-Algorithm“ Methoden im Gegensatz zu Hybridmethoden – also danach, ob in einem System nur ein einziges WSD-bezogenes Verfahren eingesetzt wird, oder ob mehrere derartige Methoden parallel zum Einsatz kommen.

Hierbei handelt es sich nicht um eine erschöpfende Liste, aber die oben angeführten Faktoren stellen eine gute Basis für eine strukturierte Einordnung realer Ansätze und Systeme dar[5]. Agirre und Martinez haben in diesem Zusammenhang 2001

[5]Eine ausführlichere Darstellung von Wissensrepräsentationsmodellen für allgemeinere Verwendungszwecke findet sich bei Helbig (vgl. [Helbig, 2001])

verschiedene Informationstypen und Wissensquellen identifiziert (vgl. [Agirre & Martinez, 2001a]), die in den folgenden Unterabschnitten an entsprechender Stelle exemplarisch zur Verdeutlichung des entsprechenden Konzeptes vorgestellt werden.

In den folgenden Abschnitten sollen nun verschiedene Gesichtspunkte der eben vorgestellten Faktoren diskutiert und anhand von Beispielen aus der aktuellen Forschung erläutert werden.

2.2.1 Einsatz von Strukturbeschreibungen

In der traditionell orientierten Textverarbeitung wird davon ausgegangen, daß sich das Textverständnis über die im Text enthaltenen lexikalischen und syntaktischen Struktur- und Begriffsmerkmale erschließen läßt. Man spricht daher auch von einer merkmalsbasierten Betrachtungsweise der Textbedeutung. Ein wesentlicher Aspekt des Ansatzes ist der über die theoretische Fundierung strukturierte Verarbeitungsprozeß, dessen verschiedene für WSD relevante Ausprägungen in den folgenden Unterabschnitten beschrieben wird.

Die in den beiden folgenden Unterabschnitten diskutierten Methoden legen bei der Erstellung von Strukturbeschreibungen eine Verarbeitung symbolischer Informationen zugrunde. Hierbei können Verfahren, die auf sprachwissenschaftlichen Grundlagen basieren, von solchen unterschieden werden, die eine eher ontologische Ausrichtung[6] besitzen.

2.2.1.1 Linguistische Methoden

Im Bereich der WSD-Forschung gibt es sehr vielfältige regelbasierte, meist linguistisch orientierte Ansätze, die sich gut in die Teilbereiche der Semiotik einordnen lassen. Diese Teilbereiche lassen sich als aufeinander aufbauende Ebenen darstellen – wobei jede Ebene in der Regel Informationen aus der vorhergehenden Ebene nutzt – und sollen zur besseren Übersicht im Folgenden kurz erläutert werden:

Lexikalische Ebene. Auf der lexikalischen Ebene werden Informationen über die lexikalische Codierung der im Text enthaltenen Wörter gewonnen. Hierunter fallen beispielsweise die folgenden, von Agirre und Martinez erläuterten Informationstypen:

- Wortarten (engl. *Part of Speech*, abgek. *PoS*): Organisationsgrundlage für Bedeutungsvarianten (z.B. existiert innerhalb vieler Wissensbasen „bank“ als Verb, sowie als Nomen)

[6] Unter „ontologisch“ soll in diesem Zusammenhang „auf die Grundstrukturen des Seienden bezogen“ verstanden werden.

- Morphologie (engl. *morphology*): Relation der abgeleiteten Wörter zu ihren Stammformen. Beispielsweise beinhaltet WordNet 6 Bedeutungsvarianten für „agreement“ und 7 für „agree“.

Gerade Information über Wortarten werden häufig in WSD-Systemen eingesetzt, da wortartenspezifisches Wissen oft relativ eng an bestimmte Bedeutungsvarianten gekoppelt ist - dies wird auch in den weiter unten aufgeführten Beispielen gezeigt werden (vgl. [Preiss, 2004], [Yarowsky, 1993], etc.)

Beispielsweise verwenden Chanen und Patrick – neben weiteren Informationen – auch Wortarteninformation sowie morphologische Informationen, um Bedeutungsvarianten anhand der Gesamtheit der verfügbaren lexikalischen und syntaktischen Informationen in dem Satz, welcher das aufzulösende Wort enthält, unterscheiden zu können. Hierbei verwenden die Autoren einen musterbasierten Ansatz zur Disambiguierung von Wörtern – wobei auch für die Anwendung auf verallgemeinerbare Textsituationen abstrahierte Muster mit Wildcards in einem manuellen Verfahren gebildet werden können. Über einen Mustervergleich erfolgt dann die Auflösung der Mehrdeutigkeit (vgl. [Chanen & Patrick, 2004]).

Alternativ hierzu zeigen Lee et al. eine Methodik auf, Wortarteninformationen (in diesem Fall: Wortarten benachbarter Wörter, vgl. [Lee et al., 2004]) zu nutzen, um Bedeutungsvarianten anhand der lexikalischen Eigenschaften ihrer unmittelbaren textuellen Umgebung identifizieren zu können.

Syntaktische Ebene. Diese Ebene behandelt die Art und Weise, wie Sätze durch Wörter gebildet werden (vgl. [Chomsky, 1957, S. 11]). Syntaktische Modelle zeichnen sich in der Regel durch eine rekursive Struktur aus. Im vollen syntaktischen Analyseprozess (*parsing*) wird hierbei versucht, bereits auf der syntaktischen Ebene die dort aufzufindenden Mehrdeutigkeiten aufzulösen und die komplette syntaktische Struktur zu bestimmen (vgl. [Abney, 1996, Allen, 1995]). Dies erfolgt häufig durch die Anwendung einschlägiger Heuristiken oder - seltener - durch Einbeziehung semantischer Information (sofern verfügbar). Im Gegensatz hierzu verfolgt das sog. „partielle Parsing“ das Ziel, flache Strukturen mit geringerem Informationsgrad aber i.A. größerer Korrektheit zu erzeugen (vgl. [Abney, 1992]). Für die syntaktische Ebene definieren Agirre und Martinez einen übergeordneten Punkt, der verschiedene syntaktische Effekte zusammenfaßt:

- Syntaktische Hinweise (engl. *syntactic clue*s): zum Beispiel ist „eat“ in „take a meal“ intransitiv - ansonsten aber transitiv.

Ein bereits erwähnter Ansatz der aktuellen WSD-Forschung unter Verwendung syntaktischer Strukturmerkmale findet sich bei Chanen und Patrick: neben lexikalischen Informationen werden vor allem syntaktische Strukturinformationen (engl.

parse trees) in den Analyseprozess integriert. Hier wird gezeigt, wie Informationen aus verschiedenen Ebenen direkt ermittelt und in eine kanonische Form zusammengeführt werden können, um für WSD-Zwecke unmittelbar dienlich zu sein (vgl. [Chanen & Patrick, 2004]).

Niu et al. verwenden dagegen prädefinierte syntaktische Muster zur Erstellung von „Context Clustern“ zum Zwecke der Word Sense Discrimination - und in einem subsequenten Schritt für eine verb-bezogene WSD (vgl. [Niu et al., 2003]). Ein Beispiel hierfür wird von den Autoren wie folgt angegeben: Das Muster *V_S* erkennt Inhalte, in denen ein Verb mit einem logischen Subjekt zusammenhängt, z.B. erfolgt nach Eingabe von „*The algorithm designed by John works*“ die Ausgabe von *V_S(design, John)* und *V_S(work, algorithm)*. Diese Angaben werden - zusammen mit weiteren Faktoren - dazu verwendet, die o.a. Cluster zu erstellen. Nach Angabe der Autoren wurde die Performanz des Verfahrens mit SENSEVAL-2 (vgl. Abschnitt 2.3.2) Systemen verglichen (über die 29 verwendeten Verben) und kann mit diesen konkurrieren.

Neben Wortarteninformationen können auch syntaktische Beziehungen zwischen Satzkonstituenten im Rahmen eines WSD-Systemes genutzt werden, wie von Lee et al. vorgeschlagen wird (vgl. [Lee et al., 2004]). Die Autoren führen zu Verdeutlichung u.a. folgende Beispiele für den Prozeß der Merkmalsfindung an:

1. Fokuswort: *attention (noun)* (b) Satz: *He turned his attention to the workbench* (c) Syntaktisches Merkmal: <*turned, VBD, active, left*>

2. Fokuswort: *turned (verb)* (b) Satz: *He turned his attention to the workbench* (c) Syntaktisches Merkmal: <*he, attention, PRP, NN*[7]*, VBD, active*>

Die individuellen Komponenten der Merkmale stellen hierbei die Daten für die spätere Auswertung über eine Support Vector Machine (SVM), die dann die eigentliche Klassifizierungsaufgabe im Sinne des WSD-Prozesses übernimmt.

Die Nutzung syntaktischer Muster über ihre semantische Interpretation wird sowohl von Strappavara et al. (vgl. [Strapparava et al., 2004]), als auch von Navigli (vgl. [Navigli & Velardi, 2004]) eingesetzt. Die zugrunde liegende Annahme ist hierbei, daß (wie auch 2001 von Gomez (vgl. [Gomez, 2001]) festgestellt) die Syntax vieler Verben durch ihre Semantik bestimmt wird, sich daher umgekehrt die Semantik aus der syntaktischen Verwendungsweise schließen läßt. Dieser Sachverhalt kann an einem fast trivialen Beispiel deutlich gemacht werden: Das englische Word *fly* besitzt in der Wortart *Verb* u.a. die Bedeutung „fliegen“, während dem Wort in der Wortart *Nomen* u.a. die Bedeutung „Fliege“ beigemessen wird.

Ein ähnlicher Ansätze findet sich bei Fernandez, hier handelt es sich um einen nicht-überwachten Ansatz, in welchem die WordNet-Glossen genutzt werden, um

[7] Bei den angegebenen Kürzeln NN, VBD, etc. handelt es sich um lexikalisch-syntaktische Informationen, beispielsweise zur Wortart.

syntaktische Muster zu erzeugen, wobei Wildcards für Pronomen und Inhaltswörter eingesetzt werden um eine gewisse Allgemeingültigkeit zu erreichen (vgl. [Férnandez-Amorós, 2004]).

Weitere syntaktische Verfahren, die mit mehreren Sätzen von Merkmalen arbeiten, finden sich bei Lin, dessen Ansatz auf einem Neuronalen Netz basiert, dessen Eingabedaten aus den syntaktischen Daten besteht, die ein Parser liefert (vgl. [Lin, 2000]): beispielsweise werden aus dem Text

> *„The guerrillas' first urban offensive, which has lasted three weeks so far and shows no sign of ending, has shaken a city lulled by the official propaganda."*

syntaktische Merkmale wie *(V shake), (V:subj:N offensive), (V:compl:N city), . . .* generiert. Diese werden genutzt, um im Zusammenhang mit einem automatisch generierten Thesaurus die Konfiguration des Neuronalen Netzes zu erzeugen, z.B. so wie das folgende Merkmal für das Wort *offensive* mit Gewichten des Netzes assoziiert wird (ähnliche Werte gleicher Wörter werden als „semantische Ähnlichkeit" definiert):

> *offensive: attack 0.183; assault 0.168; raid 0.154; effort 0.153; campaign 0.148; crackdown 0.137; strike 0.129; bombing 0.127; move 0.124; invasion 0.123; initiative 0.121; . . . conflict 0.072; . . .*

Syntaktische Abhängigkeiten werden in ähnlicher Weise wie bei den oben exemplarisch angeführten Verfahren – zumindest als Subkomponente – von vielen aktuellen Systemen genutzt (vgl. [Agirre & Martínez, 2004, Cabezas et al., 2004, Ciaramita & Johnson, 2004, Decadt et al., 2004, Escudero et al., 2004, Mohammad & Pedersen, 2004, Lee et al., 2004, Vázquez et al., 2004]).

Semantische Ebene. Das „Verstehen" der durch die Syntaxanalyse erzeugten Strukturen erfolgt in einer semantische Ebene, in welcher ein Modell der Textbedeutung konstruiert wird, die so genannte logische Form (vgl. [Allen, 1995]). Notwendig für die Überführung der syntaktischen Strukturinformation in eine Bedeutungsinformation ist ein i.d.R. komplexes Bezugssystem. Hierzu muß die Bedeutung der Wörter, wie auch der Textstrukturen verstanden werden. Eine wesentliche Funktion hat hierbei das Prinzip der „semantischen Komposition", welches besagt, daß sich die Gesamtbedeutung der zusammengefügten Textstrukturen - auch der Sätze - aus der Bedeutung ihrer Komponenten ergibt (vgl. [Frege, 1892]).

Im Zusammenhang mit der Wortsemantik definieren und erklären Agirre und Martinez einige Informationstypen, die für WSD-Verfahren genutzt werden:

- Taxonomische Organisation: z.B. zwischen „chair" und „furniture"

- Situativer Kontext: Beispielsweise können sich „chair“ und „waiter“ in einem gemeinsamen situativen Kontext befinden, etwa in der Beschreibung eines Restaurantbesuches

- Wissensdomäne: z.B. ist innerhalb der Domäne „Sport“ die wahrscheinlichste Bedeutung für „Schläger“ die von „Tennis Schläger“ (Im Vergleich zur Bedeutung im Sinne eines gewalttätigen Menschen).

- Thematischer Kontext: Über den situativen Kontext hinausgehender Bezug, der einen prototypischen thematischen Kontext herstellt, z.B. zwischen „bat“ und „baseball“ anstelle, beispielsweise, von „bat“ und „animal“; aber auch von „bat“ und „cricket“. An letzterem Beispiel wird auch der Unterschied zum Typ Wissensdomäne deutlich: „baseball“ und „cricket“ werden üblicherweise der Domäne „Sport“ zugeordnet und wären demnach über die ausschließliche Domänenzuordnung nicht unterscheidbar.

- Argumentstruktur innerhalb von Sätzen: z.B. besteht zwischen „Hund“ und „jagte“ in „Der Hund jagte den Postboten“ ein derartiger struktureller Zusammenhang

- Semantische Rolle: beispielsweise nimmt das Objekt von „eat“ in „the bad news will eat him“ die semantische Rolle *Experiencer* ein.

Vor allem das verbreitete Verfahren von Lesk nutzt intensiv semantische Relationen: basierend auf der Annahme, daß benachbarte Wörter häufig denselben thematischen Kontext teilen, werden die Definitionen der Bedeutungsvarianten dieser Wörter aus einem Wörterbuch extrahiert und miteinander verglichen (wobei der Vergleich in der ursprünglichen Version des Algorithmus über die Anzahl gleicher Wörter in den Einträgen der Bedeutungsvarianten verfolgt). Lesk stellt in seinem Artikel (vgl. [Lesk, 1986]) die Frage „how to tell a pine cone from an ice cream cone“ und gibt gleich die Antwort:

> Für das Wort *Pine* gibt es im Wörterbuch *Oxford Advanced Learner's Dictionary of Current English* zwei Hauptbedeutungen: „kind of evergreen tree with needle-shaped leaves.. .“ und „waste away through sorrow or illness...“. *Cone* hat dagegen im selben Buch drei distinkte Bedeutungen: „solid body which narrows to a point . . . “, „something of this shape whether solid or hollow. . . “ und „fruit of certain evergreen trees. . . “.
>
> Die Wörter *evergreen* und *tree* treten hierbei in jeweils 2 Bedeutungsvarianten auf, so daß, wenn die Wörter *pine cone* zusammen auftreten, ein Programm hieraus schließen kann, daß es sich bei dem gesuchten Begriff um den Baum und seine Frucht handelt.[8]

[8]

In der aktuellen Forschung erzielen Strappavara et al., Agirre und Martinez, Vazques et al. und Escudero et al. (vgl. [Strapparava et al., 2004], [Agirre & Martínez, 2004], [Vázquez et al., 2004] und [Escudero et al., 2004]) durch die zusätzliche Einbeziehung von Domäneninformationen eine höhere Systemleistung. Hierbei werden u.a. Verfahren eingesetzt, die Instanztext und Texte der Bedeutungsvarianten (Glossen) in Beziehung zu Themenbereichen setzen und hierdurch eine bessere Disambiguierung ermöglichen. Strappavara verwendet eine „Domain Relevance Estimation“ genannte Technik, um thematische informationen im Text zu identifizieren. Agirre und Martinez verwenden dagegen explizite Domäneninformationen aus der WordNet Domains Datenbasis (vgl. [Magnini & Cavagli, 2000]).

Atserias et al. verwenden schließlich eine ganze Anzahl verschiedener Quellen für die Gewinnung von Domäneninformationen, welche die Grundlage für das MEANING Projekt bilden (Ziel desselben ist es, aus vielen verschiedenen Informationsquellen eine Datenbasis für WSD-Anwendungen zu konstruieren, vgl. [Atserias et al., 2004]).

Taxonomische Relationen werden dagegen in den Verfahren von Garcia-Vega et al. und Pedersen für WSD-Zwecke eingesetzt: Es können entsprechende Informationen aus der WordNet-Datenbank - insbesondere aus Hyponym/Hypernym-Relationen - in einer Weise genutzt werden, die in gewisser Weise dem bereits oben angeführten Verfahren von Lesk entspricht, im Unterschied dazu wird allerdings die Kette der Hypernyme anstelle des Wörterbucheintrages selbst verwendet (vgl. [García-Vega et al., 2004, Pedersen, 2004]) – in ähnlicher Weise wird WordNet (in Kombination mit einer manuell erstellten Ontologie) durch Baldwin et al. eingesetzt (vgl. [Baldwin et al., 2008]). Grundlage dieser Verfahren sind die Hypothesen daß die starke Abhängigkeit des Lesk'schen Algorithmus von den verwendeten Wörtern durch die Verwendung der Hypernyme ein wenig abgeschwächt wird und daß unterschiedliche Bedeutungsvarianten auch unterschiedliche Hypernymketten aufweisen.

Pragmatische Ebene. Die auf der semantischen Ebene erstellte logische Form wird im Rahmen der pragmatischen Ebene in einem situativen und textuellen Kontext interpretiert, indem weitere Wissensquellen hinzugezogen werden[9]. Die hierdurch gewonnenen Erkenntnisse werden dann kontextabhängig in eine Reaktionshandlung, z.B. eine Antwort umgesetzt – sofern der dem Ausgangstext zugrunde liegende Sprechakt eine Frage war (Literatur zur Sprechakttheorie liefert die Weiterführung des Austin'schen Konzeptes der sprachlichen Handlung durch Searle [Searle, 1997]).

Im Orginal durch Lesk (vgl. [Lesk, 1986]), Übersetzung durch den Autor der vorliegenden Arbeit

[9]Manchmal wird zwischen Semantik und Pragmatik noch eine Diskursebene definiert; Erläuterungen hierzu sollen an dieser Stelle jedoch wegen ihrer sehr schwachen Relevanz für die WSD-Thematik entfallen.

Agirre und Martinez geben für die Nutzung der pragmatische Ebene in WSD-Systemen lediglich eine vage Begriffsbestimmung an:

- Pragmatische Aspekte: manchmal wird ein deduktiver Prozeß zur Disambiguierung benötigt (z.B. um im Satz „Sie traf mit dem Hammer den Nagel aber der Kopf flog weg“ von „Kopf“ auf „Nagelkopf“ - oder „Hammerkopf“ schließen zu können)

- Pragmatische Heuristiken, z.B. die der „Auswahlpräferenz“ (Präferenzen der Wortbedeutung); beispielsweise präferiert „eat“ in „take a meal“ Menschen als Subjekt. Als Grundlage dieser Heuristiken können auch nichtlinguistische statistische Methoden eingesetzt werden, z.B. kann für das Wort „dish“ aus unvorbereiteten Korpora die Information gewonnen werden, daß „dish“ häufig im Zusammenhang mit dem Wort „wash“ steht, aber häufiger noch im Zusammenhang zum Begriff „food“ (vgl. [McCarthy, 1997, Agirre & Martinez, 2001b]).

  Eine weitere wichtige Heuristik ist die „Frequenz der Bedeutungsvariante“ (engl. *predominant sense heuristic*): die am häufigsten verwendete Bedeutungsvariante ist üblicherweise für eine große Majorität von Exemplaren in realistischen Texten die korrekte Lesart.

Die oben angeführten Ebenen sind nicht notwendigerweise als sequentiell getrennte Module zu verstehen, vielmehr beeinflussen sich Erkenntnisse der einzelnen Stufen häufig gegenseitig in hohem Maße. Insbesondere das vollständige Parsing von Sätzen und ganzen Texten beinhaltet in realen Systemen eine starke Nutzung semantischer Operationen (vgl. u.a. [Schmid, 1996, S. 261]).

Insgesamt läßt sich feststellen, daß sämtliche Ebenen der Semiotik – entweder für sich betrachtet, oder in Kombination untereinander – für WSD-Zwecke eingesetzt werden. Ein Vorteil derartiger Ansätze ist sicherlich das Bestehen einer weitgehend akzeptierten theoretischen Basis.

2.2.1.2 Graphenbasierte Ansätze

In diesem Abschnitt sollen einige wichtige graphenbasierte Verfahren vorgestellt werden. Diese Ansätze können zwar in der Regel in ähnlicher Weise wie die regelbasierten Methoden nach sprachwissenschaftlichen Gesichtspunkten geordnet werden; an dieser Stelle soll aber vor allem die vielfältige Verwendbarkeit von Graphenalgorithmen für WSD-Zwecke in der aktuellen Forschung gezeigt werden (Eine aktuelle Untersuchung der Thematik mit explizitem WSD-Bezug findet sich bei Navigli in [Navigli & Lapata, 2007]). Subsymbolische Methoden, vor allem Konnektivistische, spielen ebenfalls eine Rolle in der (älteren) WSD-Forschung, vor allem in Bezug auf semantische Netzwerke (vgl. [Quillian, 1969, Wilks, 1975]

in [Ide & Veronis, 1998]). Sie sollen hier aber wegen ihrer thematischen Entfernung zu dem in dieser Arbeit diskutierten TSR-Ansatz nicht weiter erörtert werden.

Verbreitet sind vor allem graphenbasierte Verfahren mit eingeschränktem Vokabular, vgl. z.B. [Ginter et al., 2005]. Aufgrund des Bezuges zur vorliegenden Arbeit sollen hierbei vor allem kontextbasierte Methoden hervorgehoben werden, bei denen das Basiswissen aus einer externen Datenquelle entnommen wird, i.d.R. einem Wörterbuch, einem Thesaurus (häufig verwendet wird hierzu die elektronisch verfügbare Version von Roget's Thesaurus) oder einer lexikalischen Datenbank (z.B. WordNet).

Exemplarisch dafür sind die im Folgenden dargestellten Verfahren aus aktuellen Arbeiten:

Einer der wichtigsten Ansätze in diesem Bereich wird als „Extended Gloss Overlap“ bezeichnet. Hierbei handelt es sich um eine Weiterentwicklung des Algorithmus von Lesk (vgl. Abschnitt 2.2.1.1 auf Seite 26) durch Banerjee und Pedersen (vgl. [Banerjee & Pedersen, 2003]): Die Wörterbuchdefinition jeder Bedeutungsvariante wurde um Glossen verwandter Wörter aus WordNet bereichert bzw. um diesen ergänzt, um das Informationsvolumen für die Disambiguierung der Zielwörter zu erhöhen. Dadurch soll die starke Abhängigkeit des originalen Algorithmus von individuellen Formulierungen bzw. spezifischen Wortokkurrenzen abgeschwächt werden und die Robustheit des Ansatzes insgesamt erhöht werden.

Gelbukh fügte 2005 dem Lesk-Algorithmus in ähnlicher Weise einige Erweiterungen hinzu (vgl. [Gelbukh et al., 2005]). Seine Idee ist hierbei, nicht nur das zu untersuchende Wort anhand seiner Wörterbuchdefinition zu disambiguieren, sondern alle Wörter des lokalen Kontextes über die Glossen ihrer jeweiligen Bedeutungsvarianten einzubeziehen und über den Lesk'schen Algorithmus auf Okkurrenzen gleicher Wörter miteinander zu vergleichen, wie in Abbildung 2.1 von Gelbukh selbst dargestellt wird. Wegen des hohen Rechenvolumens wurden verschiedene

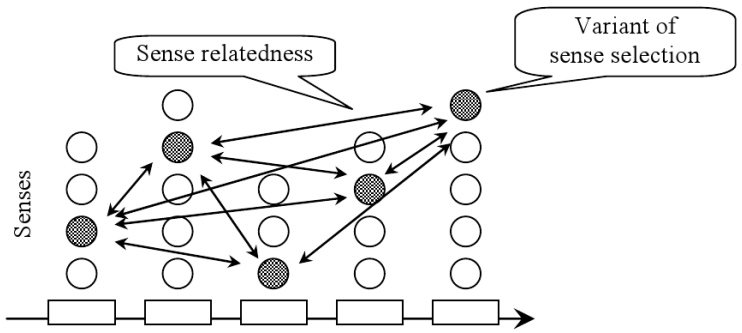

Abbildung 2.1: Lesk-Algorithmus mit WordNet Bedeutungsvarianten (schematisch)

Heuristiken zur Komplexitätsbegrenzung eingesetzt, wobei eine Effizienzsteige-

rung bis zur ca. zweifachen Geschwindigkeit erreicht werden konnte.

Auch für die Analyse und Bildung von lexikalischen Ketten (engl. "lexical chains", vgl. [Morris & Hirst, 1991]) ist WSD oft eine grundlegende Voraussetzung. Ein solcher Einsatz wird z.B. von Galley und McKeown 2003 in [Galley & McKeown, 2003] wie folgt beschrieben: Ein Graph wird so aus dem Quelltext gebildet, daß die Wörter durch die Knoten des Graphen repräsentiert werden. Die Knoten werden dann expandiert, indem sie durch die jeweilige WordNet-Bedeutungsvariante ersetzt werden, also z.B. *bank* durch *financial institution*. Wird über den Lesk'schen Algorithmus eine Relation zwischen zwei aufeinander folgenden Wort-Knoten festgestellt, wird eine Kante zwischen diesen Knoten eingefügt und über die Stärke der Relation gewichtet. Dann werden alle Werte der Kanten zweier jeweils benachbarter Knoten miteinander verglichen. Der schließlich entstehende Pfad der höchstwertigen Kanten stellt die disambiguierte Interpretierung des Ausgangstextes dar. Dieses Verfahren ähnelt dem oben angeführten Extended-Gloss-Overlap-Ansatz sehr stark, ebenso wie der von Mihalcea 2005 vorgestellte, ebenfalls Lesk-ähnliche graphische WSD Algorithmus: Aus Wortsequenzen werden Graphen gebildet, deren Knoten durch Wörter definiert werden, Kanten durch syntaktische Abhängigkeiten zwischen den Wörtern und Kantengewichtungen durch einen Rankingalgorithmus über den Graphen bestimmt werden (vgl. [Mihalcea, 2005]). Die Ähnlichkeit zwischen Bedeutungsvarianten und Textsequenzen wird über eine Berechnung der Kantengewichte in Bezug auf die Glossen der WordNet-Bedeutungsvarianten festgestellt.

Ein strukturell andersartiger Ansatz ist das „Structural Semantic Interconnections" (SSI) -Verfahren. SSI wurde 2004 durch Navigli und Velardi eingeführt (vgl. [Navigli & Velardi, 2004]) und basiert auf der Erkennung strukturierter semantischer Muster, die durch Graphen für Bedeutungsvarianten gegeben sind. Dieser Graph beschreibt semantische Verknüpfungen zwischen Konzepten, und wird über eine manuell erstellte Grammatik aus einer Vielfalt von Wissensquellen erstellt (u.a. WordNet und annotierten Korpora wie SemCor). Hierbei werden die Knoten des Graphen aus WordNet-Bedeutungsvarianten gebildet und Kanten durch semantische Relationen zwischen diesen (z.B. *kind-of*, *part-of*, etc.). Navigli und Velardi stellen diesen Sachverhalt am Beispiel der Bedeutungsvariante #1 des Wortes „market" in einer Abbildung dar, die hier als Abbildung 2.2 auf der nächsten Seite übertragen ist. Über ein Verfahren der Mustererkennung können die Graphen verschiedener Bedeutungsvarianten miteinander verglichen werden. Wie 2007 von Chin et al. gezeigt wird, kann dieses Verfahren sogar eingesetzt werden, um Ontologien zu popularisieren (vgl. [Chin et al., 2006]).

Ein Vergleich der oben angeführten Methoden findet sich bei Brody et al. (vgl. [Brody et al., 2006]), wobei SSI als das leistungsfähigste Verfahren bewertet wurde.

Zusammenfassend läßt sich feststellen, daß auch graphenorientierte Methoden – oft auf dem Algorithmus von Lesk basierend – erfolgreich für WSD-Zwecke ein-

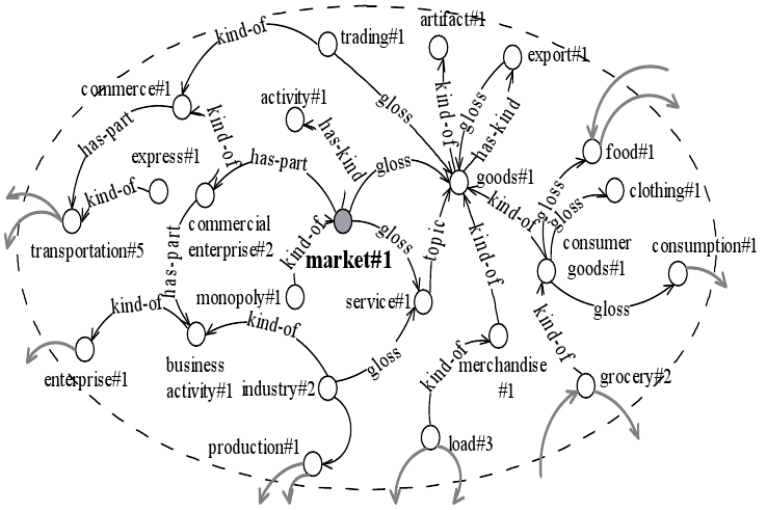

Abbildung 2.2: Graphenbasierte Repräsentation nach Navigli und Velardi

gesetzt werden.

2.2.2 Art der Informationsquelle

In der WSD-bezogenen Literatur wird in der Regel zwischen „datengesteuerten Ansätzen“ (korpusbasierten bzw. kontextuellen Methoden) und „wissensbasierten Ansätzen“ (Methoden, die externe Informationen aus Ontologien, maschinenlesbaren Wörterbüchern, etc. beziehen) unterschieden. Sehr viele reale Systeme verwenden tatsächlich einen hybriden Ansatz, in welchem verschiedene Einzelmethoden kombiniert werden. Diese Vorgehensweise wird in Abschnitt 2.2.4 erläutert und dargestellt werden.

2.2.2.1 Datengesteuerte Ansätze

Unter „datengesteuerten“ Methoden sollen hier solche verstanden werden, die Informationen unmittelbar aus verfügbaren Korpora beziehen, dies erfolgt im Wesentlichen über die Anwendung statistischer Berechnungen zur Häufigkeit von Kollokationen u.a.. Diese Verfahren werden auch zur Word Sense Discrimination angewendet (vgl. z.B. [Salton & Yu, 1975, Schütze, 1995, Schütze, 1998]).

Weil die diskutierten datengesteuerten Ansätze sehr häufig die Kenntnis von Grundlagen der linearen Algebra, insbesondere aus dem Bereich der Vektorrechnung voraussetzen, sollen diese im Folgenden kurz dargestellt werden.

Relevante Grundlagen der linearen Algebra. Die folgenden Inhalte sollen eine Basis für spätere Darstellungen herstellen. Weil es sich um verbreitete Grund-

lagen handelt werden in vielen Fällen nicht explizite Quellen angegeben. Maßgeblich Literatur war jedoch die „Kleine Enzyklopädie Mathematik“ von Gellert et al. ([Gellert et al., 1971]), das Kapitel 5 („Word-Vectors and Search Engines“) des Buches „Geometry and Meaning“ von Dominic Widdows [Widdows, 2004], sowie verschiedene Grundlagenskripte aus dem Bereich Mathematik in der Informatik.

Vektoren werden häufig in der Listenform (z.B. $\overrightarrow{A} = (2;5)$) notiert, wobei die Unterscheidung zu einem skalaren (d.h. einfach numerischen) Wert durch den übergestellten Pfeil erfolgt und die Elemente des Vektors per Semikolon voneinander abgegrenzt werden. Im Folgenden wird die Pfeildarstellung aus Gründen der Übersichtlichkeit in der Regel weggelassen werden und nur zur Abgrenzung zu Skalarwerten verwendet werden, sofern eine Verwechslungsgefahr besteht.

Vektoren werden genutzt, um (numerische) Daten geordnet zusammenzufassen, die dasselbe Objekt oder Konzept betreffen. Diese Daten werden - aufgrund der Nähe zur Geometrie (eine frühe Begriffsbestimmung des Vektors definiert diesen als gerichtete Strecke) - als „Koordinaten“ bezeichnet. Die Anzahl der Koordinaten wird als „Dimension“ bezeichnet - beispielsweise besitzt der Vektor $\overrightarrow{B} = (1;2;3;1)$ vier Dimensionen.

Man kann Vektoren miteinander addieren und einzelne Zahlen mit ihnen multiplizieren, letzteres wird als „Skalarmultiplikation“ bezeichnet (vgl. [Widdows, 2004, S. 132ff]).

DEFINITION 1 : Die *Vektoraddition* ist durch die Addition der entsprechenden Vektorkomponenten definiert:

$$\overrightarrow{A} + \overrightarrow{B} \quad = \quad (a_1 + b_1; a_2 + b_2; \ \ldots ; a_n + b_n) \tag{2.1}$$

DEFINITION 2 : Die *Skalarmultiplikation* ist äquivalent über die Multiplikation der Vektorkomponenten mit einem skalaren Wert definiert:

$$\lambda \cdot \overrightarrow{A} = (\lambda \cdot a_1; \lambda \cdot a_2; \ \ldots ; \lambda \cdot a_n) \tag{2.2}$$

DEFINITION 3 : Unter „*Vektormultiplikation*“ soll an dieser Stelle lediglich das *Skalarprodukt* verstanden werden - hierbei entsteht durch die Verknüpfung zweier Vektoren ein Skalar:

$$\overrightarrow{A} \cdot \overrightarrow{B} = (a_1; a_2; ...; a_n) \cdot (b_1; b_2; ...; b_n) = a_1 b_1 + a_2 b_2 + ... + a_n b_n$$

Vektoren sind die Elemente von Vektorräumen (engl. vector spaces). Vectorräume sind algebraische Strukturen, die die Menge V der Vektoren beinhaltet und zudem noch die Vektoraddition und die Skalarmultiplikation. Hierbei ist die Addition eine innere assoziative Verknüpfung auf V (d.h. addierte Vektoren gehören wieder zu

V und die Addition folgt dem Assoziativgesetz). Es gibt bezüglich der Addition ein neutrales Element, den Nullvektor $\overrightarrow{0}$, für den gilt $\overrightarrow{v} + \overrightarrow{0} = \overrightarrow{0} + \overrightarrow{v} = \overrightarrow{v}$ und zu jedem Vektor $\overrightarrow{v}$ läßt sich ein Gegenvektor $-\overrightarrow{v}$ konstruieren, so daß gilt $\overrightarrow{v} + (-\overrightarrow{v}) = \overrightarrow{0}$.

Für Vektoren lassen sich verschiedene Ähnlichkeits- und Distanzfunktionen definieren, die in den weiteren Kapiteln von Interesse sind. Besonders das Kosinus-Maß ist von Bedeutung, da es Ähnlichkeiten zwischen Vektoren unabhängig von der Länge einzelner Vektoren bestimmt. Die Abhängigkeit von der Vektorlänge ist ein im Anwendungsgebiet dieser Arbeit häufiger und störender Einflußfaktor - dies betrifft viele andere Ähnlichkeitsmaße, z.B. die Euklidische Distanz und Distanzrechnungen durch einfache Komponentenmultiplikation (vgl. [Widdows, 2004, Kapt. 5] zur Eignung verschiedener Ähnlichkeitsmaße).

Zuvor müssen allerdings noch einige Hilfsgrößen definiert werden:

DEFINITION 3 : Unter der *Länge* oder *Norm* eines Vektors wird der um die Richtungskomponente bereinigte Betrag des Vektors verstanden:

$$\| a \| = \sqrt{\sum(a_i^2)} \tag{2.3}$$

DEFINITION 4 : Der *Einheitsvektor* $\hat{a}$ eines Vektors a entsteht durch Teilung der einzelnen Koordinaten durch die Länge des Vektors:

$$\hat{a} = \frac{a}{\| a \|} \tag{2.4}$$

Einheitsvektoren besitzen die einheitliche Länge 1, aber unterschiedliche Richtungskomponenten. Vektoren mit der Länge 1 heißen auch „normiert“.

Mit diesen Hilfsgrößen läßt sich das Kosinus-Ähnlichkeitsmaß für Vektoren definieren

DEFINITION 5 : Die Kosinusfunktion für Vektoren betrachtet den Winkel zwischen zwei Vektoren und ist wie folgt definiert:

$$cos(a,b) = \frac{a \cdot b}{\| a \| \cdot \| b \|} \tag{2.5}$$

Eine letzte Definition ermöglicht die Repräsentation von sehr hoch dimensionierten Vektoren, in denen die meisten Werte aber gleich 0 sind:

DEFINITION 6 : Ein Indexvektor ist ein Vektor mit sehr vielen Dimensionen, in welchem nur Werte angeführt werden, die von Null verschieden sind. Hierbei wird der Index des Wertes mit angegeben.

Beispiel: Dem Vektor $A = (0;0;1.0;0;0;0;0;2.5;0;0;0;0;5.75;0;0)$ entspricht der Indexvektor $A' = (\mathbf{3}, 1.0; \mathbf{7}, 2.5; \mathbf{12}, 5.75)$[10].

In Problemen der datengesteuerten Textverarbeitung ist die Verwendung derartiger Indexvektoren in produktiven Anwendungen und für sehr hoch dimensionale Vektoren üblich (vgl. [Witten & Frank, 2005]).

Anwendung in Term-Dokument-Matrizen. Tatsächlich beruhen viele Anwendungen der Computerlinguistik auf den mathematischen Grundlagen der linearen Algebra - seit Einführung des Prinzips der Term-Vektor-Modelle 1958 durch Luhn [Luhn, 1958] wurden vielfältige Methoden zur Gewichtung bzw. Priorisierung einzelner Terme und auch zur Bildung von Termgruppen (bzw. Clustern, Kontexten) entwickelt. Der prinzipielle Ansatz stellt bis heute eine wichtige Basis vieler Verfahren im Bereich der WSD-relevanten Textverarbeitung dar (beispielsweise nutzt auch die Majorität der in der SENSEVAL-3 Evaluation eingesetzten Verfahren derartige Term Vektor Modelle).

Der wohl am weitesten verbreitete und etablierte Vektorraum-Ansatz wird in diesem Abschnitt vorgestellt: Hierbei geht es darum, aus der Häufigkeit des Auftretens von Wörtern bestimmte Rückschlüsse auf deren übliche Verwendung zu ziehen, oft in Verbindung mit Verfahren des maschinellen Lernens. Diese Methodik wird auch als *BOW*-Ansatz bezeichnet. Hiermit soll verdeutlicht werden, daß lediglich das *Auftreten* von Wörtern von Interesse ist, nicht aber strukturelle Texteigenschaften.

Eine für flache Verfahren nützliche und vielfältig genutzte Repräsentationsweise für Texte ist die sog. „Term-Dokument-Matrix“ (engl. *term-document-matrix*, vgl. [Widdows, 2004, Salton & McGill, 1983]). Hierbei können Wörter vereinfacht als „Terme“ bezeichnet werden und Texte entsprechend als „Dokumente“.

Eine solche Matrix wird folgendermaßen aus einer Zusammenstellung von Dokumenten gebildet: Aus der Menge der in den einzelnen Dokumenten auftretenden Wörter werden Indexvektoren gebildet, so daß jedes Dokument durch einen derartigen Vektor repräsentiert wird. Die Dokumente müssen hierbei Text enthalten und über eine eindeutige Referenz (einen Namen oder eine Nummer) verfügbar sein.

Die Werte der Vektorelemente können binär sein (z.B. mit der Semantik „Wort kommt im Dokument vor“ versus „Wort kommt im Dokument nicht vor“), aus der Menge der natürlichen Zahlen (z.B. Anzahl des Auftretens eines bestimmten Wortes) oder aus der Menge der reellen Zahlen (z.B. durch Normalisierung von natürlichzahligen Vektoren) stammen.

So seien zum Beispiel die folgenden Dokumente gegeben:

[10] Die Indexkomponenten sind der Deutlichkeit halber in Fettdruck aufgeführt

Dokument-Nr.	**Inhalt**
D_1	**Der Mann geht mit dem Hund**
D_2	**Der Hund jagt den Fuchs**
D_3	**Der Mann jagt den Fuchs wenn der Hund den Fuchs jagt**

Üblicherweise lassen sich nur wenige (diskriminierende) Informationen aus Bestimmungsworten und anderen, sehr häufig verwendeten Wörtern oder Ausdrücken gewinnen. Diese Wörter werden häufig als „Stopwörter“ bezeichnet und in der späteren Vektorrepräsentation fortgelassen.

Die im Beispiel verwendeten Wörter lassen sich wie folgt auf eine Indexmenge abbilden:

Index $w \in \mathbb{N}$	Wort
$w = 1$	Mann
$w = 2$	Hund
$w = 3$	Fuchs
$w = 4$	geht
$w = 5$	jagt

Dem Dokument D_3entspricht damit der Binärvektor

$$d_3 = (1;1;1;0;1)$$

und der Vektor (Elemente aus $\mathbb{N}$)

$$d_3 = (1;1;2;0;2)$$

Derartige Dokumentvektoren können in einer Matrix zusammengefaßt werden, sofern sie vom selben Typ sind - also Elemente derselben Wertemenge enthalten und dieselbe Dimension besitzen.

Für das obige Beispiel gilt dann die folgende Term-Dokument-Matrix:

Index	d_1	d_2	d_3	**Wort**
1	1	0	1	Mann
2	1	1	1	Hund
3	0	1	2	Fuchs
4	1	0	0	geht
5	0	1	2	jagt

Wie man sieht, können die Spaltenvektoren - wie beschrieben - den Dokumenten zugeordnet werden. Zudem können aber die Zeilenvektoren den ihnen entsprechenden Wörtern (Termen) zugeordnet werden. Damit wird beispielsweise das Wort Mann durch den sog. *Wortvektor* $w_1 = (1;0;1)$ repräsentiert. Das Element i eines Wortvektors w wird als *Termfrequenz* innerhalb des Dokumentes d_i definiert.

Zusätzlich zu Wortelementen werden in praktischen Verfahren oft auch zusätzliche syntaktische oder lexikalische Elemente integriert, z.B. Wortarten–Informationen.

Verschiedene Faktoren sprechen für eine Normalisierung der o.a. Vektoren - z.B. die Tatsache, daß Vektoren mit hohen Werten gegenüber Vektoren mit geringen Werten sich in der Regel zu stark in Ergebnissen auswirken (vgl. [Widdows, 2004, Kapitel 5]). Daher soll im weiteren Verlauf dieser Arbeit immer von Vektoren mit dem Wertebereich der reellen Zahlen ausgegangen werden - falls nicht anders angegeben.

Ein weiterer wichtiger Aspekt ist die Abhängigkeit der Relevanz einzelner Wortvektoren sowohl vom assoziierten Wort, als auch von bestimmten Dokumenteigenschaften. Um derartige Abhängigkeiten bestimmen zu können, wird häufig statt der einfachen Termfrequenz eine Kombination aus Term- und Dokumentfrequenzen eingesetzt, die oft nach dem sog $TF \times IDF$-Maß berechnet werden (vgl. [Aas & Eikvil, 1999], [Baeza-Yates & Ribeiro-Neto, 1999] etc.) – dieses Maß soll eine ungefähre Vorstellung davon vermitteln, wie sich die Bedeutung eines Wortes für ein Dokument in Beziehung zur allgemeinen Bedeutung des Wortes für eine Menge von Dokumenten verhält.

Die Berechnung des $TF \times IDF$-Maßes erfolgt nach der folgenden Formel (vgl. [Baeza-Yates & Ribeiro-Neto, 1999]):

$$tf \cdot idf(t,d) = \frac{freq(t)}{max(freq(d))} \cdot log(\frac{N}{d(t)})$$

(Hierbei steht das Symbol t für den beobachteten Term, d für das untersuchte Dokument, N für die Anzahl der Dokumente und $d(t)$ für die Anzahl der Doumente, die t enthalten).

In realen Systemen können Term-Dokument-Matrizen Tausende von Dokumenten und Millionen von Wörtern umfassen. Um diese Mengen auch praktisch handhaben zu können, wurde eine Vielfalt von Methoden entwickelt, um die Dimensionalität der Matrizen zu reduzieren. Eine Untersuchung dieser Methoden soll zwar nicht Gegenstand der vorliegenden Arbeit sein, die praktische Notwendigkeit von dimensionsreduzierenden Verfahren wird aber in weiteren Kapiteln noch thematisiert werden.

Um die „Ähnlichkeit“ von Vektoren zu ermitteln, werden verschiedene Algorithmen eingesetzt, beispielsweise setzen auch Garcia-Vega et al. die oben erläuterte Kosinusfunktion zum Vergleich von Wortvektoren ein (vgl. [García-Vega et al., 2004]). Vazques et al. nutzen stattdessen verschiedene Maximum Entropy und

Conceptual Density Algorithmen (vgl. [Vázquez et al., 2004]), weiterführende Beispiele der Verwendung von kontextbezogenen Vektoren sind Inhalt des folgenden Abschnittes.

Merkmale im realen Einsatz Häufig werden manuell annotierte Korpora benutzt, um spezifische vektorbasierte Sprachmodelle für WSD-Aufgaben zu erstellen, allerdings werden - aufgrund des Mangels an großvolumigen, geeigneten Korpora - bisweilen auch sehr große nicht-annotierte Datenmengen verwendet, z.B. solche, die aus dem World Wide Web (WWW) extrahiert werden.

Verschiedene Arten von Merkmalen können - je nach eingesetztem Verfahren und Art des Korpus - hieraus bestimmt werden:

Lokale Merkmale Dies sind Merkmale, die lokale Abhängigkeiten des das Zielwort umgebenden Textes nutzen (z.B. Adjazenzen, Textfenster, etc.). Wertebereiche lokaler Features sind in der Regel strukturelle linguistische Informationen über Wortarten, Wortformen, etc. (vgl. auch [Yarowsky, 1993]). Diese Merkmale sind Gegenstand von Abschnitt 2.2.1.1 auf Seite 26.

Globale Merkmale Derartige Merkmale bestehen oft aus vektorbasierten Wort- und Termdaten (auch als „Bag-Of-Words“ - BOW bezeichnet) aus größeren Textfenstern (ca. 50 bis 100 Wörter vor und nach dem Zielwort). Häufig werden diese Vektoren aus Wort-Kookkurrenzinformationen erstellt (vgl. auch [Leacock et al., 1998]). Wort- und Termvektoren sind eine wesentliche Grundlage der meisten datengesteuerten Systeme. Sie liefern einfache Modelle für die statistische Weiterverarbeitung durch übergeordnete Methoden (z.B. durch Methoden des Machine Learnings): es gibt zum Beispiel nur eine Bedeutungsvariante für „match“, die sinnvoll im Zusammenhang „football match“ steht. Diese Art der Repräsentation semantischer Zusammenhänge ist die am weitesten verbreitete (vgl. [García-Vega et al., 2004, Agirre & Martínez, 2004, Crestan, 2004, Cabezas et al., 2004, Lee et al., 2004, Ciaramita & Johnson, 2004, Artiles et al., 2004, Decadt et al., 2004, Vázquez et al., 2004, Escudero et al., 2004]).

Eine ebenfalls oft eingesetzte Erweiterung einfacher Wortvektoren sind die so genannten „*n-Gramme*“ (engl. *n-Grams*). Dabei werden Worte als „Unigramme“ bezeichnet, Paare von Wörtern als Bigramme, Folgen von 3 Wörtern als Trigramme, usw.. Der zugrunde liegende Gedanke ist hierbei, daß Kombinationen von Wörtern in der Regel eine weitaus größere Aussagekraft besitzen, als einzelne Wörter. Sie können konkurrierend zu anderen Informationen zur Disambiguierung eingesetzt werden (vgl. [Agirre & Martínez, 2004, Mohammad & Pedersen, 2004, Decadt et al., 2004, Pedersen, 2004, Lee et al., 2004])

Korpusübergreifende Merkmale Beispielsweise werden sogenannte „Bilingual Tags“ von Cabezas et al. zur Auflösung der Mehrdeutigkeiten verwendet

(vgl. [Cabezas et al., 2004]): mithilfe von Markierungen gleichartig verwendeter Wörter in Texten unterschiedlicher Sprachen werden Rückschlüsse auf mögliche intendierte Bedeutungsvarianten gezogen. Es handelt sich hierbei um eine nicht-überwachte Methode, die allerdings stark von der Verfügbarkeit geeigneter Korpora abhängig ist.

Die vektorbasierte Textrepräsentation wird häufig als Gegenstück zur traditionellen sprachwissenschaftlichen Darstellungs- und Verarbeitungsweise beschrieben: statt einer „tiefen“ linguistischen Analyse werden hier häufig „flache“ statistische Vektorraum-Verfahren eingesetzt, wobei sich die eher vagen Terme „tief“ und „flach“ auf die Stufung der eingesetzten Verarbeitungsschritte beziehen können oder auch auf die Schachtelungstiefe der entstehenden Datenstrukturen.

Der Bezug zum textuellen Kontext ist für viele WSD-Verfahren besonders wichtig. Um diesen Punkt etwas zu vertiefen, soll an dieser Stelle exemplarisch auf einige unterschiedliche Ansätze in der Literatur eingegangen werden, die über die „naive“ Nutzung im Sinne der o.a. Term-Dokument-Matrizen hinausgehen (vgl. hierzu die Beispiele im Kapitelanfang auf Seite 41, insbesondere [García-Vega et al., 2004, Agirre & Martínez, 2004, Crestan, 2004, Cabezas et al., 2004, Lee et al., 2004, Ciaramita & Johnson, 2004, Artiles et al., 2004, Decadt et al., 2004, Vázquez et al., 2004, Escudero et al., 2004]).

Purundare und Pedersen untersuchten beispielsweise 2004 in [Purandare & Pedersen, 2004] die Anwendung der von ihnen entwickelten „SenseCluster“ Software: Gewichtete Wortvektoren werden in ihrem jeweiligen textuellen Kontext summiert, wobei die Mittelwerte der Summenvektoren dann die so genannten „Kontextvektoren“ bilden, die von den Autoren auch als „Vektoren zweiter Ordnung“ (engl. 2nd order vectors) bezeichnet werden. Diese Kontextvektoren werden dann für Zwecke des Word Sense Discrimination eingesetzt. Dasselbe Prinzip verwendet Pedersen auch 2004 im Rahmen des SENSEVAL-3 Workshops [Pedersen, 2004] in den verschiedenen Varianten seines „Duluth“-Systems, sowie innerhalb seiner Untersuchungen (mit Kulkarni) zum kontextbasierten Clustering (vgl. [Kulkarni & Pedersen, 2005]). In einem weiteren Artikel wird dasselbe Verfahren zum Zwecke der „Name Discrimination“, also der eindeutigen Zuordnung von Namensbezügen in verschiedenen Zielsprachen - eine Unteranwendung von WSD, verwendet (vgl. [Pedersen et al., 2006]). Beim Prinzip der Kontextvektoren handelt es sich demnach um ein Verfahren, welches relativ vielseitig im WSD-Bereich eingesetzt werden kann, wenn vordefinierte Bedeutungsvarianten nicht verfügbar sind und dennoch Bedeutungsunterschiede zwischen Wörtern ausgearbeitet werden sollen.

Oh und Choi stellen ebenfalls einen Kontextvektor-basierten Ansatz vor (vgl. [Oh & Choi, 2002]), allerdings ist der Begriff hier technisch anders belegt: Hier werden Kontextvektoren aus den das Zielwort umgebenden Wörtern gebildet - dem lokalen textuellen Kontext - und mithilfe eines Verfahrens zur Bestimmung der lokalen Dichte („Conceptual Density“, vgl. [Agirre & Rigau, 1996] hierzu) gewichtet. Jede Bedeutungsvariante aus WordNet wird ebenfalls durch einen solchen

Kontextvektor repräsentiert, der aus dem sog. Schwerpunktvektor (engl. Centroid) von Kontextvektorexemplaren der jeweiligen Bedeutungsvariante besteht.

Eine andere Art der Erstellung von Vektoren wurde von Hassel 2005 mit der Methode „Random Indexing“ bzw. „Random Labeling“ vorgestellt (vgl. [Hassel, 2005]): Der Autor beschreibt einen Ansatz zur Textklassifikation (mit WSD Anwendung), der auf der Repräsentation von Termen durch randomisierte numerische Beschriftungen beruht. Jedem Wort wird hier zunächst ein zufällig erzeugter binärer Vektor zugeordnet. In einem Folgeschritt wird dann für jede Wortokkurrenz im Text jeweils eine Signatur erzeugt, die aus der Addition der Wort-Vektoren gebildet wird. Die Feststellung der korrekten Bedeutungsvariante in WSD-Experimenten erfolgt subsequent per Berechnung der Kosinus-Entfernung zwischen Wortokkurrenz- und Bedeutungsvarianten-Vektor. Die Konstruktion und Nutzung der Kontextvektoren basiert auf dem Verfahren der randomisierten Indizierung (vgl. [Sahlgren, 2005, Kanerva et al., 2000]). In den von Hassel durchgeführten Experimenten über drei schwedische Korpora erreicht das randomisierte Verfahren eine 80%-ige Accuracy - im Vergleich zu 56% eines Verfahrens auf Basis eines Naive Bayes Classifiers. Die Verwendbarkeit der Methode hängt von der Verfügbarkeit großer (unbearbeiteter) Korpora ab, die aber im Allgemeinen gegeben ist. Allerdings wird die Performanz nach Angabe des Autors stark von bisher unzureichend erforschten Parametern beeinflußt, z.B. des „seed“-Parameters für die Initialisierung der Zufallskomponenten.

Korpusbasierte oder Vektorraumverfahren stellen insgesamt also eine bedeutende Alternative zu traditionellen bzw. linguistisch orientierten Ansätzen dar.

2.2.2.2 Einsatz externer Wissensquellen

Wissensbasierte Systeme (engl. *Knowledge based Systems*) nutzen Informationen aus externen Quellen, z.B. aus maschinenlesbaren Lexika, Thesauri und Wörterbüchern um (sprachlich und ontologisch orientiertes) Hintergrundwissen zu sammeln. Derartige Ressourcen werden oft im Englischen als *machine readable dictionaries*, kurz *MRD* bezeichnet (vgl. [Ide & Veronis, 1998]).

Zu den wichtigsten wissensbasierten WSD-Verfahren gehört der bereits in Abschnitt 2.2.1.1 auf Seite 26 erwähnte Algorithmus von Lesk, der über die Bedeutungsdefinitionen eines Wörterbuches Wörter in ihrem Kontext zu disambiguieren sucht (vgl. [Lesk, 1986]). Problematisch ist hierbei allerdings die vollständige Disambiguierung eines Textes aufgrund der hiermit verbundenen kombinatorischen Komplexität (vgl. [Pedersen & Mihalcea, 2005]), daher werden i.d.R. Abwandlungen des Algorithmus genutzt, die die Informationsvielfalt begrenzen. Ebenfalls wichtig ist die Gruppe von Algorithmen, die eine „semantische Ähnlichkeit“ z.B. anhand von Pfadlängen zwischen Konzepten innerhalb semantischer Netzwerke o.ä. messen (vgl. [Agirre & Rigau, 1996]).

In der WSD-Literatur oft genannte externe Wissensquellen lassen sich nach Agirre und Martinez (vgl. [Agirre & Martinez, 2001a]) wie folgt kategorisieren, :

Machine Readable Dictionaries (abgek. MRDs). MRDs werden durch Bruce et al. 1992 in Bezug auf WSD untersucht (vgl. [Bruce et al., 1992] in [Agirre & Martinez, 2001a]). Rigau et al. untersuchten 1997 ebenfalls eine Anzahl MRD-bezogener nicht-überwachter Algorithmen für WSD (vgl. [Rigau et al., 1997]). Die bereits vorher an verschiedenen Stellen genannten Lesk-ähnlichen Verfahren zählen ebenfalls zu den MRD-basierten Methoden.

Ontologien. die meisten Systeme benutzen hier WordNet, allerdings werden auch andere, proprietäre Ontologien für WSD Zwecke genutzt, z.B. wurden im Mikrokosmos Projekt semi-automatisch erstellte Ontologien mit Erfolg eingesetzt (vgl. [Mahesh & Nirenburg, 1995b]). Ob es sich bei WordNet tatsächlich um eine „Ontologie“ handelt, ist allerdings umstritten – nach der Definition von Gruber (Eine Ontologie ist „eine explizite Spezifikation einer Konzeptualisierung“, also eine Spezifikation eines „repräsentationalen Vokabulars einer gemeinsamen Diskursdomäne – Definitionen von Klassen, Relationen, Funktionen und weiterer Objekte“[11] , vgl. [Gruber, 1993]) würde WordNet nicht als Ontologie bezeichnet werden sondern eher als „lexikalische Datenbank“.

Weiterhin sollen einige relevante wissensbasierte Hybridansätze erläutert werden, um die Eigenschaften wissensbasierter Methoden zu verdeutlichen. Hierunter werden Kombinationen von Verfahren mit verschiedenen externen Wissensquellen, auch von kontextuellem Wissen, verstanden. Wichtig sind vor allem die folgenden Kombinationen:

Kombinationen aus MRDs und Ontologien. In solchen Fällen werden zumeist MRDs um semantische Informationen ergänzt.

Eines der ersten Systeme dieser Art ist das bereits 1992 von Yarowsky beschriebene: hier werden statistische Modelle für WSD Zwecke erstellt, in welchen jedes Wort des Zieltextes über ein Bayes' sches Modell mit entsprechenden Kategorien aus Roget's Thesaurus (der in diesem Fall die Rolle einer Ontologie übernimmt) annotiert wird (vgl. [Yarowsky, 1992a]). Diese Zuordnungen bestimmen die jeweilige Bedeutungsvariante, da den Begriffen aus Roget's Thesaurus ebenfalls verschiedene Kategorien zugeordnet sind. In diesem Ansatz übernimmt ein Bayes' sches Verfahren die Zuordnung von Wörtern zu Kategorien.

Mihalcea und Moldovan stellten 1998 ein anderes WSD-Verfahren vor, das auf dem Prinzip der „semantischen Dichte“ basierte und Daten aus WordNet und dem WWW kombinierte, die über Suchmaschinen extrahiert wurden. (vgl. [Mihalcea

[11] Übersetzung durch den Autor dieser Arbeit

& Moldovan, 1998]). Die semantische Dichte wurde über ein Lesk-ähnliches Verfahren bestimmt; wobei aus den Resultaten der WWW-Suchanfragen geeignete Trainingskorpora erstellt wurden.

McCarthy et al. nutzten 2004 automatisch erstellte Thesauri und WordNet, um aus Wort-Häufigkeitswerten die allgemein am häufigsten verwendete Bedeutungsvariante (engl. predominant sense) festzustellen. Sie stellen fest, daß die Nutzung der häufigsten Bedeutungsvariante im SENSEVAL-Kontext als alleinige (nicht-überwachte) Heuristik für WSD eine Accuracy von ca. 40% erbringt. Daher ist diese Heuristik für WSD-Probleme allgemein sehr wichtig (vgl. [McCarthy et al., 2004b], [McCarthy et al., 2004a]). Das McCarthy-Verfahren wurde in einem Folgeansatz genutzt, um thematische Modelle für WSD zu erstellen (über eine Partitionierung der in den Dokumenten identifizierbaren Themen, vgl. [Boyd-Graber & Blei, 2007])

Kombinationen aus MRDs und Korpora. Es wird durch verschiedene Verfahren die hierarchische Organisation einiger MRDs genutzt (z.B. Roget's Thesaurus), um Mengen verwandter Wörter zu produzieren und zu verwenden (vgl. auch [Yarowsky, 1992b] und [Yarowsky, 1995]).

Kombinationen aus Ontologien und Korpora. Ebenso wie die oben angeführten musterbasierten Verfahren bietet auch die Kombination von Ontologie- und Korpusmethodik eine flache und effiziente Verknüpfung von syntaktisch-lexikalischer Information mit der semantischen Ebene.

Ein früher Ansatz, der die beiden Welten der datengesteuerten und der wissensbasierten Disambiguierung von Wörtern vereinen soll, wurde von Leacock 1998 [Leacock et al., 1998] vorgestellt: ein statistisches lernendes System wird auf automatisch aus WordNet erzeugten Trainingsdaten angesetzt. Hier wird also ein Korpus aus einer Ontologie erzeugt; die Verwendung automatisch generierter Trainingsdaten wird vom Autor als "effektiv, aber nicht so effektiv wie manuell erstellte Trainingsdaten" bezeichnet. Leider werden nur experimentelle Ergebnisse von sehr geringem Volumen angegeben, so daß eine echte Vergleichbarkeit mit nicht-überwachten Systemen nicht gegeben ist.

Eine Nutzung unabhängiger Datenquellen wurde von Nica et al. vorgestellt: hier erfolgt eine Konstruktion von syntaktischen Mustern, deren Definition aus WordNet-Daten abgeleitet wird, im Zusammenhang mit korpusbasierten Okkurrenzinformationen zur Disambiguierung von Wörtern (vgl. [Nica et al., 2004]) — in diesem Falle wird WordNet als Ontologie bezeichnet. Diese Muster dienen dazu, aus großen Korpora Information über passende Wörter bzw. Textstellen zu extrahieren, wobei die gewonnenen Werte als Maßgabe dafür dienen, welche Bedeutungsvarianten im Rahmen der WSD-Aufgabe ausgewählt werden. Das Verfahren erscheint dem von Fernandez in [Férnandez-Amorós, 2004] entfernt ähnlich, allerdings ergänzen Nica et al. ihre Ausführungen um die Beobachtung, daß die Ergebnisse

mustergesteuerter Ansätze ganz erheblich von der Qualität der syntaktischen Muster abhängt.

Cuadros et al. beschreiben gleich zwei Verfahren zur Erzeugung und Nutzung von thematischen Signaturen (engl. „topic signatures“, vgl. [Cuadros et al., 2005]) zu WSD-Zwecken. Die erste Methode („ExRetriever“) erstellt thematische Signaturen in Form von AND/OR-Ausdrücken als Teilmenge der Aussagenlogik, welche jeweils genau eine Bedeutungsvariante der WordNet-Datenbasis beschreiben. Dieselben Ausdrücke können als „Suchausdrücke“ verwendet werden, d.h. bei Eingabe in eine Suchmaschine, deren Datenbasis WordNet ist, wird die jeweilig designierte Bedeutungsvariante gefunden. Die zweite Methode („Infomap“) arbeitet ähnlich, verwendet aber einen auf der Methodik des „*Latent Semantic Indexing*“ (*LSI*, vgl. [Deerwester et al., 1990]; vgl. auch *LSA* als übergeordnetem Begriff) basierenden Ansatz, um AND/ANDNOT Signaturen zu erstellen. Die Evaluation ergab für Infomap einen deutlichen Performanzvorsprung (39,3% Precision/Recall).

Weitere Kombinationen. Die Gesamtkombination von MRDs, Korpusdaten und Ontologien wird nur in wenigen Projekten gewinnbringend eingesetzt, ein herausragendes Beispiel hierfür ist das Mikrokosmos Projekt: Beale, Mahesh und Nirenburg verknüpften dabei ein Lexikon mit einer situierten Ontologie, um ein System zur automatischen, wissensbasierten Übersetzung von spanischen und englischen Texten zu schaffen (vgl. [Beale et al., 1995], [Mahesh & Nirenburg, 1995b], [Mahesh & Nirenburg, 1995a] und [Mahesh, 1996]). Hierbei werden lexikalische und semantische Komponenten eingesetzt, um verschiedene Abstraktionsebenen miteinander zu verknüpfen – wobei der WSD Problematik eine zentrale Rolle zukommt. Die Betrachtung von Textbedeutung basiert auf einer logischen Form, die durch eine – manuell erstellte und graphenbasierte – Ontologie repräsentiert wird. Daher besteht der Anspruch des Ansatzes, in gewisser Weise eine Form von „Textbedeutung“ über interne Datenstrukturen zu repräsentieren – nicht zuletzt basiert der Mikrokosmos-Ansatz auf der Theorie der Mikrotheorien, die die Welt in kleine, beherrschbare Subsysteme abzubilden versucht. Die Objekte der Mikrokosmos-Ontologie werden TMR – für „Text Meaning Representation“ – genannt, ein Beispiel für einen solchen TMR ist in Abbildung 2.1 auf der nächsten Seite dargestellt. TMRs werden „händisch“ als *Rahmen* (engl. *Frame*) definiert und über einen Editor in die Datenbank eingegeben. Eine solche Definition enthält verschiedene Attribute und deren Werte, z.B. erhält in der Abbildung 2.1 das Konzept CONVERSATION-COMMUNICATE-2 das Attribut „Ziel“ (Destination) mit der Angabe des Kommunikationspartners (CONVERSATION-COMMUNICATE-1.Agent) als Wertebelegung. Eine Disambiguierung selbst findet über einen Algorithmus statt, der für mehrere Konzepte den jeweils kürzesten Pfad innerhalb der Ontologie findet, sowie über die Gewichtung der verschiedenen Eigenschaften dieser Konzepte und Auswahl durch selektive Restriktionen.

Listing 2.1: *Beispiel eines TMRs als komplexes Konzept in der Mikrokosmos Ontologie*

```
(make-frame CONVERSATION-COMMUNICATE-2
  (Definition
    (Value "a subevent of the complex event CONVERSATION.
       "))
    (Time-Stamp (Value "Not yet created!"))
    (Subevent-Of (Value CONVERSATION))
    (Onto-Instance-Of (Value COMMUNICATIVE-EVENT))
    (Agent (Value CONVERSATION-COMMUNICATE-1.Destination)
       )
    (Destination (Value CONVERSATION-COMMUNICATE-1.Agent)
       )
    (Aspect
      (Value (duration prolonged)
      (iteration multiple)
    )
  )
)
```

2.2.2.3 Nutzung des WWW

Obwohl das World Wide Web neben den bisher angeführten Korpora BNC, etc. ebenfalls – streng genommen – als Korpus betrachtet werden kann (vgl. [Kilgarriff & Grefenstette, 2003]), soll aufgrund der hohen Relevanz zum TSR-Ansatz – bei diesem handelt es sich ebenfalls um ein (indirekt) webbasiertes Verfahren – den webgestützten WSD-Methoden an dieser Stelle besondere Aufmerksamkeit zukommen (Eine Vertiefung in die Thematik erfolgt in Abschnitt 3.4 auf Seite 74).

Direkte Nutzung des WWW. Die Verwendung von webbasierten Verfahren ist aufgrund der immensen Größe des Korpus interessant, zumal nach Agirre und Martinez die Anzahl der Trainingsexemplare eines überwachten oder semiüberwachten Experimentes erheblichen Einfluß auf das Ergebnis haben kann (vgl. [Agirre & Martinez, 2004]). Dabei wird der menschliche Vorbereitungsaufwand im Rahmen überwachter Systeme minimiert, indem das World Wide Web zunehmend als Datenquelle genutzt wird („Web As Corpus“, vgl. [Kilgarriff & Grefenstette, 2003] für eine Einführung zu diesem Thema).

Wegen der hohen Varianz der unvorbereiteten Textdaten - beispielsweise hervorgerufen durch unübliche Formatierungen, fragmenthafte Ausdrucksweise, Verwendung von Slangausdrücken, etc. - existieren dennoch nur wenige dokumentierte Experimente im Bereich WSD. Weil der TSR-Ansatz ebenfalls auf webbasierten (allerdings manuell bearbeiteten und kategorisierten) Daten basiert, besteht häufig eine gute Vergleichbarkeit zu den unten genannten Verfahren.

Der Zugriff auf das Web erfolgt in der Regel über den Einsatz von Suchmaschinen. Ein einfaches WSD-Verfahren wird unter diesem Gesichtspunkt von Klapaftis und Manadhar vorgeschlagen: hierbei werden Sätze aus den Zieltexten durch Suchergebnisse der Suchmaschine Google ergänzt (vgl. [Klapaftis & Manandhar, 2005]) – Suchausdrücke sind dabei sowohl die Sätze selbst, als auch die Informationen aus WordNet-Synsets, die eindeutig mit Wörtern aus den Sätzen in Verbindung gebracht werden können (monosemische Wörter, d.h. Wörter mit nur einer Bedeutungsvariante), sowie Wörter aus Synsets, die in einer Relation zu den o.a. Synsets stehen (Hyponyme, Synonyme, etc.). Aus den Rückgabetexten der Suchmaschine wird ein Kontextmodell erstellt, welches dann zur Disambiguierung eingesetzt wird. Die Autoren beschreiben ein Experiment zur Evaluation anhand der ersten 10 SemCor 2.0 Dateien und berichten eine Accuracy von 58,9% (SemCor ist ein Standard-Korpus von Texten, welche mit WordNet-Bedeutungsvarianten annotiert wurden, vgl. [Mihalcea, 2002]).

Mihalcea und Moldovan erzeugen in ähnlicher Weise beliebig große Korpora, indem sie lexikalische Phrasen, welche die Bedeutung eines Wortes beschreiben, als Suchbegriff in die Suchmaschine Altavista eingeben und die Rückgabe der Suchmaschine dann als Beispielexemplare der Verwendung dieser Bedeutung interpretieren (vgl. [Mihalcea & Moldovan, 1999]). Dieses Prinzip ist zwar nicht direkt für WSD Zwecke vorgesehen, Mihalcea und Moldovon schlagen ihr Verfahren aber für die Herstellung großer Trainings- und Testdatenmengen für WSD-Systeme vor. Tatsächlich greifen dann Mihalcea und Faruque später einen ähnlichen Ansatz auf, verwenden statt der Ergebnisse von Suchmaschinen allerdings das SemCor-Korpus (vgl. [Mihalcea & Faruque, 2004]). Die Autoren bezeichnen das Verfahren als „minimal überwacht“.

Agirre et al. beschreiben in [Agirre et al., 2000] ebenfalls ein Verfahren, Informationen aus dem WWW über Suchmaschinen einzubringen, um dadurch WordNet-Konzepte um thematisch verwandte Wörter anzureichern (Erstellung von thematischen Signaturen, engl. „topic signatures“) und außerdem Cluster von kontextrelevanten Wörtern zu bilden. Die zentrale Annahme ist hierbei, daß die Bedeutung eines Wortes aus den umliegenden Wörtern geschlossen werden kann (Kontexthypothese). In einem Beispiel verdeutlichen die Autoren den zentralen Vorgang der Definition der eigentlichen Suchanfrage:

> Wort: boy, folgende Zusammenhänge können aus WordNet extrahiert werden:

synonyms	*male child, child*
gloss	*a youthful male person*
hypernyms	*male, male person*
hyponyms	*altar boy, ball boy, bat boy, cub, lad, laddie, sonny, sonny boy, boy scout, farm boy, plowboy, ...*
coordinate	*chap, fellow, lad, gent, fella, blighter, cuss, foster*
systers	*brother, male child, boy, child, man, adult male, ...*

Die Suchanfrage für die erste Bedeutungsvariante beinhaltet die o.a. Begriffe, sowie die Negation der Begriffe alternativer Bedeutungsvarianten des Wortes. Ein Exzerpt der Suchanfrage sähe wie folgt aus:

```
('boy'
   AND ('altar boy' OR 'ball boy' OR 'male person')
   AND NOT ('man' OR 'broth of a boy' OR    # sense 2
            'son' OR OR 'mama's boy' OR     # sense 3
           OR 'black')                      # sense 4
```

Basierend auf Kollokationswerten der Wörter in der Rückgabe der Suchmaschine, entsprechend der Methodik der Term-Dokument-Matrizen, wird nun die eigentliche WSD durchgeführt, indem die Werte für die einzelnen Bedeutungsvarianten addiert werden und die Bedeutungsvariante mit dem höchsten Wert als Interpretation gewählt wird.

Indirekte Nutzung des WWW. Web Verzeichnisse wurden ebenfalls für verschiedene Zwecke der Computerlinguistik eingesetzt. Da diese Idee zentral für den TSR-Ansatz ist, wird auch die im Folgenden beschriebene Arbeit mit einem Schwerpunkt auf Information Extraction und Information Retrieval vorgestellt, die - oberflächlich betrachtet - dem TSR Verfahren sehr ähnelt:

Ein früher Ansatz, in dem die ursprüngliche Struktur eines Internetverzeichnisses direkt genutzt wird, wurde von Mladenic 1998 vorgestellt: hierbei werden aus Dokumenten der verschiedenen Hierarchieebenen des Yahoo-Verzeichnisses (vgl.

[Yahoo Inc., 2004]) einzelne Bayesche Klassifikatoren erzeugt, deren summiertes Ergebnis zu Zwecken der Dokumentklassifikation eingesetzt wurde (vgl. [Mladenic, 1998]). Eine Bewertung, die das Verfahren in einen Vergleich zu anderen Verfahren setzt, ist nicht gegeben. Obwohl es sich nicht um ein dezidiertes WSD-Verfahren handelt, kann die Methodik aber denkbar gut für einen solchen Zweck eingesetzt werden, da es sich bei WSD um ein spezifisches Klassifikationsproblem handelt. An dieser Stelle wird es erwähnt, um die prinzipielle Möglichkeit aufzuzeigen, statistische Relationen innerhalb eines Internetverzeichnisses zu verankern.

Santamaria et al. beschreiben 2003 in [Santamaria et al., 2003] einen zum TSR Verfahren auf den ersten Blick sehr ähnlichen indirekten webbasierten Ansatz. Hier wie dort, wird das Open Directory Projekt (ODP, vgl. Abschnitt 3.4.2) als Datenbasis bzw. Korpus verwendet. Die Autoren begründen dies mit der Annahme, daß Internetverzeichnisse ein wesentlich strukturierteres und zuverlässigeres Korpus darstellen, als das WWW an sich. Hierbei werden ODP Pfade als „Charakterisierungen“ von Bedeutungsvarianten innerhalb von WordNet betrachtet, die mit diesen über Wort-Kookkurrenzen verknüpft werden.

Als Beispiel wird das Wort *Circuit* angegeben, welches hier exzerptiv zur Verdeutlichung wiedergegeben wird:

Bedeutungsvariante 1:

```
ODP →l(d)
Pfad = [/business/industries/electronics and
electrical/contract manufacturers]
WordNet→l(circuit1)
SynSet = [electrical device] , Gloss=[circuit,
electrical circuit, electric circuit]
```

Hieraus werden die untenstehenden Listen erzeugt - $l(d)$ ist hierbei die Menge der Begriffe, die im ODP-Pfad auftreten und $l(circuit_1)$ repräsentiert die Menge der relevanten und lexikalisch verwandten WordNet-Begriffe, d.h.der Wörter aus SynSet und Gloss, sowie der Synonyme, direkten Hypernyme, Holonyme etc.:

$l(d)$ =[business, industries, electronics, electrical, contract, manufacturers]

$l(circuit_1)$ =[electrical circuit, electric circuit, electrical device, bridge, bridge circuit, Wheatstone bridge, bridged-T, closed circuit, loop, parallel circuit, shunt circuit, computer circuit, gate, logic gate, AND circuit, AND gate, NAND circuit, NAND gate, OR circuit, OR gate, X-OR circuit, XOR circuit, XOR gate, integrated circuit, (..) instrumentality, instrumentation, artifact, artefact, object, physical object, entity]

Eine Berechnung der Ähnlichkeit von Wörtern erfolgt über eine einfache Auswertung (durch Zählung) der Wort-Kookkurrenzen in $l(d)$ und $l(w_i)$.

Bedeutungsvariante 2:

```
ODP
/sports/motorsports/auto racing/tracks
/sports/motorsports/auto racing/formula one
/sports/equestrian/racing/tracks

WordNet
SynSet=[racing circuit],
Gloss=[racetrack,racecourse,raceway,tracks]
```

Beide Beispiele für Bedeutungsvarianten von „Circuit“ setzen also verschiedene ODP-Pfade mit Einträgen in WordNet zueinander in Beziehung, sofern gleiche Wörter in den Einträgen beider Datenbasen auftreten.

Dieses Verfahren nutzt lediglich die aus den Pfadbeschreibungen selbst gewonnenen Hinweise und nicht die durch die Pfade referenzierten Beschreibungstexte. Außerdem werden die Pfadlisten nicht den Wörtern selbst zugeordnet, sondern den entsprechenden WordNet Bedeutungsvarianten. Santamaria et al. geben eine Coverage (der Anteil der WordNet Bedeutungsvarianten, die mit ODP Pfaden annotiert werden konnten) im Bereich von 79% - 88% bei 86% Accuracy auf einer Untermenge der SENSEVAL-2 Daten (29 Nomen mit insgesamt 147 Bedeutungsvarianten) an. Damit läßt sich der überwiegende Teil der untersuchten Texte durch das Verfahren beschreiben.

Eine Methode, in der das Wissen aus Internetverzeichnissen nicht nur, wie bei Santamaria et al., genutzt wird um eine Verknüpfung zwischen Webinhalten und einer Ontologie herzustellen, sondern eine Ontologie zu erzeugen, wird von Kavalec und Svatek untersucht: In [Kavalec & Svatek, 2002] wird ein verzeichnisgestütztes Verfahren mit dem Ziel des Information Extraction vorgestellt, das aus ODP-Daten Ontologien erzeugen soll. Hierzu definieren die Autoren sinnvolle Relationen zwischen einzelnen Knoten des ODP Baumes und versuchen, diese semi-automatisch zu extrahieren. Eine relevante Erfolgsmessung findet allerdings nicht statt. Die Autoren versuchen, die Menge faktischer Relationen zwischen ODP Hierarchieebenen selbst zu kategorisieren und über syntaktische Merkmale automatisiert (und scharf abgrenzbar) bestimmbar zu machen.

Abschließend kann festgestellt werden, daß auch Daten aus dem Web oder aus Internetverzeichnissen in verschiedener Weise durch WSD-Verfahren genutzt werden, wobei dieser Ansatz bei weitem nicht so häufig auftritt wie die Nutzung anderer Korpora und Wissensquellen.

2.2.3 Modellierung von Domänenwissen

In diesem Abschnitt sollen Verfahren nach dem Einsatz von Trainingsdaten zur Modellbildung unterschieden werden. Grob gesagt, lassen sich viele Verfahren im WSD-Bereich nach dem Ursprung des Domänenwissens unterscheiden:

1. **Überwachte Systeme** (engl. *Supervised Disambiguation):* Überwachte Systeme nutzen Informationen aus manuell beschrifteten Textsammlungen, in welchen eine Zuordnung der Bedeutungsvariante[12] bereits vorgenommen wurde (d.h. die bereits disambiguiert wurden).

   In diesem Fall wird das Domänenwissen in der Regel über statistische Verfahren aus den Texten selbst gewonnen (z.B. über Kookkurrenzinformationen). Methoden, die überwacht arbeiten, sind meist robust gegenüber einzelnen Formulierungen und Wortokkurrenzen aber häufig nicht robust gegenüber Änderungen der Wissensdomäne.

   Viele Systeme verfolgen hierbei ein korpusbasiertes Vektorverfahren, den sog. *bag-of-words* Ansatz, in welchem Wörter aus dem Umfeld des zu untersuchenden Wortes extrahiert und (wie in einem „Sack“, d.h. ohne innere Struktur) gesammelt werden. Eine Disambiguierung erfolgt in diesen Fällen über den Vergleich von Instanzen aus dem gespeicherten Wortumfeld und aus dem Umfeld der zu prüfenden Wortinstanz, meist über generische Verfahren aus dem Bereich des Maschinellen Lernens, z.B. Support Vector Machines, Decision Lists, Decision Trees oder Naive Bayes Classifiers (vgl. [Witten & Frank, 2000, Joachims, 1998, Yarowsky, 1994, Pedersen & Mihalcea, 2005]).

2. **Nichtüberwachte Systeme** (engl. Unsupervised Disambiguation): Nicht überwachte Systeme erstellen Gruppen von Bedeutungen aus unvorbereiteten Textsammlungen. Unsupervised Disambiguation stellt demnach eher einen Ansatz zur Word Sense Discrimination, wie oben dargestellt, dar. Ein Ansatz hierzu wurde 1995 von Yarowsky vorgestellt (vgl. [Yarowsky, 1995]) ein weiterer 1998 von Schütze (vgl. [Schütze, 1998]). Im Unterschied zu den wissensbasierten Systemen werden von einigen Autoren (z.B. Patwardhan et al., vgl. [Patwardhan et al., 2003]) hierunter ausschließlich Systeme ohne zugrunde liegende Datenbasen verstanden.

   Nichtüberwachte Ansätze beinhalten oft entweder statisches Domänenwissen in Form von Heuristiken – und sind in dieser Weise wenig robust gegenüber Änderungen der Domäne, oder sie sind prinzipiell domänenunabhängig konstruiert, wie beispielsweise das SenseLearner-System von Mihalcea und Faruque (vgl. [Mihalcea & Faruque, 2004]).

   Mihalcea und Pedersen definieren das nichtüberwachte WSD etwas strikter: hier werden überhaupt keine externen Daten zugelassen, auch nicht a

[12] In der englischsprachigen Literatur häufig als „Sense“ bezeichnet.

priori integriertes Wissen aus externen Datenbanken, dem WWW o.ä. WSD stellt sich dann als Clusteringproblem dar, d.h. als Methodik, Bedeutungsvarianten sinnvoll zu gruppieren, ohne daß die Gruppierungen bestehenden Kategorien oder Bedeutungsdefinitionen entsprechen müssen (vgl. [Pedersen & Mihalcea, 2005]). Dies ist allerdings nicht die in der hier vorliegenden Arbeit vertretene Sichtweise.

Für die Darstellung und Evaluation des TSR-Ansatzes werden die überwachten Systeme aufgrund ihrer Abhängigkeit von zusätzlichen Faktoren (Güte und Volumen der Trainingsdaten, Angemessenheit des Trainingsverfahrens, etc.) nur von randständigem Interesse sein, dennoch handelt es sich um einen wichtigen Bereich der NLP-Forschung, dessen Position gegenüber den nichtüberwachten Ansätzen aufgezeigt werden mußte.

2.2.4 Hybride Methoden

Unter Hybrid- oder „Ensemblemethoden“ werden im Zusammenhang dieser Arbeit komplexe WSD-orientierte Verfahren verstanden, welche mehrere Einzelansätze auf geeignete Weise kombinieren, um bessere Resultate zu erhalten. In der Regel liefern die untergeordneten Algorithmen eine Empfehlung bezüglich der angestrebten Disambiguierung, manchmal wird zusätzlich ein „Gütefaktor“ mitgeliefert, der die Vertrauenswürdigkeit des Ergebnisses beziffern soll. Die Eingabe dieser untergeordneten Methoden hängt dabei vom eingesetzten Algorithmus ab, häufig handelt es sich um den textuellen Kontext, welcher durch das einzelne Verfahren selbst aufbereitet wird (z.B. durch Vektorisierung oder flache lexikalische Analyse). Abbildung 2.3 stellt einen solchen Hybridansatz schematisch dar.

Alle aktuellen Systeme verwenden eine hybride Methodik, in welcher oft unterschiedliche Prinzipien zur Anwendung kommen – meist erfolgt eine Kombination aus wissens- und korpusbasierten Algorithmen. Derartige Ensemble-Systeme zeigen in der Regel eine bessere Leistung als „Single-Idea“ Systeme, da sie sich häufig nutzen lassen, um die Vorteile von Einzelverfahren zu kombinieren. Die Leistungsfähigkeit verschiedener Ensemblemethoden wurde in diesem Zusammenhang von Dietterich (vgl. [Dietterich, 1997]) untersucht:

1. **Direct Voting.** Jede Komponente hat eine "Stimme"; die Bedeutungsvariante mit den meisten Stimmen wird "gewählt": üblicherweise werden die Ergebnisse untergeordneter Verfahren gewichtet und miteinander verglichen; das Ergebnis mit dem besten Resultat wird dann nach außen weitergereicht. Diese Methode ist in Abbildung 2.4 auf Seite 55 dargestellt. Ein Beispiel hierfür findet sich bei Florian, welcher ein Votingverfahren über die Ergebnisse von 5 prädefinierten syntaktischen Mustern beschreibt (vgl. [Florian & Yarowsky, 2002]). Ein weiteres solches System wurde von Agirre 2004 eingesetzt, hierbei wurden die Ergebnisse der untergeordneten Systeme (Ent-

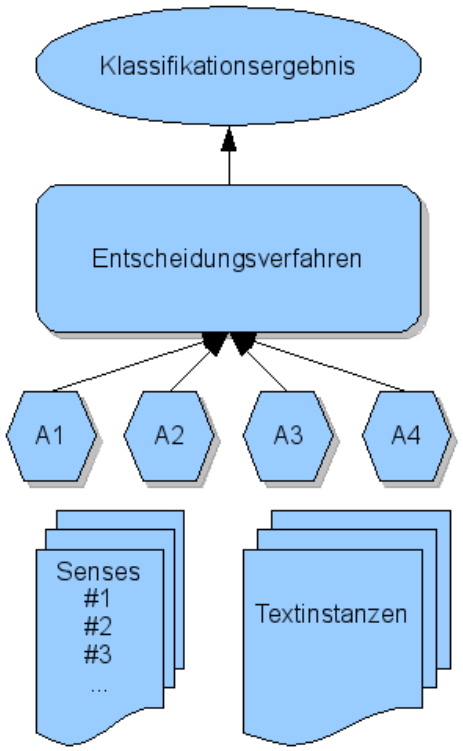

Abbildung 2.3: Schematischer Aufbau eines Hybridverfahrens

scheidungslisten, Naive Bayes, SVM und ein einfaches Vektorraumverfahren) verglichen und das beste Ergebnis gewertet (vgl. [Agirre & Martínez, 2004]).

2. **Probability Mixture.** Die Summe von Wahrscheinlichkeiten, die aus der Bewertung der Einzelkomponenten abgeleitet wird, bestimmt die Bedeutungsvariante, wie in Abbildung 2.5 auf Seite 56 gezeigt wird.

3. **Rank-based Combination.** Das Ranking jeder Komponente wird summiert; die Bedeutungsvariante mit dem geringsten Wert (d.h. mit der höchsten Platzierung, wobei Platz 1 den besten Wert markiert) gewinnt - dieser Sachverhalt wird in Abbildung 2.6 auf Seite 57 schematisch aufgezeigt.

Die Verfahren unterscheiden sich nach Dietterich offenbar nur gering in ihrer Effektivität, wobei aber nach Preiss (vgl. [Preiss, 2004]) die Ensemble-Methoden allgemein erfolgreicher sind als jede Einzelkomponente für sich betrachtet. Aktuelle Verfahren berücksichtigen zudem zunehmend neuere Methoden der Statistik, u.a. die Methode der „Latenten Dirichlettschen Allokation“ (abgek. *LDA*; ein Verfahren zur Erstellung probabilistischer Modelle über Sammlungen diskreter Daten).

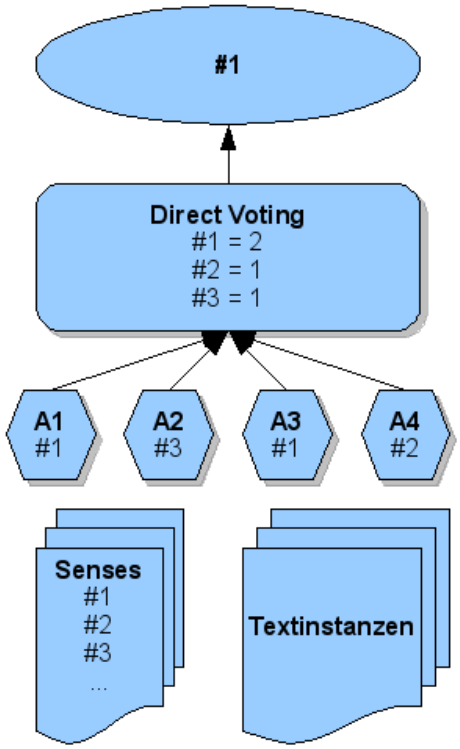

Abbildung 2.4: Hybridmethode Direct Voting

Hierbei werden die Daten der Komponenten durch den LDA-Algorithmus gesammelt und z.B. durch einen Naive-Bayes Klassifizierer bewertet werden (vgl. [Cai et al., 2007]).

Ein „Pseudo-hybrider“ Ansatz, in welchem verschiedene Algorithmen integriert sind, die jedoch je nach Wortart des zu disambiguierenden Wortes angesprochen werden, ist das erweiterte JIGSAW-Verfahren, das von Basile et al. vorgestellt wurde (vgl. [Basile et al., 2007]). Diese Art der Hybridisierung ist jedoch im Kontext dieser Arbeit nicht gemeint.

Hybride Verfahren liefern in der Regel die besten Ergebnisse (vgl. hierzu [Stevenson & Wilks, 2001] und [Brody et al., 2006]), allerdings sind die Beiträge der einzelnen untergeordneten Methoden opak - ein Hybridsystem eignet sich also nicht zu Analyse der Leistungsfähigkeit eines singulären WSD-Ansatzes.

2.3 Evaluation von WSD-Verfahren

Computerbasierte Textverarbeitungsverfahren, besonders im Bereich der WSD-Forschung, können in vielen Fällen unter Zuhilfenahme einiger Standardberech-

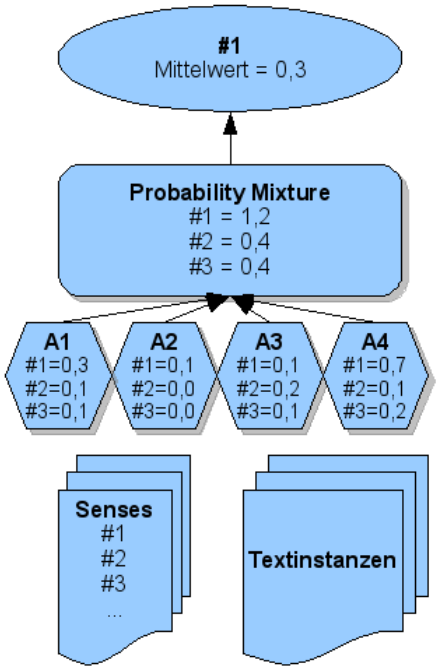

Abbildung 2.5: Hybridmethode Probability Mixture

nungen und –datenmengen hinsichtlich der Qualität ihrer Leistung bewertet und miteinander verglichen werden.

Einerseits sollen daher in diesem Abschnitt die wesentlichen, im WSD-Bereich gebräuchlichen Metriken erläutert werden und andererseits werden die SENSEVAL-Workshops vorgestellt werden, eine Reihe von Veranstaltungen, die das Ziel haben, Bestrebungen im WSD-Bereich zu konsolidieren und hierfür eine experimentelle Standardumgebung zu liefern.

2.3.1 Übliche Metriken

Die in diesem Abschnitt aufgeführten Metriken der Performanzmessung für WSD-Verfahren entstammen ursprünglich dem Bereich *Information Retrieval*, in welchem die Suche von Informationen in Dokumentsammlungen thematisiert wird. Sie lassen sich aber auch sehr gut auf andere Bereiche der Textverarbeitung – wie dem WSD-Bereich – übertragen.

Man geht hierbei grundlegend davon aus, das infrage stehende Textverarbeitungsproblem auf ein Klassifizierungsproblem zurückführen zu können. Eine einfache derartige Situation kann wie folgt beschrieben werden: eine Menge von Dokumenten soll auf eine geeignete Art und Weise so partitioniert werden, daß diese Partitionierung einer vorgegebenen Spezifikation genügt. Beispielsweise kann WSD

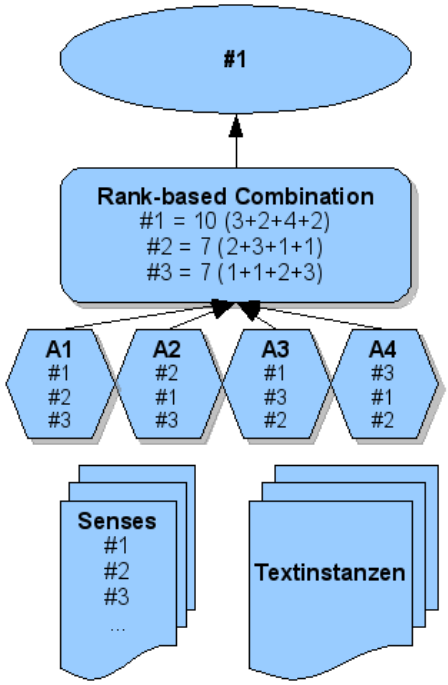

Abbildung 2.6: Hybridmethode Rank-based Combination

dergestalt als Klassifizierungsproblem interpretiert werden, daß für jede Bedeutungsvariante eine Partition definiert wird und subsequent alle Instanzen des nach WSD aufzulösenden Textkorpus einer dieser Partitionen zugeordnet werden. Es gibt viele Abwandlungen dieses Grundproblems, z.B. Probleme, in denen mehrere Lösungen ausgegeben werden können, also keine Partitionierung der Basismenge erzeugt wird, sondern eine Anzahl von Untermengen oder Clustern. Für die Zwecke dieser Arbeit soll aber die eben gegebene Definition genügen.

Wenn man die Performanz derartiger Verfahren untersucht, sind vier Quantitäten dabei von besonderer Bedeutung für jede Partition (vgl. [Aas & Eikvil, 1999], [van Rijsbergen, 1979], [Sebastiani, 2002] u.a.):

Symbol	**Beschreibung**
a	Die Anzahl der *korrekt zugeordneten* Dokumente
b	Die Anzahl der *inkorrekt zugeordneten* Dokumente
c	Die Anzahl der *inkorrekt nicht zugeordneten* Dokumente
d	Die Anzahl der *korrekt nicht zugeordneten* Dokumente

Unter Zuhilfenahme dieser Quantitäten lassen sich die folgenden gängigen Meßzahlen definieren:

DEFINITION 1: Precision – Genauigkeit der Klassifizierung (Anzahl der richtigen Zuordnungen im Vergleich zur Menge aller Zuordnungen)

$$precision = \frac{a}{(a+b)}$$

DEFINITION 2: Recall – Vollständigkeit der Klassifizierung (Anzahl der getroffenen korrekten Zuordnungen im Vergleich zur insgesamt möglichen Anzahl der Zuordnungen)

$$recall = \frac{a}{(a+c)}$$

DEFINITION 3: Fallout – Vermeidung von Fehlklassifikationen (Anzahl der fehlerhaften Zuordnungen entsprechend der Definition von *Recall*)

$$fallout = \frac{b}{(b+d)}$$

DEFINITION 4: Accuracy – Korrektheit der Klassifizierung (Anzahl der richtigen Entscheidungen im Vergleich zur insgesamt möglichen Anzahl aller Entscheidungen - eine Entscheidung ist hierbei die Zuordnung zu einer Klasse oder die Unterlassung der Zuordnung)

$$accuracy = \frac{(a+d)}{(a+b+c+d)}$$

DEFINITION 5: Error (Anzahl der falschen Entscheidungen entsprechend der Definition von *Accuracy*)

$$error = \frac{(b+c)}{(a+b+c+d)}$$

Für das Zusammenziehen der Werte über alle Partitionen gibt es zwei Wege:

1. Das *Micro-Averaging* (*Feinmessung*) wird verwendet, wenn alle Dokumente des Textkorpus gleich gewichtet sein sollen; hierbei werden die Werte für die Quantitäten a bis d summiert und danach die Meßzahlen precision, recall, usw. berechnet.

2. Das *Macro-Averaging* (*Grobmessung*) gewichtet alle Partitionen gleich; es werden zunächst die Meßzahlen für jede Partition einzeln berechnet, und dann die Ergebnisse zusammengefaßt.

Verschiedene Vor- und Nachteile der Metriken wurden 2003 von Cohn diskutiert (vgl. [Cohn, 2003]): Nach Cohn sind *precision* und *recall* sehr verbreitete Metriken, es besteht daher eine gute Vergleichbarkeit zwischen verschiedenen Experimenten sofern die experimentelle Grundlage (Daten, Versuchsaufbau) vergleichbar

ist; der Wert *precision* besitzt allerdings nur eine echte Aussage im Zusammenhang mit dem entsprechenden Wert für *recall*, es müßte also eigentlich immer heissen: „X Prozent *precision* bei vorgegebenem Prozentsatz Y *recall*“. Cohn bewertet die Metrik *accuracy* als leicht verständlich, aber leider wird lediglich die Anzahl Fehler gemessen und daher nicht der Schweregrad einer Fehlklassifikation reflektiert (Dies trifft auch auf die *precision* Metrik zu). Er bewertet das (nicht verbreitete und daher oben nicht aufgeführte) Verfahren der „Semantische Distanz“ (vgl. [Resnik & Yarowsky, 1999]) als beste Metrik: feingranulare Unterschiede zwischen Bedeutungsvarianten werden weniger stark gewichtet als gröbere Unterschiede. Es handelt sich aber um ein Verfahren, welches derzeit allerdings noch keine Verbreitung erfahren hat, recht kompliziert ist, vergleichsweise empfindlich gegenüber den Parametern einer Gewichtungsmatrix ist und daher nicht im Zusammenhang mit dieser Arbeit eingesetzt wird.

Weitere verbreitete Maßzahlen basieren z.B. auf dem sog. F-Wert (einer Kombination von precision und recall), sollen aber im Rahmen dieser Arbeit ebenfalls nicht weiter behandelt werden.

Für die in späteren Kapiteln erfolgenden Untersuchungen sind in der Regel nur die Werte für *accuracy*, sowie *precision* und *recall* relevant. Die Bedeutung dieser Meßzahlen ergibt sich im Wesentlichen aus ihrer Verbreitung: Da der in dieser Arbeit vorgestellte Ansatz mit anderen Ansätzen verglichen wird, muß die Vergleichsbasis identisch sein.

2.3.2 Ein praktisches Bewertungsverfahren: die SENSEVAL-Workshops

Einen Meilenstein der Erforschung der Disambiguierung von Wortmehrdeutigkeiten stellt die Reihe von SENSEVAL-Workshops dar, die auf WSD–Verfahren ausgerichtet sind und eine Standard-Evaluationsumgebung für WSD–Verfahren bieten. Grundlegend für die Bewertung der TSR Verfahren im Umfeld WSD ist die Existenz des durch die betreffende wissenschaftliche Gemeinschaft anerkannten und standardisierten Evaluationsszenarios, welches durch die SENSEVAL-Workshops gegeben ist. Im Rahmen dieser Arbeit wurden daher die experimentellen Bedingungen von SENSEVAL-3 verwendet, um die Vergleichbarkeit der Daten zu gewährleisten. Aus diesem Grunde sollen - abgesehen von den in Abschnitt 2.1 vorgestellten Betrachtungen - die SENSEVAL-Systeme in der Diskussion der aktuellen Forschung auch eine besondere Beachtung erfahren.

Die Auflösung von Wortmehrdeutigkeiten ist ein zentrales Thema der Verarbeitung natürlichsprachlicher Texte. Aus diesem Grunde, und weil diesbezügliche Arbeiten in der Vergangenheit untereinander schlecht vergleichbar waren, wurde 1997 die SENSEVAL Organisation gegründet. SENSEVAL wird von einem kleinen Komitee unter dem Dach der *Special Interest Group on the Lexicon of the Association for Computational Linguistics* (Abk. ACL-SIGLEX) betreut und hat die

Schaffung von Kriterien und die Organisation von Wettbewerben und Konferenzen für die Auswertung und den Vergleich von WSD-Systemen zum Ziel. Der erste Wettbewerb wurde im Rahmen des SENSEVAL-1 Workshops 1998 durchgeführt (vgl. [Edmonds, 2002]). Aufgrund des Erfolges dieser ersten Veranstaltung folgten die SENSEVAL-2 (2001) und SENSEVAL-3 (2004) Workshops.

In der vorliegenden Arbeit steht der Vergleich zu SENSEVAL-3 als der bisher letzten Veranstaltung dieser Art im Vordergrund, da SENSEVAL-3 eine akzeptable Basis für die Untersuchung des Forschungsstandes repräsentiert. Nahezu alle Verfahren, die zur Evaluation im SENSEVAL-3 Rahmen eingereicht wurden, nutzen kombinierte Ansätze - dies ist, wie bereits in Abschnitt 2.2.4 auf Seite 53 gezeigt wurde, auch die effektivste Vorgehensweise. Es ist daher nur in wenigen Fällen möglich, die Leistung derartiger Systeme einer singulären Repräsentationsweise oder Funktionalität zuzuordnen. Aus diesem Grund können die im Weiteren vorgestellten Verfahren auch nicht anhand ihrer Performanz, sondern lediglich über ihre unterschiedlichen Detailansätze verglichen werden. Die Ergebnisse der einzelnen Systeme werden durch Mihalcea et al. in [Mihalcea et al., 2004] vorgestellt.

Fast alle SENSEVAL-3 Systeme folgen dem Hybridprinzip, verschiedene lexikalische, syntaktische und semantische Informationen mit entsprechenden Methoden zu verarbeiten und die verschiedenen Ergebnisse dann mithilfe eines übergeordneten Verfahrens zusammenzuführen. Eine Vielzahl von untergeordneten Informationen, die aus den Texten gewonnen werden, werden benutzt, um die oben angeführten Methoden mit Daten zu versorgen.

SENSEVAL-3 bietet verschiedene Bereiche für unterschiedliche Anwendungstypen und -sprachen. Der hier zugrunde gelegte Teil wird als „English Lexical Sample“ bezeichnet und durch Mihalcea et al. beschrieben (vgl. [Mihalcea et al., 2004]). Die Beschreibung enthält eine Definition der Aufgabe und der zu verwendenden Ressourcen, sowie Ergebnisse der teilnehmenden Systeme:

Ziel der Aufgabe „English Lexical Sample“ war es, die Mehrdeutigkeit einzelner englischer Zielwörter in vorgegebenen Texten aufzulösen, indem diese Wörter mit bestimmten Bedeutungsvarianten verknüpft wurden. Die genutzten Quelltexte wurden aus der Textsammlung des British National Corpus (BNC) extrahiert und von verschiedenen Editoren unter Zuhilfenahme des „Open Mind Word Expert System“ bearbeitet, einem webbasierten System zur Bearbeitung textueller Korpora (vgl. [Chklovski & Mihalcea, 2002] in [Mihalcea et al., 2004]). Die Aufgabe wurde von 27 Teams aus verschiedenen Erdteilen mit insgesamt 47 Systemen bearbeitet.

Als Quelle der verwendeten Bedeutungsvarianten wurde für Substantive und Adjektive die WordNet Datenbank der Version 1.7.1 benutzt (vgl. [Miller et al., 2005]). Die Quelle der Verb-Bedeutungsvarianten war die Wordsmyth-Datenbasis (vgl. [Kennedy & Broquist, 2004]), da die Verbdefinitionen in WordNet als ungenügend empfunden wurden. Die Anzahl der einzelnen Wortformen und die durchschnittliche Anzahl der entsprechenden Bedeutungsvarianten der Wortformen sind in Tabelle 2.1 auf der nächsten Seite aufgeführt.

Klasse	Anzahl Worte	Ø[13]BV
Nomen	20	5,8
Verben	32	6,31
Adjektive	5	10,2
Summe	57	6,47

Tabelle 2.1: SENSEVAL-3 Inventar der Bedeutungsvarianten (BV)

Als Vorgabe für die (für alle Verfahren, also auch derjenigen mit Verwendung von Trainingsdaten) anzustrebende Leistung wurde das sog. *Inter-Tagger Agreement* gewählt, also der Prozentsatz der übereinstimmenden, durch die menschlichen Editoren vorgegebenen Bedeutungsvarianten. Das Inter-Tagger Agreement lag für die beschriebenen Daten der English Lexical Sample Aufgabe bei ungefähr 67,3%.

Zusätzlich wurde der Wert für *Agreement by Chance* bestimmt, d.h. für die Übereinstimmung der menschlichen Bewertungen mit einem zufällig ausgewählten Wert für die Bedeutungsvariante. Hierfür wurde das Verfahren der sog. Kappa-Statistik (vgl. [Cohen, 1960]) angewendet, um den für diese Arbeit bedeutsamen Wert zu erhalten: den *micro-average* Wert für *K* (Durchschnittswert der Anzahl der Bedeutungsvarianten, und des zufällig ausgewählten Wertes über alle zu bestimmenden Wörter) Für micro-average *K* beträgt der Wert für eine zufällig richtige Auswahl 0,20, es besteht also ein Verhältnis von 1 zu 5.

2.4 Zwischenergebnis

In diesem Kapitel wurde WSD als aktive, in ihrer Definition nicht unumstrittene Disziplin vorgestellt und zu ähnlichen Problemstellungen, wie Word Sense Discrimination abgegrenzt.

Die WSD-bezogene Forschung ist in der Topologie der NLP-Forschung sehr komplex verteilt und läßt sich nur anhand verschiedener Dimensionen charakterisieren, von denen die wichtigsten Richtungen vorgestellt wurden:

Eine Unterteilung anhand der Nutzung von Strukturbeschreibungen ist möglich, da häufig verschiedene Ebenen der Semiotik für WSD-Zwecke eingesetzt werden, wie auch subsymbolische und graphenorientierte Methoden. Hierbei läßt sich feststellen, daß einfache syntaktische oder lexikalische Muster eine flache Verarbeitung ohne aufwändiges Parsing ermöglichen, sich dennoch Aspekte der Textbedeutung ableiten lassen.

Eine weitere Charakterisierungsmöglichkeit besteht aufgrund der verwendeten Datenquellen: korpusbasierte oder Vektorraumverfahren stellen eine Alternative zu traditionellen bzw. linguistisch orientierten Ansätzen dar. Tiefe Ansätze, wie bei der gemischten Verwendung von MRDs, Korpora und Ontologien, besitzen eine

starke eigene Betrachtungsweise der Textbedeutung, die sich wesentlich auf die semantische Interpretation syntaktischer Strukturen stützt, bzw. auf die Auswertung semantisch angelegter Strukturen. Versuche, korpusbasierte mit wissensbasierten Verfahren zu kombinieren sind häufig recht erfolgreich und nutzen ebenfalls eine Assoziation von semantischer logischer Form zu textuellen Strukturen. Im Unterschied zu tiefen Ansätzen ist die Interpretation einer syntaktische Basis hier aber nicht fest vorgegeben. Besonders Methoden, die direkt (per Suchmaschine) oder indirekt (über Internetverzeichnisse) auf webbasierten Daten beruhen, sind im Kontext dieser Arbeit wichtig. Die Verwendung des WWW als Textkorpus liefert große Textmengen zu verschiedenen Anwendungshypothesen. Die erhaltenen - unvorbereiteten - Daten sind allerdings von vergleichsweise schlechter Qualität, so daß die Verwendung von Internetverzeichnissen anstelle einfacher Suchmaschinen zu bevorzugen ist.

Zusätzlich lassen sich WSD-Systeme in überwachte und nicht-überwachte Systeme unterteilen, wobei allerdings für die Darstellung und Evaluation des TSR-Ansatzes die überwachten Systeme nicht betrachtet werden.

Im Zusammenhang hierzu können Systeme weiterhin nach ihrem Status als „Single-Method“ System oder Hybridsystem klassifiziert werden, wobei hybride Verfahren in der Regel zwar die besten Ergebnisse liefern, die Beiträge der einzelnen untergeordneten Methoden aber durch die Verwendung als kombiniertes System verschleiert werden und sich daher Hybridverfahren für die Vorstellung oder Evaluation des TSR-Ansatzes nicht gut eignen.

Viele der in den entsprechenden Abschnitten angeführten Ansätze finden sich auch real in den SENSEVAL-3 Systemen wieder, wie in Abschnitt 2.3 auf Seite 55 aufgeführt, mit sehr unterschiedlichen Ergebnissen (vgl. [Mihalcea et al., 2004]).

Weiterhin wurden verschiedene übliche Metriken zur Bewertung der Leistung von WSD-Systemen vorgestellt, sowie ein kurzer Überblick über die SENSEVAL-Workshops als Lieferant einer standardisierten Versuchsumgebung für WSD-Experiment gegeben.

Kapitel 3

Theorie und Grundlagen

Wie bereits in der Einleitung erläutert wird, ist eine Vertiefung der Grundlagen, die für NLP- und insbesondere WSD-Zwecke genutzt werden, notwendig, um eine angemessene Repräsentation der Informationen und einen korrekten Transfer des Wissens zwischen Modulen eines NLP-Systems gewährleisten zu können.

Kapitel 2 beschreibt hierzu zwar gängige Methoden der aktuellen Forschung, allerdings werden Kernfragen zur theoretischen Basis von Textbedeutung und Weltwissen meist kaum angesprochen. Aus diesem Grund sollen in diesem Kapitel eben diese Fragestellungen diskutiert werden, sowie deren Relation zu einer möglichen Nutzung von Textbedeutung und Weltwissen in NLP-Systemen - insbesondere zum Zweck der Auflösung von Wortmehrdeutigkeiten.

Der Erläuterung des in dieser Arbeit verwendeten Begriffes von „Textbedeutung“ ist hierbei Abschnitt 3.1 gewidmet. Da diese Interpretation von Textbedeutung vor allem einen pragmatischen Bezug zu Informationen aus verfügbaren Hintergrundwissen besitzt, sollen im Abschnitt 3.4 auf Seite 74 verschiedene denkbare und sinnvolle Quellen von Hintergrundwissen diskutiert werden. Neben dem Bezug von Textbedeutung und Hintergrundwissen spielt die *Strukturierung* von Hintergrundwissen in Form von „Kategorien“, sowie die Nutzung dieser Kategorien eine wichtige Rolle für die Vervollständigung der theoretischen Basis; aus diesem Grund wird die Rolle von Kategorien in den Abschnitten 3.4.2.1 auf Seite 78 und 3.5 auf Seite 89 weiter behandelt.

3.1 Zum Begriff der Textbedeutung

Die begriffliche Klärung dessen, was unter der Bedeutung von Wörtern oder Texten zu verstehen ist, ist ein äußerst schwieriges Problem. Die Sprachphilosophie, die Linguistik und auch die Psychologie haben bis heute keine allgemein zufriedenstellende, umfassende Definition finden können. Wohl aber gibt es viele Erklärungsansätze für einzelne Aspekte der Textbedeutung, wie auch für das Gesamtphänomen.

Frege definiert die Extension, d.h. die Menge der Gegenstände, die unter einen Begriff fallen, als dessen „Bedeutung“ - die Intension, also die Zusammenfassung der Merkmale eines Begriffes und deren Beziehungen untereinander, bezeichnet er als „Sinn“ (vgl. [Frege, 1892]).

Die klassische Linguistik geht dagegen in der Regel von einer auf der begrifflichen Intension basierenden Bedeutungshypothese aus (vgl. [Bärenfänger, 2002]).

Das „Verstehen“ im Sinne der linguistisch orientierten Textverarbeitung erfolgt durch die Interpretation syntaktischer Strukturen, über welche ein Modell der Textbedeutung konstruiert wird - die so genannte logische Form (vgl. [Allen, 1995]). Diese besteht häufig aus Ausdrücken einer bekannten Formalisierung der Logik, z.B. der Prädikatenlogik. Hierbei wird angenommen, daß „Verstehen“ die zielgerichtete Umsetzung der Textbedeutung beinhaltet.

Einfachere Ansätze der automatisierten Textverarbeitung vertreten eine Auffassung der Textbedeutung, in welcher den im Text vorhandenen Wörtern bestimmte Einträge einer Ontologie zugeordnet werden. Dies entspricht im Wesen der von dem Philosophen Ludwig Wittgenstein kritisierten „naiven Auffassung von Bedeutung“, in welcher die Wörter „durch Zeigen“ mit Dingen verknüpft werden (vgl. [Wittgenstein, 1984b]). Viele musterbasierte ontologische Verfahren (vgl. beispielsweise [Soderland et al., 1995, Soderland, 1999]), folgen ebenfalls einer derartigen Semantiktheorie.

O. Bärenfänger faßt in [Bärenfänger, 2002] wesentliche Kritikpunkte dieser klassischen Ansätze zusammen:

- Traditionelle (intensionale) Semantiken reduzieren die Bedeutung eines Wortes häufig auf ein gerade hinreichendes Minimum von Merkmalen, ohne jedoch der Mehrdimensionalität der Wortbedeutung gerecht zu werden - also beispielsweise Fragen danach, welche Merkmale „optional“ sind (und aus welchen Gründen), welche zusätzlichen Merkmale hinzugenommen werden können, ohne die essentielle Bedeutung zu verändern etc.
- Merkmale von Wörtern werden nicht unterschiedlich gewichtet – damit können Begriffe wie „Achten“, „Schätzen“, „Lieben“, etc. nicht semantisch differenziert werden.
- Wortbedeutungen werden als diskrete Kategorien begriffen. Damit sind vage Begriffe wie „Einsetzen der Dämmerung“ nicht faßbar.
- Das allgemeine Mehrdeutigkeitsproblem gilt als ungelöst.
- Die Frage, welche Merkmale „essentiell“ sind, bzw. zur Menge der minimalen Merkmalen gehören, ist nicht geklärt.

Während die o.a. Punkte prinzipielle Problembereiche klassischer Merkmalssemantiken betreffen, gibt es darüber hinaus weitere, aus der computerlinguistischen Praxis motivierte Kritikpunkte:

- Die merkmalsbasierte, semiotikorientierte Textverarbeitung hängt stark von der Qualität der lexikalischen und syntaktischen Analyse ab, insbesondere können nur vollständig analysierte Sätze überhaupt in eine logische Form überführt, d.h. „verstanden“ werden und alle Fehler können in gleichem Maße zur Falschheit des Ergebnisses beitragen. Dies entspricht offensichtlich nicht den sprachlichen Tatsachen, da kleinere syntaktische Fehler für das Verstehen eines Textes nicht dieselbe Relevanz besitzen, wie sehr grobe Strukturfehler.
- Merkmalssemantiken beziehen sich zudem üblicherweise auf die Bedeutung von Sätzen. Diskurstheorien, also Theorien über satzübergeordnete Strukturbildungen, werden in der Regel nicht unterstützt.

Diese Kritik am herkömmlichen Verständnis von Textbedeutung läßt vermuten, daß eine andere Sichtweise möglicherweise erfolgreicher wäre – besonders in Bezug auf die Behandlung von Phänomenen textueller Mehrdeutigkeiten.

Im Mittelpunkt der Auffassung von Textbedeutung, die im Zusammenhang mit der vorliegenden Arbeit vertreten wird, steht der wittgenstein'sche Begriff der „Familienähnlichkeit“ von Wörtern und seine Ausführungen zum Sprachgebrauch, sowie eine hierarchische Gliederung von pragmatisch orientierten Kategorien.

Der Philosoph Ludwig Wittgenstein beschreibt in seinen "Philosophischen Untersuchungen" seine Vorstellungen von Bedeutung wie folgt:

1. Die *Familienähnlichkeit* zwischen Begriffen ermöglicht es, diese bestimmten *semantischen Kategorien* zuzuordnen. Wittgenstein erläutert dies am Beispiel der Kategorie "Spiel": er bemerkte, daß es es keine Menge von Merkmalen gibt, die sich uneingeschränkt für alle Spiele angeben lassen. Er schließt hieraus, daß es verschiedene Mengen von (inhomogenen) Merkmalen gibt, die von einigen - aber nicht allen - Elementen einer Kategorie geteilt werden (vgl. [Wittgenstein, 1984b, S. 278, §67]). Beispielsweise geht es beim Fußball um das Gewinnen, und um das Ballspiel. Schach ist ein Brettspiel und es geht um das Gewinnen und beim Ballwerfen wiederum geht es um das Spielen mit einem Ball. Hierdurch entsteht ein „Netzwerk“ von Merkmalen, die auf unterschiedliche Weise von den Elementen der Kategorie Spiel geteilt werden. Dieses Netzwerk bestimmt die von Wittgenstein intendierte Auffassung von der Familienähnlichkeit der Begriffe.
2. Die Bedeutung der Wörter wird in sehr vielen Fällen (aber nicht allen) über ihren *Gebrauch* bestimmt . Wittgenstein distanziert sich damit von denotationalen Ansätzen, in denen beispielsweise einem Begriff die Bedeutung fehlen würde, sobald der referenzierte Gegenstand nicht mehr existiert (vgl. [Wittgenstein, 1984b, S. 262, §§43ff]). An anderer Stelle beschreibt er seine Vorstellung des Gebrauchs als ihre „alltägliche Verwendung“, wobei die Umstände des Gebrauchs eine wichtige Rolle spielen (vgl. [Wittgenstein, 1984b, S. 300, §§116,117ff]).

Der derart über Gebrauch, Kategorisierung und inhaltliche Ähnlichkeiten definierte Begriff der Textbedeutung wird auch dem in dieser Arbeit entwickelten TSR-Ansatz zugrunde gelegt: die Objekte der Textbedeutungsrepräsentation des Ansatzes sollen eine Familienähnlichkeit von Begriffen beschreiben, die aus deren Gebrauch resultiert. Es handelt sich also bei der „TSR-Semantik“ um eine Auffassung, die sowohl semantische, als auch pragmatische Bezüge (insbesondere durch den Bezug zum Weltwissen über das Wissen um den Gebrauch der Wörter) beinhaltet und die ihrer Natur nach vage ist.

Exkurs Sprechakttheorie Tatsächlich werden die „Philosophischen Untersuchungen” Wittgensteins häufig als Wendepunkt der Sprachphilosophie betrachtet, der den Begriff der sprachlichen Handlung einführt (vgl. [Ohler, 1986]). Diese Auffassung, die in der Prägung des Begriffs der „Sprechakte” und deren Untersuchung durch Austin (vgl. [Austin, 1968]) und - vertiefend - durch Searle (vgl. [Searle, 1971]) ihren Ausdruck findet, ist eindeutig pragmatisch geprägt, wenngleich sich bei Searle auch signifikante Unterschiede zu den Ansätzen Wittgensteins finden (vgl. [Ohler, 1986, Wallner, 1983]). Dennoch ist nicht der Sprechakt vordergründiger Gegenstand der Betrachtung, sondern das Wort (und, weiterführend, das Textfragment). Die oben eingeführte Theorie soll Beziehungen zwischen Wörtern und Textfragmenten beschreiben können, auch wenn diese eben nicht im pragmatischen Zusammenhang eines Sprechaktes stehen.

3.2 Wichtige Positionen der Computerlinguistik

Der in dieser Arbeit vorgestellte TSR-Ansatz entspricht weder in seiner theoretischen Ausrichtung, noch in seiner Realisierung dem gängigen Verständnis von Textbedeutung in NLP-Systemen. Aus diesem Grund sollen einige wichtige, diesbezügliche Positionen aus dem Bereich der Computerlinguistik vorgestellt werden sollen.

Die Repräsentation von Textbedeutung kann - im konventionellen Verständnis von Semantik - entweder auf einfachen musterbasierten Heuristiken (dies wird häufig als „flache“ Verarbeitung bezeichnet), oder auf komplexeren Sprachphänomenen und -strukturen basieren (oft als „tiefe“ Verarbeitung verstanden). Eine andere wichtige Dimension ist die Komplexität der Bedeutungsanalyse: einige Ansätze beziehen ihre Informationen ausschließlich aus der lexikalischen oder syntaktischen Ebene, in anderen Fällen erfolgt eine vollständige, oft logikbasierte Analyse. Außerdem ist entscheidend, ob überhaupt eine semantische Theorie zugrunde gelegt wird, oder ob eine solche Theorie zugunsten eines rein empirischen Ansatzes aufgegeben wird. Ein weiteres Kriterium ist die pragmatische Anwendung: inwieweit werden externe Informationen herangezogen, und zu welchem Grad wird der textuelle Kontext erschlossen? Schließlich ist es nicht unerheblich, ob ein Ansatz

auf verschiedenen Granularitätsebenen funktioniert, d.h. ob er z.B. auf die Wortebene oder auf die Satzebene beschränkt ist, oder einen Wechsel zwischen den Ebenen gestattet.

Einige maßgebliche Ansätze sollen im Weiteren exemplarisch und anhand wichtiger Vertreter vorgestellt werden, um anschließend über ein Raster eine Abgrenzung vornehmen zu können:

BOW Die in der Majorität der flachen Verfahren genutzte Repräsentationsweise der Vektorraummodelle (bzw. Bag-of-Words oder BOW Modelle) stellt nicht die Bedeutung der Texte dar, sondern lediglich die Texte selbst - wobei die Elemente der Vektoren durch Wörter, Terme, Phrasen etc. gebildet werden können. Derartige kollokative Verfahren werden von Velardi et al. (vgl. [Velardi et al., 1991]) als „flach“ bezeichnet, da sie nicht direkt die Bedeutung von sprachlichen Ausdrücken beschreiben oder analysieren, sondern deren Verwendung. Über die Verwendung wiederum wird versucht, eine bedeutungsvolle Relation zwischen Wörtern und anderen sprachlichen Elementen herzustellen. Wesentliche Nachteile in Bezug auf die Textbedeutung sehen Velardi et al. in der mangelnden Fähigkeit kollokativer Verfahren, Glaube, Vorbedingungen, Wissen um Ursache-Wirkung Prinzipien abzubilden; dennoch attributieren die Autoren flachen Systemen (Im Zusammenhang mit geeigneten semantischen Lexika) einen großen Wirkungsgrad.

LSA Komplexer ist die Bedeutungsrepräsentation im Falle der PLSA (probabilistische Latente Semantische Analyse): Keller und Bengio beschreiben 2005 einen Vergleich zwischen Vektorraum-basierten Textrepräsentationen und solchen auf Basis von PLSA (vgl. [Keller & Bengio, 2005]). Die PLSA (als Weiterführung der LSA, vgl. [Landauer et al., 1998]) versucht, Wörter ähnlicher Bedeutung über das SVD Verfahren (Singular Value Decomposition) zu Termen zusammen zu ziehen. Wörter, Terme, und Dokumente werden hierbei durch Zahlenvektoren repräsentiert. Es ist allerdings bisher unklar, wie genau die Art und Weise der Transformation der numerischen Vektorwerte Einfluss auf die Textbedeutung nimmt.

NN Die von Keller und Bengio entwickelte eigene Lösung der Repräsentation semantischer Inhalte durch Neuronale Netze nutzt eine Kodierung der Bedeutung in Form von Gewichtungen innerhalb der beschriebenen Netzwerke (vgl. [Keller & Bengio, 2005]). Deren Natur bleibt allerdings auch hier verborgen - eine Beweisführung durch ein Rückschlußverfahren ist beispielsweise nicht möglich.

IE Ein wichtiger Ansatz im Bereich *Information Extraction* ist die automatisierte Konstruktion von flachen Text- und Syntaxmustern, welche zur Erstellung von Konzeptinstanzen führen sollten. Ein solches System war beispielsweise CRYSTAL von Stephen Soderland (vgl. [Soderland et al., 1995]). Neuere

Verfahren bedienen sich häufig WWW-gestützter Methoden (vgl. [Bhandari, 1999]).

Syntaxmuster Unter der Verwendung syntaktischer Muster soll ein Verfahren, wie etwa das von Chanen und Patrick (vgl. [Chanen & Patrick, 2004]) oder das von Niu et al. (vgl. [Niu et al., 2003]) etc. verstanden werden, in welchem interpretationsfähige syntaktische Merkmale gefunden werden. Dieser Ansatz läßt sich prinzipiell um diskursbasierte Methoden erweitern, z.B. unter Berücksichtigung lexikalischer Ketten (engl. *lexical chains*, vgl. [Morris & Hirst, 1991]).

Ontologien / Lexika Einfache Bedeutungsdarstellungen verknüpfen Wörter direkt mit externen Datenbasen, z.B. Ontologien (im Mikrokosmos Projekt, vgl. [Mahesh & Nirenburg, 1995b], [Beale et al., 1995]) oder lexikalische Datenbanken wie z.B. WordNet (vgl. [Ramakrishnan & Bhattacharyya, 2003]). Grundlage hierfür ist die einfache Hypothese, der Name stehe für das Ding. Die Betrachtung derartiger konzeptbasierter Verfahren als „flach“ ist aber nicht unumstritten: konzeptuelle Verfahren gelten für Velardi et al. als tief, da diese „tief in die Sprache eingebettet“ seien (vgl. [Velardi et al., 1991]).

PG & LF Durch die formale Ausrichtung des Ansatzes ist die Repräsentationsweise der Bedeutung stark festgelegt, obwohl in der Literatur verschiedene Verfahren zur Erzeugung der logischen Form genannt werden, z.B. durch verschiedene Ausprägungen von Phrasenstrukturgrammatiken - etwa „Generalised phrase structure grammars“ (GPSG), vgl. [Ramsay, 1985], deren Nachfolger „Head-driven phrase structure grammars“ (HPSG), „Lexical functional grammars“ (LFG), vgl. [Kaplan, 1989],u.a. oder von „Link grammars“, vgl. [Sleator & Temperley, 1991],

Die Granularität des Ansatzes bewegt sich lediglich auf Satzebene - andere Textfragmente, z.B. Diskurse, können nur bedingt abgebildet werden.

Im Raster von Tabelle 3.1 auf der nächsten Seite ist dieser Ansatz durch das Kürzel *PG & LF* repräsentiert (für *Phrase Grammar and Logical Form*).

In Tabelle 3.1 auf der nächsten Seite ist ein Vergleich der oben angeführten Verfahren unter Berücksichtigung der verschiedenen Kriterien dargestellt - hierbei wird aber lediglich nach der prinzipiellen Ausprägung („+“ = stark, „–“ = gering) unterschieden. Um den an ihn gestellten Anforderungen gerecht werden zu können, müßte der TSR-Ansatz Eigenschaften in allen Bereichen aufweisen – dies soll mit der infrage gestellten Nennung des TSR-Ansatzes verdeutlicht werden. Im weiteren Verlauf dieses Kapitels und in Kapitel 4 wird aber grundsätzlich argumentiert werden, daß dies auch den Tatsachen entspricht.

	Syntax	Semantik	Pragmatik	Theorie	Granularität
BOW	–	–	+	–	–
LSA	–	–	+	–	+
NN	–	+	+	–	–
IE	–	+	+	+	–
Syntaxmuster	+	–	+	–	+
Ontologien / Lexika	–	+	–	+	–
PG & LF	+	+	–	+	–
TSR [?]	+	+	+	+	+

Tabelle 3.1: Abgrenzungsraster Textverarbeitungsansätze

Die einzelnen Ansätze zeigen - je nach Ausprägung - unterschiedliche Problemstellungen:

- Aufgrund der Komplexität der Aufgabenstellung besteht für Verfahren, die eine starke Verknüpfung von syntaktischer und semantischer Ebene beinhalten (PG & LF) eine relativ hohe Fehlerwahrscheinlichkeit, wobei eine Fehlerbehandlung in einem klassischen monotonen logischen Kalkül, welches häufig zur Umsetzung der logischen Form genutzt wird, nicht vorgesehen ist.

- Es hat sich gezeigt, daß viele Mehrdeutigkeiten auf syntaktischer oder semantischer Ebene nur durch Informationen der pragmatischen Ebene aufgelöst werden können, so daß ein Rückfluss eben dieser Informationen in diese Ebenen stattfinden müsste - dieser wird aber häufig nicht durch ein theoretisches Modell formalisiert. Die Auflösung von Mehrdeutigkeiten ist für die Feststellung der Textbedeutung jedoch offensichtlich fundamental wichtig. Dieser Punkt betrifft vor allem die Ansätze PG&LF und Ontologien/Lexika.

- In semantikschwachen Ansätzen kann die Logik von Aussagen oft nicht nachvollzogen werden (z.B. Negation) - dies gilt in besonderem Maße für den BOW Ansatz

- Ohne einen pragmatischen Bezug sind feine Strukturen und Beziehungen zwischen textuellen Strukturen häufig nicht nachvollziehbar. Verschiedene Einzelverfahren sind zwar oft über Ensemblemethoden kombinierbar, jedoch werden selten Querbezüge aufgedeckt. In der Regel können auch Sprecherintention bzw. Sprechakte nicht nachvollzogen werden, die für das Textverstehen relevant sind. Bedeutungsnuancen, die in der Wahl sprachlicher Feinheiten verborgen sind, werden durch Verfahren ohne pragmatische Komponenten selten erfaßt.

- Allen argumentiert, daß einfachere Verfahren zwar derzeit erfolgreicher erscheinen mögen, die Erforschung komplexer Methoden aber wesentlich bessere Erkenntnismöglichkeiten bietet und damit wahrscheinlich ultimativ erfolgreicher sein wird. Exemplarisch führt er hierzu das bekannte „Eliza"-Programm Joseph Weizenbaums an, welches scheinbar in der Lage ist, Fragen zu beantworten – tatsächlich handelt es sich nur um eine psychologische Irreführung (vgl. [Allen, 1995, S. 8ff]). Nach Allen wären Bestrebungen, eine „echte" Frage-und-Antwort-Funktion zu realisieren, in jedem Fall wünschenswerter, schon durch den zu erwartenden Erkenntnisgewinn bei der Erforschung der Thematik.

Trotz der stark unterschiedlichen Natur der Ansätze finden sich einige Versuche, tiefe und flache Ansätze zu kombinieren:

- Dahlgren berichtet 1992 von experimentell unterstützten Vermutungen, daß Menschen dazu neigen, zunächst eine „flache" Strategie bei der textuellen Informationsgewinnung zu verfolgen, und erst nachgelagert - sofern das angestrebte Erkenntnisziel nicht erreicht werden konnte - das zeitmäßig „teurere" tiefe und inferenzbasierte Teilverfahren zu nutzen (vgl. [Dahlgren, 1992]).

- Crysmann et al. beschreiben eine hybride Architektur, die über eine Mediationsschicht Informationen der verschiedenen Verfahren zu integrieren versucht (vgl. [Crysmann et al., 2001]).

- Neumann et al. untersuchen eine Divide-and-Conquer Strategie, durch die über ein flaches Verfahren topologische Informationen gewonnen werden. Die erhaltenen Strukturen werden in einem nachfolgenden Schritt durch Spezialgrammatiken, d.h. durch ein tiefes Verfahren analysiert (vgl. [Neumann et al., 2000]).

Der oben eingeführte Begriff von „Textbedeutung" behandelt einen Aspekt von Bedeutung, der über die Wittgenstein'sche Familienähnlichkeit von Wörtern, über ihre Verwendung und über ihre Zuordnung zu Kategorien definiert wird. Dies entspricht nicht der intensional ausgerichteten Sichtweise, wie sie beispielsweise in PG&LF und in Ontologien propagiert wird und auch nicht der extensionalen Betrachtungsweise, die u.a. BOW und LSA unterstellt werden kann.

Aus diesem Grund läßt sich der TSR-Ansatz in einigen Anwendungsbereichen – WSD ist nur ein Beispiel hierfür – als alleiniges Verfahren einsetzen, häufig ist aber anzunehmen, daß eine Kombination von TSR- und weiteren Ansätzen erfolgreicher sein würde.

Insbesondere eine Integration von TSR-Methodik und Inferenzmechanismen wäre aus den Gründen, die auch für die Anwendung von Hybridverfahren (s.o.) sprechen, wünschenswert. Dies erscheint auch aus der folgenden Argumentation heraus sinnvoll: Nach Davoudi (vgl. [Davoudi, 2005]) benutzen (menschliche) Leser

extensiv Inferenztechniken, um Bedeutungsmodelle für Texte zu konstruieren, u.a. indem sie Auslassungen durch die Ergebnisse eigener Schlüsse füllen und das Gelesene situativ ausarbeiten - dies gilt nach Angabe des Autors selbst für sehr einfache reale Texte. Damit ist das Ziehen von Schlüssen ein essentieller Bestandteil des Verstehensprozesses. Inferenz wird durch den Autor als ein kognitiver Prozess betrachtet, der zur Konstruktion einer Bedeutungsrepräsentation genutzt wird. Dieser Prozess kommt zudem offenbar schon zu Beginn des Verstehensprozesses zur Anwendung. Davoudi definiert das „Textverstehen“ in diesem Zusammenhang als Konstruktion einer kohärenten Repräsentation der Textinformation - dies beinhaltet auch die durch Inferenzprozesse gewonnene „implizite“ Information. Damit stellen die Wörter und Phrasen jeglicher sprachlicher Kommunikation lediglich einen groben Umriß der gesamten übertragenen Information dar: der Leser muß, um geschriebene Sprache verstehen zu können, mehr Informationen nutzen können, als durch den Text selbst geboten werden (z.B. Hintergrundwissen, Weltwissen, etc.).

3.3 Korrelation zur Prototypensemantik

Ebenso wie ähnliche Forschungen im Bereich der kognitiven Psychologie (vgl. Abschnitt 3.5 auf Seite 89) wurden auch prototypische Effekte in Reaktionszeitexperimenten erforscht. Aus den Ergebnissen wurde hieraus zunächst die ebenfalls auf dem Familienähnlichkeitsbegriff Wittgensteins beruhende Prototypentheorie, dann die Prototypensemantik entwickelt (vgl. [Bärenfänger, 2002]).

Die Prototypensemantik ist daher eine aus dem Umfeld der Prototypentheorie entstandene Semantik, die auf linguistischen und psychologischen Grundlagen beruht. Zentrales Thema ist hierbei die Frage nach dem Wesen der Textbedeutung, insbesondere von Wortbedeutung (vgl. [Bärenfänger, 2002, Overberg, 1999]).

Wichtige Prinzipien der Prototypensemantik sind die

- *prototypische Auffassung der Wortbedeutung*, d.h. der Bezug zu einem „Prototyp“: Dieser wurde zunächst als mentale Repräsentation eines besonders typischen Vertreters einer Kategorie definiert (vgl. [Meinhardt, 1984]), in neueren Veröffentlichungen steht jedoch mehr der Prototyp als „Grad der Repräsentativität eines Objektes“ im Vordergrund (vgl. [Bärenfänger, 2002, S. 9]). Damit wird nicht mehr von einem „Prototypen“ gesprochen, sondern genauer von „prototypischen Effekten“.

- *Familienähnlichkeit* der Begriffe im Wittgenstein’schen Sinne: prototypische Eigenschaften sind solche, die für eine Gruppe von Objekten, die durch ihre Familienähnlichkeit verbunden ist, repräsentativ scheinen. Damit müssen bestimmte Objekte nicht unbedingt alle Merkmale besitzen, die dem Prototypen zugesprochen werden - so ist beispielsweise das fliegen für Vögel prototypisch, auch wenn z.B. Pinguine nicht fliegen können (vgl. [Bärenfänger,

2002, S. 10], [Overberg, 1999, 3.4.1]). Es ergibt sich eine *Kategorisierung* von Wörtern, wobei sich die Wortbedeutung über die Zugehörigkeit zu einer Kategorie inhaltlich ähnlicher Wörter definiert (vgl. [Overberg, 1999]). Zu beachten ist hierbei, daß es sich um eine graduelle Zugehörigkeit handelt, wie im folgenden Punkt verdeutlicht wird.

- Begriffe des *prototypischen Zentrums* und der *prototypischen Peripherie:* Besonders typische Vertreter einer Kategorie stehen hierbei dem Zentrum nahe und eher untypische Vertreter werden der Peripherie zugeordnet (vgl. [Meinhardt, 1984]). Im Zentrum besteht also ein maximaler Grad der Repräsentativität. Man spricht auch von der „*Typikalität*“ von Merkmalen, wenn die Zuordnung dieser Merkmale zur Positionierung nahe des prototypischen Zentrums führt (vgl. [Blutner, 1995]).

Die Stärken des Ansatzes liegen vor allem in der Behandlung unscharfer Sprachphänomene und in der Betrachtung der Wortbedeutung in Hinblick auf kategorische „semantische Zentren“, da hierdurch viele Probleme klassischer Merkmalssemantiken aufgelöst werden (Mehrdeutigkeit, Abgrenzungsprobleme diskreter Kategorien, Gleichwertigkeit semantischer Merkmale, Eindimensionalität des Bedeutungsbegriffes, vgl. [Bärenfänger, 2002], [Blutner, 1995]). Unter einer solchen Merkmalssemantik soll hier eine Semantik verstanden werden, die auf die Intension der Begriffe, oder auf ihre Extension ausgerichtet ist (vgl. [Blutner, 1995]).

Nach Bärenfänger liegen die Schwächen der Prototypensemantik vor allem in der fehlenden klaren Definition des Prototypbegriffes selbst (dieser wird lediglich operational verwendet) und in der ungeklärten Frage nach dem Anwendungsbereich - beispielsweise besitzen Zahlenkategorien selten prototypische Struktur (z.B. ist die Zahl 3 nicht mehr oder weniger prototypisch als die Zahl 4).

Die Prototypensemantik kann als komplementär zur traditionellen Merkmalssemantik begriffen werden, da sich zwar die Bedeutungsbegriffe und auch die Herleitung der Bedeutung erheblich unterscheiden, sich beide Theorien jedoch aufgrund der Verschiedenartigkeit ihrer „Lücken“ gut ergänzen können. Diese Ansicht wird auch von Helbig vertreten, der Prototypensemantik und Merkmalssemantik – bei ihm „Kategorielle Semantik“ genannt – als ein sich gegenseitig im semantischen Sinne ergänzendes Paar von Modellen begreift (vgl. [Helbig, 2001]). Bärenfänger schlägt 2002 entsprechend vor, für bestimmte Arten von Kategorien jeweils entweder die Prototypensemantik oder die klassische Merkmalssemantik zu verwenden (vgl. [Bärenfänger, 2002]).

In Bezug auf die Theorie des TSR-Ansatzes lassen sich einige offensichtliche Punkte feststellen:

- Sowohl Prototypensemantik als auch TSR-Theorie begründen sich auf der Familienähnlichkeit zwischen Begriffen. Beide Ansätze sind damit inhärent

auf „vage“ Begriffe ausgerichtet - über verschiedene Gewichtungsmechanismen kann der Grad der Ähnlichkeit festgestellt werden. Beide Ansätze beruhen ebenso auf einem Kategorienbegriff, der über die Zuordnung von Wörtern definiert wird. Bezüglich der Vorstellung einer „Prototypikalität“ in diesem Sinne sind Prototypensemantik und TSR-Theorie stark verwandt.

- In der Prototypensemantik stehen Zentrum und Peripherie einer Kategorie im Vordergrund, während der TSR-Ansatz zwar die Relevanz einer Kategorie in Bezug zum Begriff beinhaltet (über die Gewichtung von Pfaden), stärker jedoch die Position eines Begriffes zwischen den jeweiligen Zentren verschiedener Kategorien betrachtet. Dies resultiert aus der unterschiedlichen Interpretation von Zentralität: in der Prototypensemantik stehen Objekte nahe dem prototypischen Zentrum, die eine große Anzahl von prototypischen Merkmalen aufweisen. In der TSR-Theorie ist die Zentralität über die Verwendungsrelevanz der Begriffe definiert: Innerhalb einer Kategorie steht ein Objekt dem Zentrum nahe, wenn der entsprechende Begriff eine hohe Relevanz in zur Kategorie gehörenden Texten besitzt. Damit kann ein Objekt auch verschiedenen Zentren nahe sein - ein Umstand, den die Prototypensemantik nicht problematisiert.

- Während die Prototypensemantik offenbar auf ontologische Kategorien abhebt (Beispiele benennen in der Regel ontologische Klassen wie *Vogel*, *Farbe*, etc.), entstehen die Kategorien innerhalb der TSR-Theorie aufgrund ihrer sprachlichen Verwendung. Damit entsteht für die TSR-Theorie nicht die Frage nach dem Anwendungsbereich: dieser ist bereits in der Definition der tatsächlichen Kategorisierung enthalten. Zum Beispiel ist nach der Prototypensemantik die Zahl 1969 nicht typischer als die Zahl 2, in der TSR-Theorie finden sich zu beiden Zahlen viele weitere Verwendungen, so daß eine Unterscheidung sehr wohl möglich ist.

- Die Prototypensemantik ist auf die Diskussion der Wortbedeutung eingeschränkt, da Kategorien und prototypische Effekte für Sätze oder Texte nicht definiert werden. Der TSR-Ansatz als praktische Anwendung der TSR-Theorie ist dagegen in der Lage, Texte von sehr unterschiedlicher Granularität zu behandeln.

Aus dieser Diskussion läßt sich folgern, daß der TSR-Ansatz auf ähnlichen Prämissen beruht, wie die Prototypensemantik (und auch die Vorstellung der Familienähnlichkeit zwischen Begriffen teilt), in der praktischen Ausprägung jedoch viel stärker auf den tatsächlichen Sprachgebrauch ausgerichtet ist als diese. Damit lassen sich die Argumente der Abgrenzung zur Merkmalssemantik für den TSR-Ansatz übernehmen, dieser teilt jedoch nicht die angeführten Schwächen der Prototypensemantik. Darüber hinaus behandelt die Prototypensemantik durch die Konzentration auf die Wortbedeutung innerhalb einer bestimmten Kategorie einen kleineren

Ausschnitt der Textbedeutung als der TSR-Ansatz – der Anwendbarkeitsumfang ist jedoch insbesondere für praktische Probleme der Textverarbeitung wichtig.

3.4 Überlegungen zur Wissensbasis

Von zentraler Bedeutung für die praktische Umsetzung der im vorhergehenden Abschnitt dargestellten Theorie und der weiter in dieser Arbeit vorgeschlagenen Methoden ist – wie aus der Einleitung bereits hervorgeht – die angemessene Verfügbarkeit von Hintergrundwissen für WSD-Zwecke.

Ein gebrauchsorientierter Begriff von Textbedeutung verlangt offensichtlich nach Wissen über den allgemeinen Gebrauch der Wörter und der Sprache. In diesem Sinne stellt das World Wide Web das größte bekannte Korpus dar, in welchem Informationen über den Gebrauch der Wörter (durch den Gebrauch selbst) vorhanden sind – wie von Kilgariff und Grefenstette argumentiert wird (vgl. [Kilgarriff & Grefenstette, 2003]). Von diesen werden auch einige Vorteile des „Web als Korpus“ dargelegt:

- Das WWW bietet als Korpus das größte Datenvolumen, größer als alle anderen bekannten Korpora. Im Zusammenhang mit dem Zipf'schen Gesetz der Verteilung von Wortfrequenzen wird von den Autoren verdeutlicht, warum ein so großes Korpus in einigen Fällen benötigt wird: beispielsweise beinhaltet das British National Corpus (BNC) trotz seiner Größe (~100 Millionen Wörter) für den Großteil der 10.000 Wörter der englischen Kernsprache jeweils weniger als 50 Instanzen, so daß für diese Wörter eine nur unzureichende Berechnung statistischer Zusammenhänge erfolgen kann (z.B. im Rahmen von Kookkurrenzberechnungen). Um einen geschätzten Mengenvergleich zu bieten, geben die Autoren exemplarische Frequenzen für bestimmte Phrasen an, z.B. ist der Ausdruck „medical treatment“ im BNC nur 414 mal vertreten, während die Suchmaschine „AlltheWeb“ den Ausdruck nach Angabe von Kilgariff und Grefenstette 2003 ca. 1,5 Millionen mal gefunden hat.

- Das WWW ist frei verfügbar und bietet für viele Inhalte einen schnellen Zugriff

- Das WWW ist inhärent multilingual, während andere große Korpora in der Regel einsprachig sind.

Kilgariff und Grefenstette erwähnen allerdings auch verschiedene computerlinguistische Nachteile:

- Weil die Benutzung von Suchmaschinen die Standardmethode ist, Inhalte des WWW zu erschließen, folgen viele der Nachteile direkt hieraus:

  - In der Regel liefern Suchmaschinen nur eine begrenzte Anzahl von Ergebnissen, auch wenn theoretisch viel mehr Ergebnisse im Index enthalten sind
  - Für Suchergebnisse wird oft kein textueller Kontext geliefert, bzw. ist dieser Kontext zu klein (nur einige Wörter in der Umgebung des Suchbegriffes)
  - Ergebnisse unterliegen - aus linguistischer Sicht betrachtet - fremdartigen Einflüssen, z.B. werden bestimmte Ergebnisse vorrangig behandelt, weil die Suchmaschinenbetreiber dies als Dienstleistung verkaufen
  - Statistische Angaben (z.B. die Anzahl der gefundenen Instanzen) sind oft unzuverlässig, da sie teilweise von technischen Faktoren (Belastung der Suchmaschine, etc.) abhängen

- Es sind keine präzisen statistischen Berechnungen[1] möglich, da das Gesamtvolumen des WWW unbekannt ist, d.h. die Anzahl der Wörter bzw. Seiten im WWW.

Aus Sicht der bisherigen theoretischen Diskussion ist noch ein weiterer nachteiliger Punkt relevant: das WWW ist inhärent unstrukturiert, d.h. es gibt keine Merkmale des WWW, die im Sinne der Semantik des TSR-Ansatzes direkt nutzbar sind, weil sie Annahmen über die Familienähnlichkeit von Wörtern zulassen, oder weil Domänen- bzw. Kategorieninformation vorhanden wäre.

Um die Vorteile des WWW zu nutzen und gleichzeitig die o.a. Nachteile auszugleichen, lassen sich sogenannte Internetverzeichnisse verwenden, wie in Abschnitt 3.4.2 auf Seite 77 beschrieben wird - zuvor sollen jedoch einige mathematische Grundlagen, die Anwendung in Internetverzeichnissen finden, erläutert werden.

3.4.1 Graphentheorie

Die folgenden Abschnitte über Internetverzeichnisse, sowie die Erläuterungen zum TSR-Ansatz in den weiteren Kapiteln dieser Arbeit nutzen Teilgebiete der Graphentheorie, insbesondere der Baumtheorie. Aus diesem Grund soll an dieser Stelle eine kleine Einführung erfolgen, um eine gemeinsame Basis der relevanten Grundlagen herzustellen (Maßgebliche Quellen waren wieder die „Kleine Enzyklopädie Mathematik“ von Gellert et al. ([Gellert et al., 1971]) und das Buch „Fundamente der Graphentheorie“ von L. Volkmann ([Volkmann, 1999])).

Da Graphen das allgemeinere Konstrukt darstellen, folgen zunächst die relevanten Definitionen für Graphen.

[1] Hierunter sind keine Abschätzungen, beispielsweise von lexikalischen Distributionen gemeint, sondern die Verwendung statistischer Formeln, die die Bekanntheit einer Menge voraussetzen – beispielsweise solche zur Berechnung von Häufigkeitsverteilungen.

DEFINITION 1: Ein gerichteter Graph G ist ein Tupel (V,E), V ist eine Menge von Knoten (engl. Vertices) und E eine Menge von Kanten (engl. Edges). Es gilt $E \subseteq V \times V$.

Eine *Schleife* entsteht, wenn Start- und Endknoten einer Kante gleich sind. Es handelt sich um einen *endlichen Graphen*, wenn V endlich ist.

DEFINITION 2: Ein *gefärbter Graph* $G = (V,E,f)$ entsteht durch das Hinzufügen einer Abbildung f zur Graphendefinition, wobei f eine Abbildung in die Menge der natürlichen Zahlen ist, d.h. jedem Knoten v wird ein Wert $f(v) = n \in \mathbb{N}$ zugewiesen.

Wenn f in $\mathbb{R}$ abbildet, spricht man von *gewichteten Graphen,* damit erhält jeder Knoten v ein *Gewicht* $f(v) = x \in \mathbb{R}$. Da auch Kanten gewichtet sein können, spricht man auch vom *Knotengewicht*.

DEFINITION 3: Ein *beschrifteter Graph* $G = (V,E,f,l)$ wird zudem durch eine Beschriftung l (engl. Label), d.h. eine symbolische Abbildung, ergänzt. Die Beschriftung weist jedem Knoten einen Namen zu. Die Beschriftung wird oft auch als *Benennung* bezeichnet.

DEFINITION 4 : Gegeben sei ein Graph $G = (V,E)$ und eine Folge von Knoten $W = (v_1, v_2, \ldots, v_n)$ mit $v_i \in V$, dann nennt man W einen *Weg* in G.

Ein Graph ist *zusammenhängend*, wenn es zwischen jeweils zwei paarweise unterschiedlichen Knoten mindestens einen Weg gibt.
W ist ein Pfad in G, wenn kein Knoten in W mehrfach auftritt, d.h. es gilt $\forall i, j : v_i \neq v_j$ wenn $i \neq j$.
Der Wert $n-1$ wird als *Länge* des Pfades W bezeichnet.

Für Graphen sind verschiedene, den Mengenoperationen entlehnte, Operationen definiert:

DEFINITION 5: Gegeben seien zwei Graphen $G_1 = (V_1,E_1)$ und $G_2 = (V_2,E_2)$, dann läßt sich ein Graph $G_1 + G_2$ durch Vereinigung der Knoten- und der Kantenmengen konstruieren. Entsprechend läßt sich ein Graph $G_1 - E_2$ konstruieren, der durch Entfernung der Knoten E_2 aus G_1 entsteht (äquivalent kann ein Graph $G_1 - V_2$ gebildet werden). Voraussetzung für die Anwendung dieser Operationen ist, daß G_1 und G_2 vom selben Typ sind.

DEFINITION 6: Gegeben sei ein Graph ohne Mehrfachkanten $G = (V,E)$. Ein Graph $G_1 = (V_1,E_1)$ heißt *Teilgraph* oder *Untergraph* von G, wenn E_1eine Teilmenge von E und V_1 eine Teilmenge von V ist. Umgekehrt heißt G *Obergraph* von G_1.

Handelt es sich bei $G = (V,E,f)$ um einen gewichteten Graphen, dann muß die Gewichtung f_1 des Subgraphen $G_1 = (V_1,E_1,f_1)$ der Gewichtung f entsprechen, d.h. es gilt für jeden Knoten v die Relation $f(v) = f_1(v)$.

Mit Hilfe der oben angeführten Definitionen ist nun eine Bestimmung von sogenannten „Bäumen“ möglich.

DEFINITION 7: Ein *Baum B* ist ein zusammenhängender Graph (V,E) ohne Schleifen und ohne Mehrfachkanten. Damit sind je zwei verschiedene Knoten durch genau einen Pfad verbunden.

Ein Baum wird als *gerichtet* oder auch als *gewurzelt* bezeichnet, wenn er lediglich gerichtete Kanten besitzt. Hierbei gibt es einen ausgezeichneten Knoten, die so genannte *Wurzel* (engl. root). Alle Kanten zeigen entweder zur Wurzel hin (*In-Bäume*) oder von der Wurzel weg (*Out-Bäume*).
Knoten, die keine Ausgangskante besitzen (d.h. keine Kante, die von ihnen weg zeigt), werden *Blätter* genannt.
Die Knoten v_m, die von einem beliebigen Knoten v durch eine gerichtete Kante unmittelbar erreichbar sind, werden als *Kindknoten* von v bezeichnet. Umgekehrt wird v *Elternknoten* von v_m genannt. Knoten, die denselben Elternknoten besitzen, werden als *Geschwisterknoten* bezeichnet. Die von v aus mittelbar (d.h. über einen Pfad beliebiger Länge) erreichbaren Knoten v_k werden als *Nachfahren* bezeichnet; v ist dann der *Vorfahr* der Knoten v_k.
Die *Höhe* eines gerichteten Baumes ist die Länge des längsten Pfades zwischen der Wurzel und einem Blatt.
Äquivalent hierzu gilt als *Tiefe* eines Knotens die Länge des Pfades zwischen dem Knoten und der Wurzel.
Die *Ordnung* eines gerichteten Baumes ist die maximal auftretende Geschwisterzahl innerhalb des Baumes (d.h. der maximale Ausgangsgrad).

3.4.2 Internetverzeichnisse

Internet-Verzeichnisse (engl. *Web Directories*) dienen der strukturierten angeleiteten Suche (engl. *Browsing*) durch manuell erstellte hierarchisch nach „Kategorien“ gegliederte Listen von Webreferenzen (engl. *Link*). Häufig sind weitere Informationen, z.B. Beschreibungen der Kategorien, Kurzbeschreibungen der Webreferenzen, Querverweise zu anderen Kategorien, etc. enthalten. Eine Kategorie stellt in der Regel - dies ist aber nicht zwingend erforderlich - eine thematische Einteilung dar[2].

In Bezugnahme auf die TSR-Semantik und auf die auf Seite 74 diskutierten Eigenschaften des Web als Korpus lassen sich noch folgende Merkmale von Internet-Verzeichnissen feststellen:

- Die enthaltenen Daten werden zum Teil aus dem WWW erhoben (Webreferenzen, Titel und Beschreibungen der Webreferenzen, falls verfügbar, vgl.

[2]Eine Erläuterung zur Kategorienbildung innerhalb des Open Directory Projects ist unter der URL `subcategories http://dmoz.org/World/Deutsch/guidelines/subcategories.html` zu finden

Abbildung 3.3 auf Seite 81) und zum Teil manuell hinzugefügt (Beschreibungen der Kategorien, Ergänzungen zu den Beschreibungen der Webreferenzen, ebenso). Daher kann sowohl das Vorhandensein „echter“ Gebrauchsinformationen - mit Potential zum Erreichen eines großen Datenvolumens - vorausgesetzt werden (in der Regel deskriptiven Sprachgebrauches, da es sich eben um Beschreibungen handelt), als auch über die manuellen Korrekturvorgänge und Ergänzungen

- Die Kategorien entsprechen weitgehend den „Kategorien“ im Sinne Wittgensteins, wobei durch die Hierarchisierung der Kategorien der Internet-Verzeichnisse eine zusätzliche Strukturierung erfolgt. Über diese Strukturierung sind dann „Entfernungsberechnungen“ zwischen den Kategorien möglich, z.B. über ein „Tree Edit Distance“-Verfahren, welches - grob beschrieben - berechnet, wie groß ein Unterschied zwischen zwei verschiedenen Kategorien ist (vgl. [Bille, 2005] für eine Vorstellung verschiedener diesbezüglicher Algorithmen).

Beispiele derartiger Internet-Verzeichnisse sind das (kommerzielle) Verzeichnis „Yahoo“ (vgl. [Yahoo Inc., 2004]) und das (freie) Gemeinschaftsprojekt „Open Directory Project“ (vgl. [Netscape Communications Corporation, 2004]), abgekürzt *ODP* - welches auch das in dieser Arbeit als Datengrundlage verwendete Internetverzeichnis ist. Das ODP wurde als Datenquelle gewählt, weil es sich um eines der größten derartigen Verzeichnisse handelt und die Daten zudem frei verfügbar und zugänglich sind. Außerdem werden Kategorien und Inhalte des ODP von einer Vielzahl freiwilliger Editoren betreut und unterliegen damit keiner zentralen Steuerung - dies gewährleistet einen durchschnittlichen Gebrauch von Wörtern und Kategorien, der vergleichsweise unabhängig von einer kommerziellen Zielrichtung oder der soziokulturellen Ausrichtung der Ersteller ist. Die Datenstrukturen des ODP werden in regelmäßigen Zeitabständen in Form von XML Dateien veröffentlicht und sind daher gut für die Zwecke dieser Arbeit einsetzbar. Prinzipiell wären jedoch andere vergleichbare Datenquellen ebenso geeignet, die ein großes Datenvolumen enthalten und eine nach Kategorien geordnete Struktur besitzen, wie auch im Abschnitt 3.4.3 auf Seite 87 diskutiert wird.

Die Eigenschaften eines Internetverzeichnisses, die für diese Arbeit wesentlich sind, werden in den folgenden Abschnitten zusammengefasst.

3.4.2.1 Funktion von Kategorien

Ein Internetverzeichnis ist hierarchisch in Kategorien organisiert. Diese Kategorien entsprechen den in Abschnitt 3.3 auf Seite 71 geforderten funktionalen Eigenschaften von Kategorien.

Entfernt man eventuell bestehende Querbezüge (z.B. Aliase), so erhält man eine Baumstruktur (vgl. Abbildung 3.1), d.h. ein Internetverzeichnis kann dann mathematisch als gerichteter, gewichteter, beschrifteter und zusammenhängender Graph

(V, E, f, l) ohne Schleifen oder Mehrfachkanten betrachtet werden. Pfade im Internetverzeichnis, d.h. Folgen von Knoten $W = (v_1, v_2, \ldots, v_n)$ mit $v_i \in V$, in denen kein Knoten mehrfach auftritt, werden durch Verkettung der Beschriftungen $l_1, l_2, \ldots, l_n$ der Knoten denotiert. Der Pfad */Top/Computers* der WWW-Ansicht des ODPs ist zum Vergleich in Abbildung 3.2 auf der nächsten Seite dargestellt, Pfade wie

/ Top / Business
/ Top / Business / Banking
/ Top / Arts / Movies / Titles / 2

lassen sich aus Abbildung 3.1 ableiten.

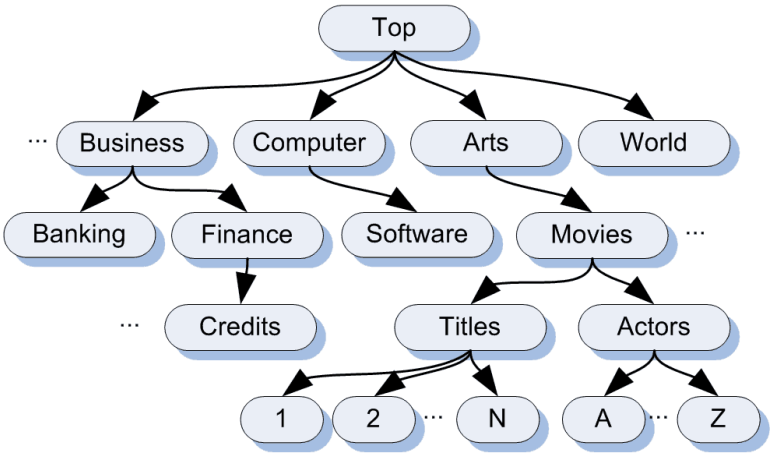

Abbildung 3.1: Exzerpt ODP Struktur

Kategorien in Internetverzeichnissen dienen zum schnellen Auffinden von Inhalten. Sie entsprechend daher nicht der Begrifflichkeit der Kategorien im aristotelischen Sinne, d.h. sie sind eben nicht im aristotelischen Sinne "kategorisch": die in ihnen enthaltene Information kann auch in anderen Kategorien enthalten sein (Kategorien partitionieren die Menge der enthaltenen Informationen nicht). Die enthaltene Information repräsentiert zudem nicht die gesamte, aus den Sachverhalten der Welt ableitbare Informationsmenge, sondern nur die, die von dem jeweiligen Bearbeiter einer Kategorie subjektiv als für die Kategorie relevant erachtet wird. Hieraus folgt, daß die Abgrenzung zwischen den Kategorien insgesamt vage und unpräzise sind.

Beispielsweise enthält das ODP Internetverzeichnis die in Abbildung 3.3 auf Seite 81 angegebene Kategorie, welche einige in ihrer Bedeutung offensichtlich völlig unterschiedliche Einträge beinhaltet.

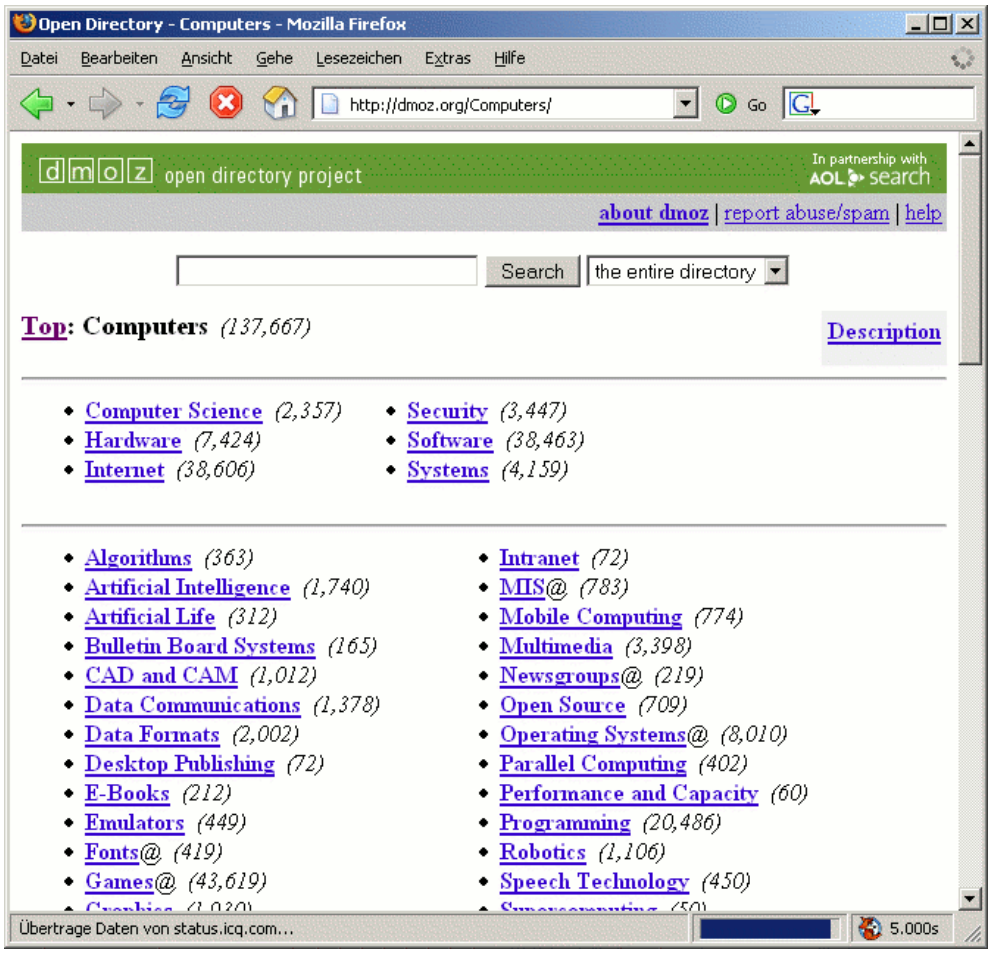

Abbildung 3.2: ODP Ebene */ Top / Computers* (WWW-Ansicht)

Aus diesem Beispiel läßt sich schließen, daß es nicht ausschließlich individuelle Wörter oder Begriffe sein können, die die Einträge ihrer Kategorie zuordnen: z.B. enthalten die Einträge (a) und (b) keine gemeinsamen bedeutsamen Wörter. Damit handelt es sich auch nicht um „semantische Kategorien“ im linguistischen Sinne.

Es lassen sich die Kategorien der Internet Verzeichnisse nun derart interpretieren, daß ihre Inhalte eine Familienähnlichkeit im Wittgenstein’schen Sinne aufweisen. Im oben genannten Beispiel stehen z.B. Filme bestimmter Art ("Independent" Genre, Titel fängt mit "R" an) im Mittelpunkt der Betrachtung. Die begriffliche Bestimmung der Kategorie selbst dient als gemeinsames Merkmal.

Dies läßt sich so zeigen: In den Satzformen

- *„X ist ein guter Film“,*

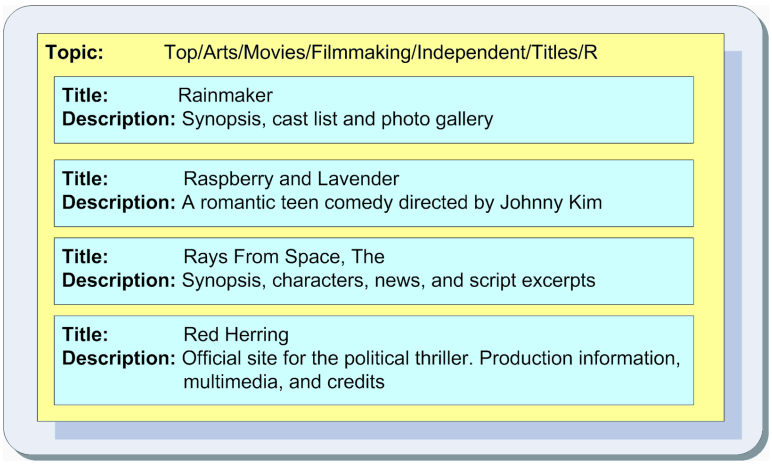

Abbildung 3.3: Einträge einer ODP Kategorie

- *„Neulich habe ich den Film X gesehen“ und*
- *„Der Film X ist recht ungewöhnlich“*

läßt sich der Platzhalter X durch ein beliebig gewähltes Element (a) bis (d) aus Abbildung 3.3 ersetzen - die Bedeutung ist in jedem Falle klar. Offenbar lassen sich die Elemente (a) bis (d) im Merkmalskontext der genannten Kategorie auf dieselbe Weise gebrauchen (lediglich die Referenz ändert sich).

Darüber hinaus können die Elemente des Verzeichnispfades, der zur Kategorie führt, ebenfalls als Träger familienähnlicher Merkmale betrachtet werden - im Beispiel von Abbildung 3.3 wären dies die Merkmale der Begriffe *Arts*, *Movies*, *Filmmaking*, *Independent* und *Titles,* wobei der Begriff *Movies* im Kontext *Arts* zu sehen ist (und demnach keine uneingeschränkte Interpretationsfähigkeit besitzt), *Filmmaking* im Kontext *Arts / Movies* usw. .

Wittgenstein zeigt zwar, daß derartige Merkmale ebenfalls “unexakt” sind, erklärt aber an selbiger Stelle, daß dies nicht notwendigerweise zur Ungültigkeit des Merkmalvergleiches selbst führen muß.

Durch das Beispiel von Abbildung 3.3 wird auch die Auffassung der Bedeutung über ihren Gebrauch deutlich: „Red Herring“ besitzt hier eine Bedeutung als Titel im Zusammenhang mit einem Kinofilm, der die durch die Kategorie angegebenen Merkmale aufweist - und nicht z.B. die umgangssprachliche Bedeutung als ungewöhnlich gefärbtes Exemplar der Gattung Clupea.

Zu beachten ist allerdings, daß ein Internetverzeichnis prinzipiell nur Verweise auf Webseiten, sowie eine kurze Beschreibung der jeweiligen Seite enthält. Dies hat einige bedeutsame Konsequenzen:

- Der Schwerpunkt der enthaltenen Inhalte entspricht den inhaltlichen Gegebenheiten des WWW. Damit stehen weniger alltägliche Situationen und Begriffe im Vordergrund, sondern im WWW beliebte Themen und Begriffe.

- Metasprachliche Ausdrücke finden sich in sehr vielen Kategorien. Da diese Ausdrücke auch Wörter der Umgangssprache enthalten können, besitzen viele derart verwendete Wörter eine ausgeprägte Ungenauigkeit. Das Wort *page* tritt z.B. häufig in der Formulierung „This page is about ...“ auf, die sehr häufig verwendet wird. Damit spiegelt das Auftreten dieses Wortes in bestimmten Kategorien nicht mehr die Familienähnlichkeit zur Kategorie wider[3].

- Titel von Einträgen sind prinzipiell als Titel von Webseiten anzusehen und die Beschreibungen der Einträge beschreiben nicht den Titel des Eintrages, sondern den Inhalt der jeweiligen Webseite. Beispielsweise beschreiben die Begriffe *Synopsis*, *Cast List* und *Photo Gallery* nicht den Film „Rainmaker“ (oder gar den Begriff „Rainmaker“), sondern die Webseite, auf die sich der Eintrag bezieht (und deren Inhalt offenbar vom Film „Rainmaker“ bestimmt wird, sowie von der Tatsache, daß es sich um einen Film handelt).

- Die Kategorien selbst sind vom verwendeten Internetverzeichnis abhängig. Kommerzielle Verzeichnisse wie Yahoo etc. besitzen einen Mitarbeiterstab, der nach internen Maßgaben Webseiten in das Verzeichnis aufnimmt und Einträge redigiert, so daß eine gewisse Konsistenz der Kategorisierung anzunehmen ist. Da das im TSR Rahmen verwendete ODP jedoch – wie bereits angeführt – ein offenes und nach kooperativen Gesichtspunkten erstelltes Internetverzeichnis ist, in welchem die Einträge und auch die Kategorien von jedem WWW-Benutzer erstellt und geändert werden können wird die möglicherweise geringere Konsequenz der Kategorisierung von der Tatsache aufgewogen, daß das ODP offenbar sehr viel besser die tatsächlichen Bedürfnisse der WWW-Benutzer spiegelt.

3.4.2.2 Strukturelle Eigenschaften

Jedem Knoten $v_i \in V$ des Verzeichnisbaumes (V, E, l, d) (vgl. Abbildung 3.4 auf der nächsten Seite) entspricht eine Beschriftung l_i, sowie eine Beschreibung $d_i =< p_i, t_i, R_i, C_i >$ mit den in Tabelle 3.2 auf Seite 84 beschriebenen Merkmalen.

Die WWW-Ansicht eines Knotens mit Pfadangabe (*/ Top / Computers / Software / Operating Systems / Linux / Distributions*), Titel („Debian“), Unterknoten („APT

[3]Eine Umgehungsstrategie für dieses Problem ist die Einführung einer Obergrenze für die Größe von TSRs oder für ihre Breite (zur Definition dieser Begriffe vgl. Abschnitt 4.1.3). Eine echte Problemlösung wurde jedoch noch nicht erarbeitet.

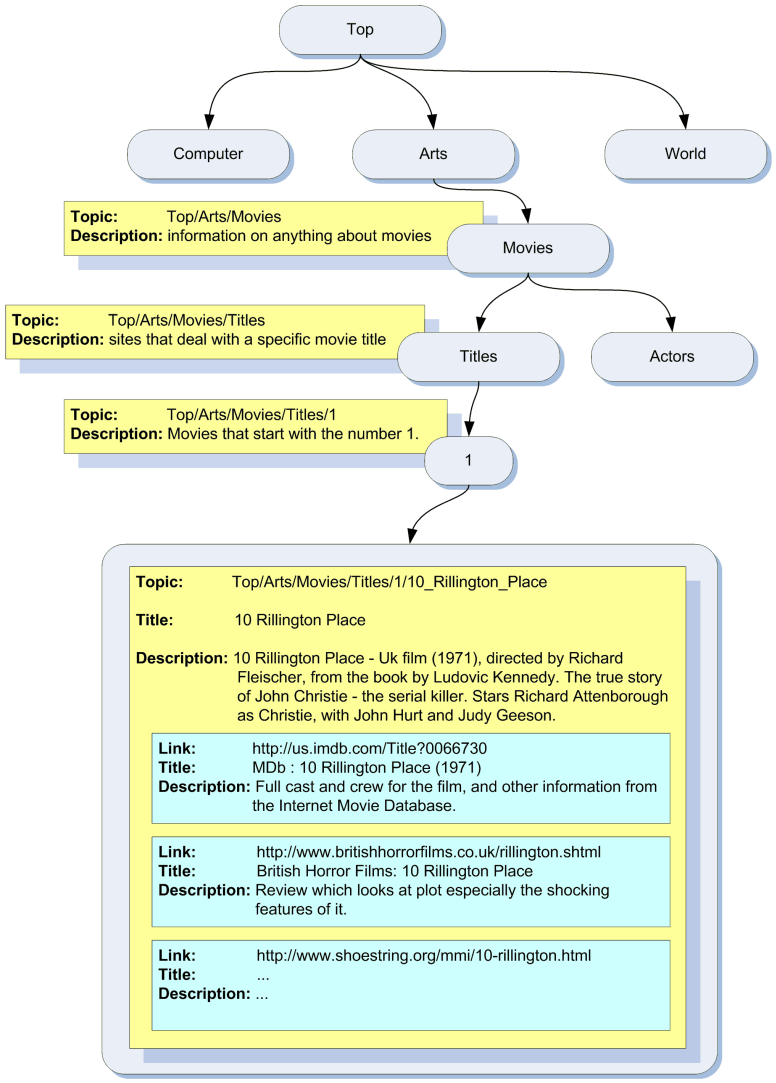

Abbildung 3.4: Exzerpt ODP Knoten

Sources“, „Flavours“, „GNOME“, „KDE“, „People“) und verschiedenen Referenzen („Debian GNU/Linux“, ...) wird beispielhaft in Abbildung 3.5 auf Seite 85 angegeben. Querbezüge („See Also“, „This Category in other languages“) werden nicht ausgewertet.

Zudem weist ein Internetverzeichnis folgende Eigenschaften auf:

- Es gibt einen Wurzelknoten v_0 mit dem Namen $l_0 = Top$” (WWW-Ansicht in Abbildung 3.6 auf Seite 86)
- Die Knoten des Verzeichnisbaumes lassen sich durch Traversion in einer

Symbol	Beschreibung
l_i	Lokale Beschriftung des Knotens v_i. Diese muß nur gegenüber den jeweiligen Geschwisterknoten eindeutig sein, nicht aber gegenüber beliebigen anderen Knoten des Baumes und ist Inhalt des Feldes $Title$ oder letztes Element der Pfadangabe in $Topic$. Die durch die Konkatenation der Bezeichnungen in den Pfaden des Baumes entstehende Zeichenkette p ist in Abbildung 3.4 auf der vorherigen Seite als Feld $Topic$ gekennzeichnet, und kann dem Tupel $n_i$als (optionale) Pfadinformation hinzugefügt werden.
t_i	Eine textuelle Beschreibung. Diese Beschreibung dient der Vorstellung der Kategorie gegenüber dem Benutzer des Verzeichnisses und kann als „Definition“ des Knotens interpretiert werden. In Abbildung 3.4 wird die Beschreibung durch Konkatenation der Felder $Description$ und $Title$ erzeugt.
R_i	Eine beliebig große (ggf. auch leere) Menge von Referenzeinträgen. Jeder Referenzeintrag besteht aus einem WWW-Link und einer Kurzbeschreibung der durch den Link referenzierten Website. Abbildung 3.4 repräsentiert einen Referenzeintrag durch *Link*-Elemente, deren *Title* und $Description$ Felder eben die Kurzbeschreibung darstellt.
C_i	Eine beliebig große (ggf. ebenfalls leere) Menge von Kindknoten, bzw. Unterkategorien, diese wird in Abbildung 3.4 als Pfeil vom Elternknoten zum Kindknoten aufgeführt.

Tabelle 3.2: Baumknoten eines Internetverzeichnisses

sequentiellen Folge $v_0, \ldots, v_n$ mit $n = |V| - 1$ repräsentieren. Hierdurch ist anstelle der einfachen Nennung der Knotennamen eine eindeutige Nummerierung der Knoten v_i über die Indices i möglich. Diese eher technische Eigenschaft ist Voraussetzung für eine Darstellung in Vektorrepräsentation und Einbettung in existierende wortvektorbasierte Verfahren (der Verlust struktureller Information wird hierbei in Kauf genommen).

- Die Semantik der Eltern-Kindknotenbeziehungen ist durch das o.a. Kategorienparadigma vorgegeben.

Obwohl für den TSR-Ansatz derzeit die tatsächlichen Textdaten des ODP genutzt werden, sollen dennoch – um die Möglichkeit zukünftiger diesbezüglicher Weiterentwicklungen zu eröffnen – verschiedene Ansätze aus der Literatur vorgestellt werden, Daten eines Internetverzeichnisses zur Verarbeitung vorzubereiten:

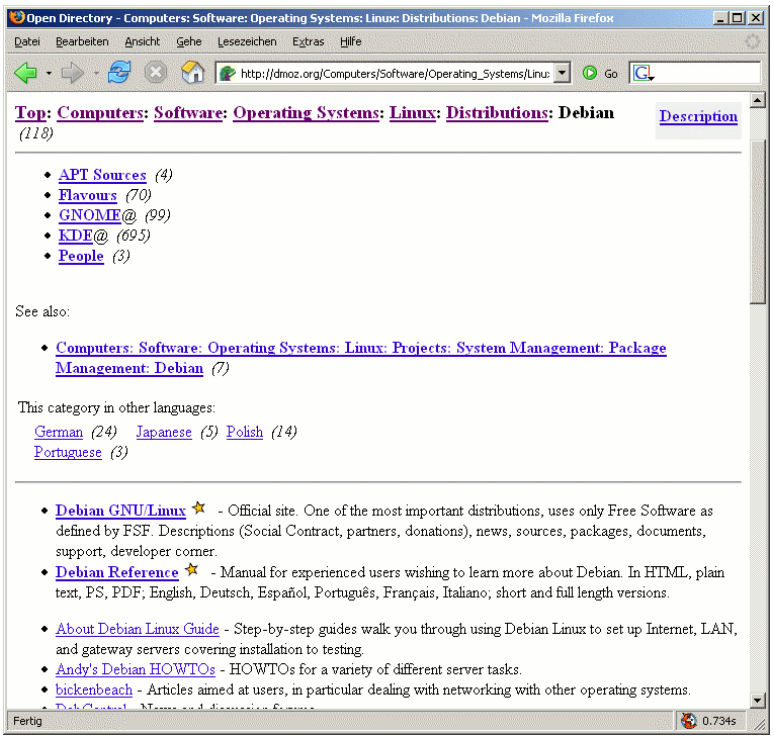

Abbildung 3.5: ODP Referenzen (WWW-Ansicht)

Anstelle einer nicht vorverarbeiteten Quelle wie dem ODP kann ein Internetverzeichnis auch Ausgangsbasis für verschiedene Zwischenschritte sein, die der Verbesserung der Datenqualität dienen sollen. Einige mögliche Verfahren sollen an dieser Stelle exemplarisch genannt werden, obwohl die Einbindung ihrer Ergebnisse den Rahmen dieser Arbeit übersteigen würde:

Lafourcade beschreibt beispielsweise ein System zur automatischen Erstellung hierarchischer Taxonomien (vgl. [Lafourcade, 2002]). Eine solche Taxonomie könnte - bei vergleichbar hohem Datenvolumen - ein denkbarer Ersatz des ODP für den Einsatz im TSR-Ansatz sein, oder auch ergänzende Strukturen erstellen. So genannte "Konzeptvektoren" werden zur Repräsentation „semantischer Informationen“ genutzt, die hinter Wörtern oder Ausdrücken vermutet werden und von den Autoren als „Ideen“ bezeichnet werden. Aufgabe des Systems ist die lexikalische Disambiguierung und die thematische Analyse der Eingangstexte. Das zugrunde liegende Verfahren basiert auf der Rodget’schen These, daß „die Sprache durch eine Menge von Konzepten erzeugt“ werden kann (vgl. [Rodget, 1852], in: [Lafourcade, 2002]). Technisch handelt es sich um einen Bootstrapping-Prozess, der - beginnend mit Begriffen und Wörtern aus einem Wörterbuch - aus den Wortdefinitionen Vektoren erzeugt.

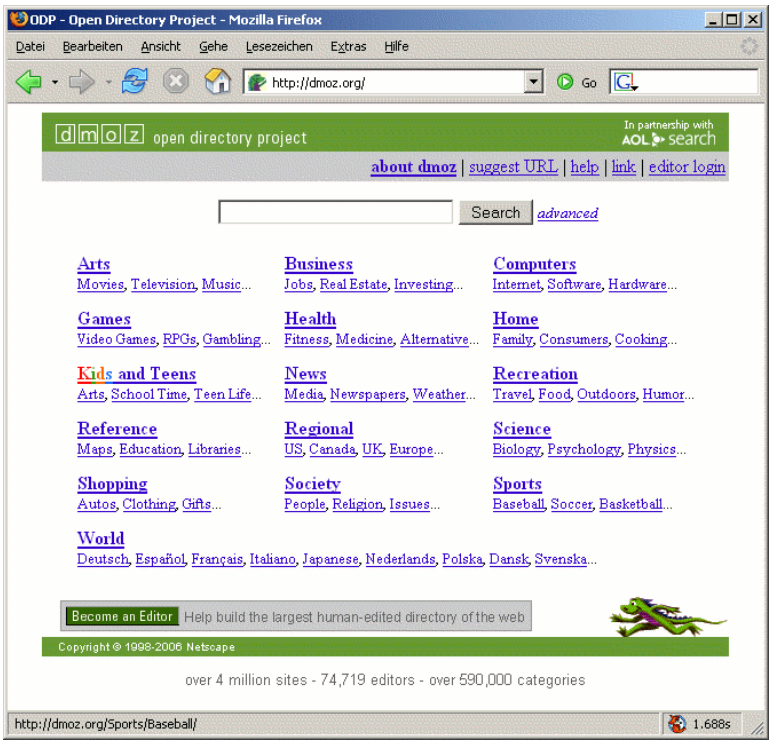

Abbildung 3.6: ODP oberste Ebene (WWW-Ansicht)

Beispielsweise wird für den Begriff "quotation" der folgende Vektor erzeugt:

$$v('quotation') = stockexchange[0,7], language[0,6], classification[0,52], ...$$

Zur Analyse des Ähnlichkeitsgrades zwischen Vektoren wird das Kosinusmaß verwendet; der Grad der thematischen Verwandtschaft wird darüber hinaus über den Arcuscosinus festgestellt. Kontextvektoren können auf verschiedene Weise miteinander kombiniert werden:

1. Normierte Summe: $V = X \otimes Y \qquad |v_i = \frac{(x_i + y_i)}{\|V\|}$
2. Normiertes Termprodukt: $V = X \otimes Y \qquad |v = \sqrt{x_i \cdot y_i}$
3. Konzeptualisierung: Ein Term selektiert Bedeutungen eines anderen Terms; d.h. er wirkt als "Kontext". Formal: $\gamma(X,Y) = X \oplus (X \otimes Y)$

Das Verfahren ist primär auf die Erstellung von Taxonomien ausgerichtet, und lediglich sekundär auf WSD. Leider werden keine Ergebnisse präsentiert, so daß ein direkter Performanzvergleich zu anderen Verfahren nicht möglich ist.

Ein semi-automatisches Verfahren zur Erstellung von Taxonomien wird von Widdows in [Widdows, 2003] beschrieben. Hierbei wird die Struktur der Taxonomie

von Hand erstellt und später einzelne Wörter durch eine Kombination von LSA auf Term-Dokument-Matrixdaten und PoS-Informationen in eine der taxonomischen Struktur entsprechende Nachbarschaftsbeziehung gesetzt. Das Ziel von Widdows Ansatz ist ebenfalls in erster Linie die Erstellung der Taxonomie. Wie bereits durch Lafourcade argumentiert, könnte aber auch eine syntaktische und thematische Disambiguierung einzelner Wörter erfolgen.

Der Bezug zur TSR-Methodik entsteht über eine offenbar ähnliche Auffassung der Textbedeutung: die von Widdows erstellten Taxonomien könnten in diesem Sinne ebenfalls die Rolle des Internetverzeichnisses als Datenquelle bei der Erstellung der Datenbasis für den TSR-Ansatz übernehmen, oder die Informationsdichte des tatsächlich genutzten Internetverzeichnisses erhöhen.

Inwiefern derartige ergänzenden Automatismen die späteren Ergebnisse des TSR-Ansatzes beeinflussen würden, soll hier nicht konkretisiert werden, prinzipiell würde jedoch die Aussagefähigkeit des Systems erhöht, da der Datenbasis mehr Wörter bekannt wären und eine feine Kategorisierung ermöglicht würde.

Die Ersetzung des Internetverzeichnisses durch eine gänzlich andersgeartete Datenbasis wird im folgenden Abschnitt erörtert.

3.4.3 WordNet als alternative Datenquelle

Weil die Nutzung der bereits erwähnten lexikalischen Datenbank Wordnet (und vergleichbaren Wissensbasen, wie z.B. FrameNet (vgl. [Baker et al., 1998])) besonders in der neueren Literatur stark verbreitet ist, soll in diesem Abschnitt eine kurze Diskussion darüber erfolgen, inwiefern WordNet-Daten die im TSR-Ansatz verwendeten ODP-Daten substituieren könnten.

WordNet dient vielen Verfahren als Ausgangs-Datenbasis, in ganz ähnlicher Weise, wie für den TSR-Ansatz das ODP genutzt wird. Zwar gibt es mittlerweile eine Anzahl ähnlicher Datenbasen, WordNet hat jedoch durch seine freie Verfügbarkeit und die breite Nutzung durch die wissenschaftliche Gemeinschaft eine bedeutende Rolle inne. WordNet soll an dieser Stelle auch in Vertretung für vergleichbare Datenbasen untersucht werden, da es sich hierbei um die größte und bekannteste Datenbank dieser Art handelt.

WordNet ist ein lexikalisch orientiertes Datenbanksystem, welches eine Vielzahl englischer Wörter (Nomen, Verben, Adjektive) mitsamt der Beschreibungen ihrer Bedeutungen enthält, sowie bestimmte syntaktische und semantische Relationen zwischen den Bedeutungsvarianten. Eine erste Beschreibung findet sich bei Fellbaum (vgl. [Al-Halimi et al., 1998]). Eine in WordNet beschriebene Bedeutungsvariante wird im Englischen als „Sense“ bezeichnet (Ein stilisiertes Beispiel hierzu wurde bereits auf Seite 48 vorgestellt). WordNet dient zudem als Repository für semantische Daten in SENSEVAL-3.

Innerhalb von WordNet entstehen durch die explizit gespeicherten semantischen und lexikalischen Beziehungen (Synonymie, Hypernymie, Antonymie u.a.) zwischen den Wörtern hierarchische Strukturen - dadurch sind hierauf basierende Methoden dem TSR-Ansatz oft in gewisser Weise ähnlich.

Ein gutes Beispiel hierfür ist das folgende Verfahren, in dem gewichtete und beschriftete Graphenstrukturen für WSD-Zwecke eingesetzt werden:

Canas et al. konstruieren aus WordNet-Daten Mengen von "Konzepten" (vgl. [Canas et al., 2003]). Diese Konzepte bilden die Knoten eines Graphen, wobei gewichtete Kanten über lexikalische WordNet-Relation definiert werden. Die Graphen werden als Repräsentation von Domänenwissen interpretiert und dienen zur Disambiguierung von Wörtern, indem die Gewichte von Pfaden innerhalb des Graphen miteinander kombiniert und verglichen werden. Die Autoren berichten eine 75%ige Erfolgsquote in einem WSD-Experiment, in dem allerdings nur 12 Exemplare verglichen und bewertet werden. Die durch Canas et al. erstellten Graphen könnten aufgrund ihrer strukturellen Ähnlichkeit ebenso wie die im vorhergehenden Abschnitt erstellten Taxonomien in gewisser Weise das im TSR-Ansatz verwendete Internetverzeichnis um weitere Strukturen und anders geartetes Hintergrundwissen (in diesem Falle: um lexikalische Zusammenhänge und modelliertes Domänenwissen) ergänzen oder ersetzen (dies gilt auch andersherum: TSR-Strukturen könnten ggf. die von Canas konstruierten Graphen ergänzen oder ersetzen)

Dieser Ansatz scheint dem von Agirre und Rigau (vgl. [Agirre & Rigau, 1996]) zu ähneln; wie dort auch werden graphische, gewichtete Strukturen aus einem Wörterbuch (WordNet) erstellt und zur Auflösung von Wortmehrdeutigkeiten verwendet, wobei sich die Algorithmen zur Ähnlichkeitsmessung allerdings unterscheiden.

Der graphische Ansatz, sowie die Verwendung des Ähnlichkeitsbegriffes stellen eine Vergleichbarkeit beider Verfahren zum TSR-Ansatz her. Durch die Verwendung von WordNet als Datenquelle der Konzepte erscheint die Datenbasis zwar konsistenter als ein Internetverzeichnis, allerdings auch im Umfang wesentlich eingeschränkter, da WordNet sehr viel kleiner als ein Internetverzeichnis (insbesondere das ODP) ist.

Ein eher indirekt auf den lexikalisch-syntaktischen Relationen WordNets basierender Ansatz wird durch Niu et al. beschrieben: Die Autoren verwenden den semi-überwachten graphischen Algorithmus "Label Propagation" (abgek. *LP*), in welchem sowohl beschriftete als auch unbeschriftete Wortexemplare einem Text entnommen und in einem Graphen als Knoten repräsentiert werden (vgl. [Niu et al., 2005]); die Beschriftung erfolgt hierbei durch die Auswahl einer WordNet--Bedeutungsvariante. Die Kanten des Graphen werden aus syntaktischen Relationen der Wörter im Text gebildet. Ziel des Verfahrens ist es, geeignete Bedeutungsvarianten für unbeschriftete Knoten aufgrund der Nachbarschaftsbeziehungen zu beschrifteten Knoten zu finden und infolgedessen die mit diesen Knoten verknüpften Wörter zu disambiguieren. Durch ein Experiment wurde festgestellt, daß der LP

Algorithmus für kleine Mengen von Trainingsdaten besser geeignet ist als SVM gestützte oder Bootstrapping-Verfahren. Für größere Mengen von Trainingsdaten zeigen dagegen nach Angabe der Autoren alle evaluierten Ansätze eine vergleichbare Leistung.

Im Zusammenhang mit dem TSR Verfahren ist dieser Ansatz – neben der wie oben beschrieben Ergänzungsmöglichkeit der TSR-Datenbasis – auch insofern interessant, als daß auch umgekehrt Daten aus dem TSR-Ansatz in die Eingangsdatenmenge einfließen könnten, um zusätzlich zu den verwendeten WordNet-Relationen über TSR-Funktionen weitere Nachbarschaftsbeziehungen zwischen Knoten sichtbar zu machen. Dies wäre ein gutes Beispiel für die Erweiterung eines existenten semantisch orientierten Verfahrens um TSRs als semantisch-pragmatische Komponente. Auch mittelbare Verfahren, wie das von Altintas et al. vorgestellte semantische Ähnlichkeitsmaß auf Basis von WordNet-Hierarchien (vgl. [Altintas & Coskun, 2005]), könnten in dieser Weise verwendet werden.

Nicht zuletzt soll darauf hingewiesen werden, daß sich zunehmend auch das internetbasierte Lexikon Wikipedia als mögliche Datenquelle präsentiert und in dieser Form auch – wenn derzeit auch nur selten – genutzt wird, z.B. durch Mihalcea (vgl. [Mihalcea, 2007]). Eine ausführliche Erörterung der diesbezüglichen Möglichkeiten würde aber den Rahmen dieses Kapitels übersteigen.

3.5 Analogien zur kognitiven Psychologie

Unabhängig von der, in den vorherigen Abschnitten vorgestellten, eher philosophisch motivierten Betrachtungsweise bestehen zudem Korrelationen zwischen dem TSR-Ansatz und bestimmten Methoden und Ansätzen der kognitiven Psychologie. Diesbezügliche mögliche Anknüpfungspunkte sollen daher in diesem Abschnitt diskutiert werden.

Die kognitive Psychologie ist ein interdisziplinär angelegtes Teilgebiet der Psychologie, welches mit der menschlichen Umsetzung von Wissen, Erkennen und Verstehen beschäftigt ist.

Die herrschende Meinung folgt hierbei der Theorie von John R. Anderson: das Gehirn wird hier als Komposition funktionaler Komponenten betrachtet, die zum Zwecke der menschlichen Kognition verknüpft agieren (vgl. [Anderson et al., 1977]).

Anderson beschreibt 1980 weiterhin seine empirisch unterlegten Überlegungen zum Wesen von „Konzepten“, „Propositionen“ und „Schemata“ als kognitive Einheiten (vgl. [Anderson, 1980]). Die diesbezüglichen Aspekten seiner ACT Theorie (einer modellorientierten Theorie der kognitiven Funktionsweise des menschlichen Gehirns, vgl. [Anderson, 1996]) sollen im Folgenden vorgestellt und mit ihren Entsprechungen im TSR-Ansatz verglichen werden.

Nach Anderson gibt es experimentelle Hinweise auf die Existenz von Konzepten, Propositionen und Schemata als kognitive Einheiten:

- *Konzepte* sind atomare, bedeutungstragende Einheiten
- Wohlgeformte Konfigurationen von Konzepten werden als *Propositionen* bezeichnet[4]. Propositionen repräsentieren die Bedeutung von Sätzen (vgl. [Anderson, 1980, S. 131].
- *Schemata* dienen zur Gruppierung von Propositionen. Sie können offenbar rekursiv auftreten (d.h. Schemata können weitere Schemata beinhalten) und somit eine Hierarchie bilden. Sie ermöglichen die Betrachtung gemeinsamer Merkmale der enthaltenen Propositionen auf übergeordneter Ebene (vgl. [Anderson, 1980, S. 158 ff]).

Das von Anderson angegebene Beispiel einer solchen Hierarchie ist in Abbildung 3.7 auf der nächsten Seite (vgl. [Anderson, 1980, S. 143]) wiedergegeben. Die Abbildung soll den mentalen Zugriffsweg auf das kognitive Konzept *lawyer* (Jurist) verdeutlichen: Anderson beschreibt, daß Versuchspersonen sehr schnell in der Lage seien, anhand vorgegebener Sätze einen begrifflichen Zusammenhang zwischen einem Rechtsanwalt und einem Restaurant herzustellen. Der Zusammenhang zwischen dem Rechtsanwalt und einer Zugfahrt würde dagegen aufgrund der ungewöhnlicheren Situation weniger schnell hergestellt. In der Abbildung werden nun z.B. die Propositionen, welche die Konzepte *enter* und *restaurant*, *order* und *hamburger* bzw. *pay* und *check* beinhalten, durch das Schema *restaurant* (Schemata sind jeweils durch Ellipsen dargestellt, Konzepte durch ihre begrifflichen Entsprechungen) gruppiert.

Die von Anderson beschriebene Schematik enthält den Kategorien der Internetverzeichnisse wesensverwandte Strukturen. Tatsächlich dienen ja auch Internetverzeichnisse der schnellen Auffindung von bedeutsamen Informationen durch menschliche Benutzer – es liegt daher nahe, eine der von Anderson skizzierten kognitiven Strukturen ähnliche Organisation innerhalb der Internetverzeichnisse (und damit auch innerhalb der TSRs) zu vermuten.

Analog zur Abbildung 3.7 ließe sich z.B. ein TSR für das Wort *lawyer* mit den Pfaden */ top / restaurant / enter*, */ top / restaurant / restaurant* etc. erstellen (Darstellung in Abbildung 3.8). Die Werte in den Klammern repräsentieren hierbei (zufällig gewählte) Beispielswerte für menschliche Reaktionszeiten, die in der Anderson'schen Methodik die „Nähe“ bzw. Relevanz der Konzepte im Schema darstellten.

Auffällig ist die Ähnlichkeit der Anderson'schen Schematik zur Struktur der TSRs. Tatsächlich besteht auch eine starke Analogie der Komponenten:

[4]Hierbei ist die Definition von „Wohlgeformtheit“ für den Vergleich zum TSR-Ansatz irrelevant und soll nicht vertieft werden.

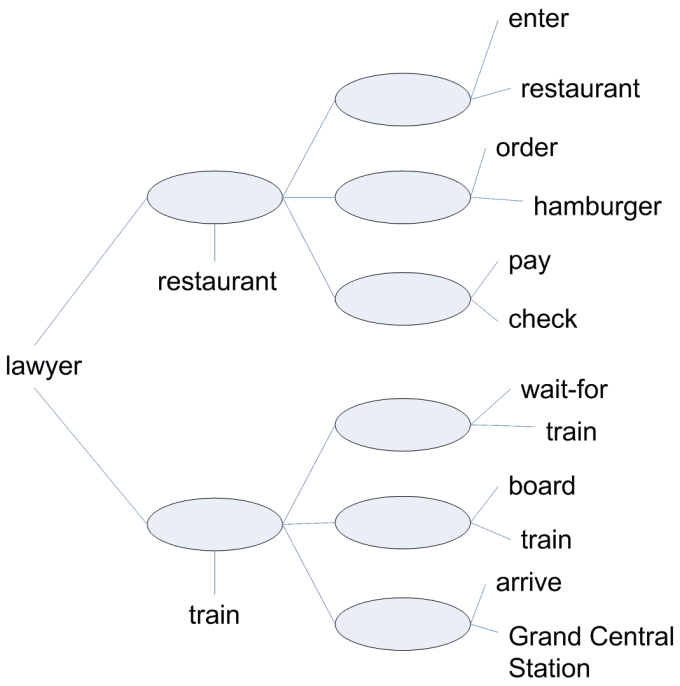

Abbildung 3.7: Hierarchie Schemata, Propositionen und Konzepte

- Die Schemata Andersons entsprechen den Kategorien der TSRs: beide dienen der Gruppierung relevanter Inhalte.
- Die Propositionen Andersons sind mentale Entsprechungen von (vollständigen) Sätzen. TSRs benötigen keine mentale Basis und arbeiten somit direkt auf den Sätzen. Damit beinhaltet die propositionale Ebene zwar eine zusätzliche Abstraktionsebene, dies führt jedoch zu keinem prinzipiellen Widerspruch.
- Eine ähnliche Analogie besteht zwischen Konzepten und TSR-Termen.
- Anderson untermauert seine Theorie zwar mit empirischen Ergebnissen, die von ihm gemessenen Reaktionszeiten menschlicher Subjekte fließen jedoch nicht direkt in sein Modell ein. Tatsächlich verknüpft er diese Werte aber mit der Assoziation von Konzepten zu bestimmten Schemata. Dies läßt sich als ein Hinweis auf eine Priorisierung der mentalen Verarbeitung interpretieren - mithin eine Repräsentation von Relevanz, die der Gewichtungsweise von TSR-Knoten im Wesen entspricht.

Dieser Argumentation folgend lassen sich TSRs als ein sinnvoller Ausdruck kognitiver Strukturen auffassen, wobei die Strukturierungsgrundlage wiederum auf der Nutzung der Sprache bzw. der Verwendung von Wörtern beruht. Die Annahme, daß

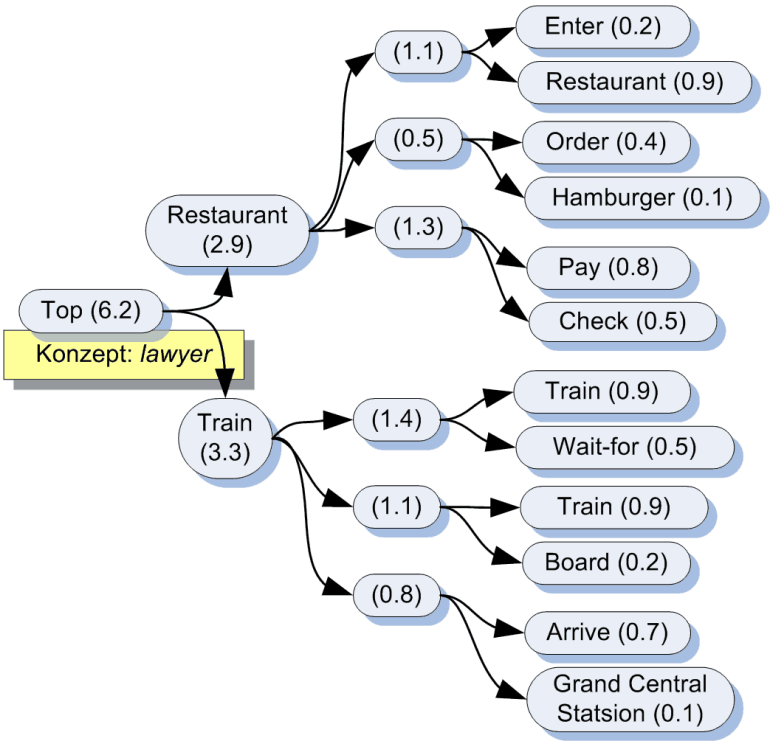

Abbildung 3.8: Andersons „lawyer“ Beispiel in TSR-Darstellung

semantisch-pragmatische Inhalte durch TSRs repräsentiert werden können, kann somit durch empirische Ergebnisse aus dem Bereich der kognitiven Psychologie gestützt werden.

3.6 Zwischenergebnis

Wie in diesem Kapitel dargelegt wurde, handelt es sich beim TSR-Ansatz um eine Methode der Repräsentation eines besonderen Aspektes der Textbedeutung, die sich über die Familienähnlichkeit von Wörtern, ihre Verwendung und über ihre Zuordnung über eine hierarchische Gliederung von pragmatisch orientierten Kategorien definiert.

Ausgangspunkt einer derartigen Auffassung von Textbedeutung sind die in Abschnitt 3.1 dargelegten Kritikpunkte der klassischen Merkmalssemantiken. Die TSR-Theorie ähnelt in vielen Punkten der Prototypensemantik: es besteht eine ähnliche Behandlung unscharfer Sprachphänomene und der Betrachtung der Wortbedeutung in Hinblick auf kategorische „semantische Zentren“, zudem soll auch die Prototypensemantik viele Probleme klassischer Merkmalssemantiken aufzulösen helfen (Mehrdeutigkeit, Abgrenzungsprobleme diskreter Kategorien, Gleich-

wertigkeit semantischer Merkmale, Eindimensionalität des Bedeutungsbegriffes, usw.).

Wichtig für die angestrebte praktische Umsetzung der TSR-Theorie ist die Verfügbarkeit von Hintergrundwissen, beispielsweise in Form von WWW-gestützten Daten (das WWW bietet als Korpus das größte verfügbare Datenvolumen). Obwohl in bisherigen Verfahren die Benutzung von Suchmaschinen üblich ist, scheinen Internetverzeichnisse besser für eine TSR-Praxis geeignet zu sein: Internetverzeichnisse dienen der strukturierten angeleiteten Suche durch manuell erstellte, hierarchisch nach Kategorien geordneten WWW-Inhalten. Sowohl die Funktion von Kategorien als auch deren strukturelle Eigenschaften sind dabei im Sinne der TSR-Theorie gut nutzbar. Eine alternative Datenbasis wäre die lexikalische Datenbank WordNet, die ebenso von vielen anderen Verfahren genutzt wird.

Die Interpretation der Objekte des TSR-Ansatzes, der TSRs, erfolgt also im Zusammenhang mit einem Internetverzeichnis: das Internetverzeichnis stellt die Menge der möglichen Sinnbezüge dar, die das Auftreten von Wörtern auf die Tatsachen der Welt abbildet. Die betrachtete Welt ist hierbei die Menge der im Internetverzeichnis gespeicherten Inhalte, mithin eines Teilbereiches der Inhalte des World Wide Web. TSRs repräsentieren demnach einen komplexen Ausschnitt der Welt, der durch den Sinn des jeweilig assoziierten Wortes gegeben ist. Diesen Umstand kann man als Repräsentation eines Aspektes der Wortbedeutung interpretieren.

Neben theoretischen philosophischen Bezügen bestehen auch gewisse Korrelationen zwischen dem TSR-Ansatz und bestimmten Methoden und Ansätzen der kognitiven Psychologie, speziell einigen Theorien von John R. Anderson: das Gehirn wird dort als Komposition funktionaler Komponenten (im Sinne kognitiver Einheiten) betrachtet – für die Anderson'schen „Konzepte“, „Propositionen“ und „Schemata“ lassen sich entsprechende Analogien im TSR-Ansatz finden.

Das folgende Kapitel soll die Hypothese, daß semantisch-pragmatische Inhalte durch TSRs abgebildet werden können (und das damit verbundene theoretische Gerüst) nun inhaltlich untermauern und den Vorschlag eines praktischen Ansatzes, sowie die Beschreibung einer prototypischen Realisierung liefern.

Kapitel 4

Realisierungsvorschlag

Gegenstand dieses Kapitels und Schwerpunkt der vorliegenden Arbeit ist der vom Verfasser entwickelte Vorschlag einer Repräsentation gebrauchsorientierter Aspekte der Textbedeutung, der durch die theoretischen Grundlagen des Kapitels 3 vorbereitete TSR-Ansatz[1].

Kernobjekte des TSR-Ansatzes sind die sogenannten TSR-Bäume oder einfach TSRs, die die TSR-Semantik nach Abschnitt 3.1 realisieren und auf der Datenbasis eines Internetverzeichnisses – wie in Abschnitt 3.4.2 beschrieben – basieren. Diese TSRs sind Inhalt des Abschnittes 4.1. Nach einer eher „technischen Beschreibung“ der TSRs in Abschnitt 4.1.1 wird in Abschnitt 4.1.2 ein Konstruktionsverfahren für die sog. „initialen“ TSRs angegeben (d.h. der aus den originalen ODP-Daten abgeleiteten TSRs).

Beim TSR-Ansatz ist aber nicht nur die Konstruktionsweise spezifischer Datenstrukturen von Interesse, sondern es werden ebenso bestimmte Funktionen und Operationen auf diesen Strukturen definiert, die für die eigentliche Nutzbarkeit des Ansatzes wesentlich sind. Die TSR-bezogenen Eigenschaften und Funktionalitäten werden daher in Abschnitt 4.1.3 auf Seite 107 behandelt.

Weil der TSR-Ansatz seine Eignung als Lösungsverfahren für die WSD-Problematik belegen soll, wurde zudem ein prototypisches System realisiert, welches in Abschnitt 4.2 auf Seite 133 erläutert wird.

Schließlich erfolgt in Abschnitt 4.3 auf Seite 137 eine Zusammenfassung in Form eines Zwischenergebnisses.

[1] Einzelne Inhalte dieses Kapitels wurden bereits im Rahmen des Second Workshop on Text Meaning and Interpretation vorgestellt, vgl. [Winnemöller, 2004]

4.1 Der TSR-Ansatz

TSRs sollen - wie bereits erläutert - über Zusammenhänge zwischen Wörtern und Textfragmenten, die auf dem Begriff der Familienähnlichkeit beruhen, und basierend auf dem Gebrauch dieser Wörter über eine Kategorisierungsweise Zwecken der Text- und Sprachanalyse dienen.

Durch den Vergleich von TSRs untereinander soll sich beispielsweise feststellen lassen, inwiefern sich zwei bestimmte Wörter gebrauchsähnlich sind - indem ermittelt wird, ob sie in ähnlichen Zusammenhängen verwendet werden (wobei die Ähnlichkeit der Zusammenhänge, wie auch deren thematische Ausrichtung über die erwähnte Kategorisierung bestimmt wird). Auch die Sprache, in welcher ein Wort verwendet wird sowie der Verbreitungsgrad eines Wortes innerhalb der Sprache sollte sich ermitteln lassen.

Über diese Unterscheidungsmöglichkeit lassen sich TSRs in vielen schwierigen Bereichen der automatischen Textverarbeitung einsetzen, in denen eine semantisch-pragmatische Analyse sinnvoll ist. Neben der in dieser Arbeit vertieften Problematik der Wortmehrdeutigkeiten gehören z.B. das Information Retrieval, die automatische Textkategorisierung und die semantische Textsuche zu möglichen Anwendungsgebieten.

4.1.1 Graphentheoretische Definition

Die Frage nach dem Wesen der TSRs kann unter vielen Gesichtspunkten betrachtet werden. In diesem Abschnitt soll zunächst eine Definition auf Basis der Graphentheorie und in Bezug auf die Datenbasis des verwendeten Internetverzeichnisses gegeben werden.

Um vorab – zum besseren Verständnis dieser eher formalen Zusammenhänge – einen schnellen Eindruck zu vermitteln, wie ein TSR aussieht, ist in Abbildung 4.1 auf Seite 98 ein hypothetisches (und stark vereinfachtes) Beispiel für das englische Wort *account* angegeben[2].

Aus der Sichtweise der Graphentheorie heraus betrachtet, lassen sich TSR als beschriftete, gewichtete, gerichtete und zusammenhängende Bäume $TSR(t) = \{V, E, f, l\}$ mit den in Tabelle 4.2 auf der nächsten Seite angeführten strukturellen Eigenschaften beschreiben.

Zudem gelten die folgenden Eigenschaften:

- TSRs können als zusammenhängende, endliche, ungewichtete *Teilbäume* (im Sinne eines Teilgraphen) des mithilfe des verwendeten Internetverzeich-

[2]Tatsächlich wurde ein TSR bereits in Abbildung 3.8 auf Seite 92 dargestellt, hier geht es jedoch um das Aufzeigen verschiedener Darstellungsmöglichkeiten von TSRs; es wird deshalb ein einfacheres (und daher geeigneteres) Beispiel eingeführt.

Symbol	Beschreibung
t	Einem TSR ist genau ein *Ausdruck der (natürlichen) Sprache,* ein *Term t* zugeordnet; dies kann ein einzelnes Wort sein, es kann sich aber prinzipiell um einen beliebigen Text bzw. Sequenz von Wörtern handeln. Als *Wörter* werden in diesem Zusammenhang Sequenzen alphanumerischer Zeichen (in verschiedenen sprachabhängigen Kodierungen) verstanden, die durch eine beliebige Anzahl von nicht-alphanumerischen Zeichen (z.B. Satzzeichen wie Punkte, Kommata, Leerzeichen, etc., aber auch Bindestriche, usw.) voneinander abgegrenzt werden. Damit sind sämtliche in der genutzten Datenbasis auftretenden Wortformen sprachunabhängig enthalten. - sowohl rein numerische Angaben (z.B. „007“ oder auch Jahresangaben, z.B. „1982“) als auch teilnumerische Angaben (z.B. „RFC1024“), nicht jedoch Begriffe wie z.B. „Nord-Süd-Gefälle“, diese werden in ihre Komponenten aufgespalten (d.h. aus den Wörtern „Nord“, „Süd“ und „Gefälle“ gebildet).
f_i, f_{TSR}	TSRs sind *gewichtet:* dies gilt sowohl für die einzelnen Knoten v_i (Knotengewichtungen f_i), als auch Gewichtung des Gesamtbaumes f_{TSR}). Die Gewichtung des Gesamtbaumes ist durch den Normfaktor gegeben, vgl. Abschnitt 4.1.2.
l_i	Die Knoten sind *beschriftet* durch die Bezeichnungen der Knoten des zugrunde gelegten Internetverzeichnisses (vgl. 4.1.2 auf Seite 101). Die Beschriftung kann in eine *Nummerierung* überführt werden (einige der folgenden Abbildungen zeigen daher die numerische Beschriftungsweise), um eine Überführung in eine Vektorrepräsentation ähnlich der Vektorraumdarstellungsweise von Dokumenten und Termen zu erleichtern, sofern diese gewünscht sein sollte (z.B. um den TSR Ansatz mit bestehenden Verfahren zu verknüpfen). Damit wird ein TSR aus diesen eher technischen Gründen nicht mitsamt der aus dem Internetverzeichnis entnommenen Beschriftung gespeichert, wie in Abbildung 4.1 dargestellt, sondern nummeriert, wie in Abbildung 4.2 dargestellt.

Tabelle 4.2: Eigenschaften von TSRs

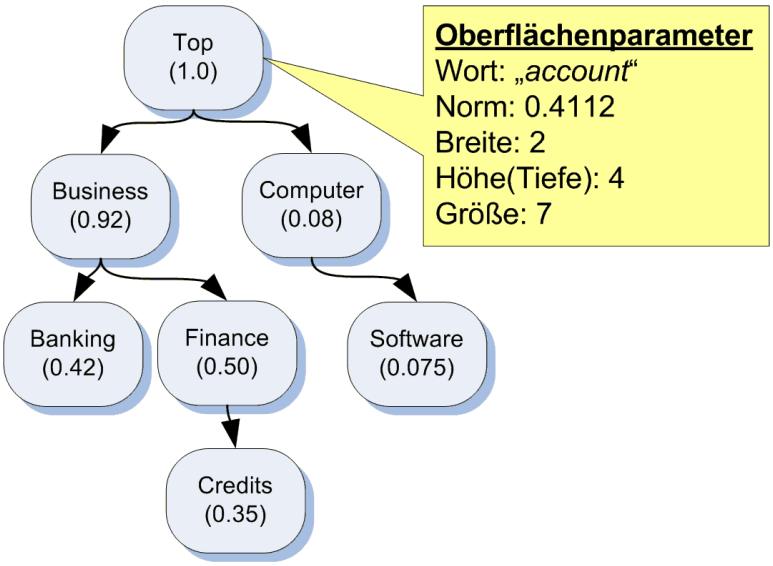

Abbildung 4.1: Ein TSR für den Begriff *account*

nisses konstruierten Verzeichnisbaumes (im weiteren Text synonym als „Maximalbaum“ bezeichnet) interpretiert werden, der die Gesamtmenge aller bekannten Pfade bzw. Knoten repräsentiert (vgl. Konstruktion von TSRs in Abschnitt 4.1.2). Die Eigenschaften des Zusammenhängens wie auch der Endlichkeit ergeben sich durch die Nähe zum Internetverzeichnis als Datenbasis, da dieses ebenfalls zusammenhängend und endlich ist.

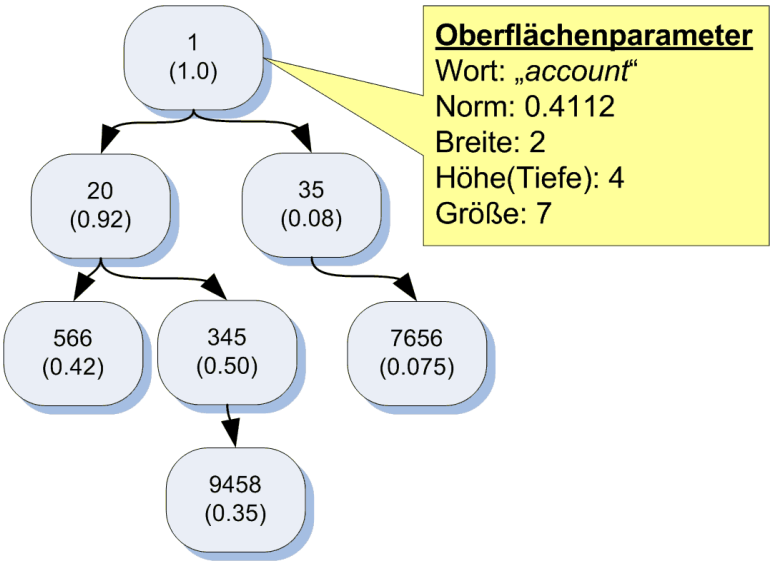

Abbildung 4.2: Nummerierter TSR für den Begriff account

- TSRs können aber ebenso als *Isomorphismus* der Struktur des Internetverzeichnisses verstanden werden, in welchem in der Regel die meisten Knoten den Wert 0 erhalten und in der Repräsentation nicht auftreten.

- Trennt man Struktur- und Gewichtungsinformation der TSRs, so erhält man

  1. einen zum TSR isomorphen (ungewichteten) Baum $S(t) = \{V, E, l\}$
  2. eine geordnete Menge $F(t) = \{V, f, l\}$ deren Elemente Paare aus natürlichen Zahlen (den Knotenbenennungen) und reellen Zahlen (den Knotengewichten) sind. Diese Menge F läßt sich als Indexvektor interpretieren, wobei die Nummerierung der TSR-Knoten nun als Indexnummer des Maximalvektors interpretiert werden, der aus dem Maximalbaum konstruiert werden kann (in diesem Falle wären die Indexnummern überflüssig, da die Beschriftung durchgehend und ohne Sprünge erfolgt). Dieser Indexvektor läßt sich, sofern strukturelle Änderungen auf den assoziierten ungewichteten Baum übertragen werden, als Grundlage der Berechnung des Ähnlichkeitsmaßes von TSRs, sowie der verschiedenen weiteren Operationen (Verschmelzung etc.) nutzen.

     Die entstehenden Strukturen sind in Abbildung 4.3 nach dem o.a. Beispiel wiedergegeben. Die Verbindung des Baumes S und des Vektors F bleibt bestehen, so daß sich strukturelle Veränderungen der einen Komponente in einer äquivalenten Änderung der anderen Komponente niederschlägt. Wird beispielsweise ein Wert in F auf Null gesetzt, also aus dem Indexvektor „gelöscht“, so wird auch der entsprechende Knoten in S gelöscht - und umgekehrt.

     Diese Erzeugung von Indexvektoren dient der Herstellung von Daten, die mit gängigen Systemen kompatibel sind, beispielsweise lassen sich Indexvektoren in vielen Machine Learning Systemen verwenden, z.B. in WEKA (vgl. [Witten & Frank, 2005]). Eine spätere Rekonstruktion des Baumcharakters ist in diesem Falle nur aufwändig über eine Analyse der Pfade über die Knotenreferenz möglich.

- Die TSR-„Oberflächenparameter“ Höhe, Breite, Anzahl der Knoten und Norm des Wurzelknotens V_0 besitzen eine eigene Semantik im Rahmen der TSR Methodik, die in Abschnitt 4.1.3 auf Seite 107 erläutert wird.

  In den in dieser Arbeit vorgestellten Experimenten wird der durch die Vektorisierung hervorgerufene Verzicht auf berechnete Oberflächenwerte in Kauf genommen, weil diese nur selten verwendet werden und der Performanzgewinn in Begriffen von Speicherplatz und Geschwindigkeit durch die Reduktion auf die Vektordarstellung erheblich ist (ein TSR in Vektordarstellung benötigt pro Element 8 Bytes für Element-ID und Elementwert plus einem Overhead für die verwendete Hashfunktion (1 Byte für je zwei Vektorelemente) – ein TSR in Baumdarstellung benötigt dagegen pro Knoten 8 Bytes für die Knoten-ID und den Knotenwert, sowie einen 32-Byte Zeiger auf

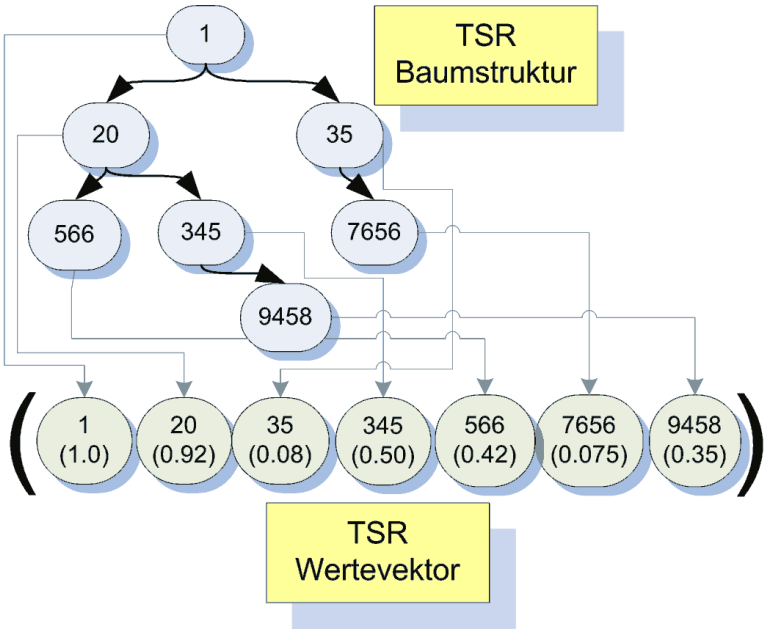

Abbildung 4.3: Beispiel TSR „account“ mit struktureller Trennung

die Knoten-Speicherstruktur). Ein schneller Überblick über die Basiseigenschaften eines TSRs ist dann aber nicht mehr möglich.

Die in diesem Abschnitte gegebenen Erläuterungen zur Struktur dienen als Grundlage der Darstellung des Konstruktionsverfahrens im folgenden Abschnitt 4.1.2 auf der nächsten Seite, sowie als Basis der Diskussion der funktionalen Eigenschaften von TSRs in Abschnitt 4.1.3 auf Seite 107. Zudem lassen sich schon aus ihrer Definition einige grundlegende Merkmale von TSRs ableiten:

- TSRs bieten keinen intrinsischen Inferenzmechanismus, und auch die Verarbeitungstiefe ist stark eingeschränkt (vgl. Abschnitt 3.1). TSRs entstammen nicht aus einer logisch-analytischen Sprachbetrachtung; sie bestimmten nicht die Wahrheit oder Falschheit von Sätzen bzw. Propositionen.

- Sie sind keine unmittelbare Repräsentation der Texte oder Dokumente selbst, d.h. die in ihnen enthaltene wesentliche Information ist nicht im Text bzw. Dokument enthalten und umgekehrt.

- TSRs sind nicht ontologieorientiert (vgl. Abschnitt 2.2.2.2 auf Seite 44) und eine Kommunikationssynchronisation über Rahmenkonstrukte o.ä. ist nicht angestrebt. Dasselbe gilt für musterbasierte Ansätze: TSRs könnten sicherlich über Platzhalter abstrahiert werden, dies ist aber nicht Gegenstand der aktuellen Forschung.

4.1.2 Konstruktion der initialen TSRs

Das Gesamtkonzept der TSRs beinhaltet nicht nur die Definition der eigentlichen Baumstrukturen, sondern auch verschiedene Operationen auf diesen Strukturen, insbesondere eine Methode zur Erstellung von „initialen“ TSR-Bäumen aus Internetverzeichnissen. Die Formulierung „initial“ soll hierbei ausdrücken, daß TSRs auch noch auf anderen Wegen - speziell durch Ableitungsverfahren - erzeugt werden können, hierfür jedoch bereits eine Basismenge von TSRs vorhanden sein muß.

Zur Konstruktion der TSRs werden die im Internetverzeichnis enthaltenen Wörter (mit Ausnahme bestimmter Wörter, u.a. sogenannte Stopwörter) verwendet – dies betrifft den gesamten Text der Einträge, nicht nur deren Titel o.ä.. Einige Beispiele hierzu finden sich in Abbildung 3.3 auf Seite 81.

TSRs sind, wie in Abschnitt 4.1.1 auf Seite 96 erläutert, beschriftete und gewichtete Bäume, die mit jeweils einem bestimmten Wort oder Textfragment assoziiert sind. Die Pfade der TSRs entsprechen den hierarchischen Kategorien des zugrunde gelegten Internetverzeichnisses; eine Bewertung erfolgt aus einer Berechnung der "Relevanz" der Wörter innerhalb der jeweiligen Kategorie - eine Denkweise, die dem Bereich des „Information Retrieval“ entlehnt ist (vgl. [van Rijsbergen, 1979]), die aber auch in der WSD-Forschung verbreitet ist. Ein mit einem bestimmten Wort assoziierter TSR besagt also zunächst, in welchen Kategorien des genutzten Internetverzeichnisses dieses Wort zu finden ist - und welcher Wert diesem Wort innerhalb der Kategorie zukommt.

Es stellt sich nun die Frage, inwiefern sich die Interpretation der Inhalte und Kategorien von Internetverzeichnissen auf die TSRs übertragen läßt: Jeder Pfad im TSR beschreibt über die Gewichtung des Pfades die Relevanz des TSR-Terms in einer Kategorie des zugrunde gelegten Internetverzeichnisses.

Abbildung 4.4 auf der nächsten Seite liefert ein Beispiel[3] für diesen Sachverhalt: im Bedeutungskontext *Movies* besitzt der Begriff *Herring* nur eine sehr geringe Relevanz, d.h. die Assoziativität zwischen *Movies* und *Herring* ist sehr schwach (absoluter Wert in der Klammer). Dagegen ist die Assoziativität zwischen *Herring* und *Fish* stark ausgeprägt, d.h. *Herring* besitzt für die Kategorie *Fish* eine starke Bedeutung – man könnte auch sagen: dieser Begriff ist dem prototypischen Zentrum der Kategorie offenbar recht nahe.

Die in Abschnitt 3.4 auf Seite 74 beschriebenen Datenstrukturen des Internetverzeichnisses werden zur Erstellung der initialen TSRs nach dem folgenden Verfahren transformiert; hierbei wird für jeden Knoten v_i des Verzeichnisbaumes

[3] Hierbei handelt es sich – zur Verdeutlichung – um den stark gekürzten Exzerpt eines realen Beispieles

(V,E,l,d) nach den Erläuterungen von Abschnitt 3.4.2.2 auf Seite 82 die Beschriftung l_i, sowie eine Beschreibung $d_i =< p_i, t_i, R_i, C_i >$ angenommen (eine Übersicht des Transformationsprozesses ist in Abbildung 4.5 auf der nächsten Seite dargestellt):

1. **Erstellung einer Wort-Pfad-Liste** L:

   Für jeden Knoten v_i des Internetverzeichnisses wird die gesamte Beschreibung d_i in einzelne Wörter $d_i \rightarrow \{w_0, \ldots, w_{|d_i|}\}$ (nach der Definition von Wörtern gemäß Abschnitt 4.1.1) zerlegt.

   Jedem Wort w_j (mit $0 \leq j \leq |d_i|$) wird die Anzahl seiner Instanzen (d.h. wie of das jeweilige Wort im Text von d_i vorkommt) als Gewichtung g_j zugeordnet, so daß eine Menge von Tupeln $< t_j, w_j >$ entsteht. Für alle Elemente dieser Menge wird nun in der Liste L ein wie folgt definierter Eintrag erzeugt: $l_j =< w_{ij}, p_i, g_{ij}, |d_i| >$.

   Beispielsweise wird für den in Abbildung 3.4 dargestellten Knoten der in Abbildung 4.6 auf Seite 104 angegebene Listenanfang erzeugt.

   Es ist hierbei möglich, aus Effizienzgründen bestimmte Elemente des Internetverzeichnisses miteinander horizontal oder vertikal zu verschmelzen. Dies führt zur Reduktion der Menge der zu speichernden Pfade und damit

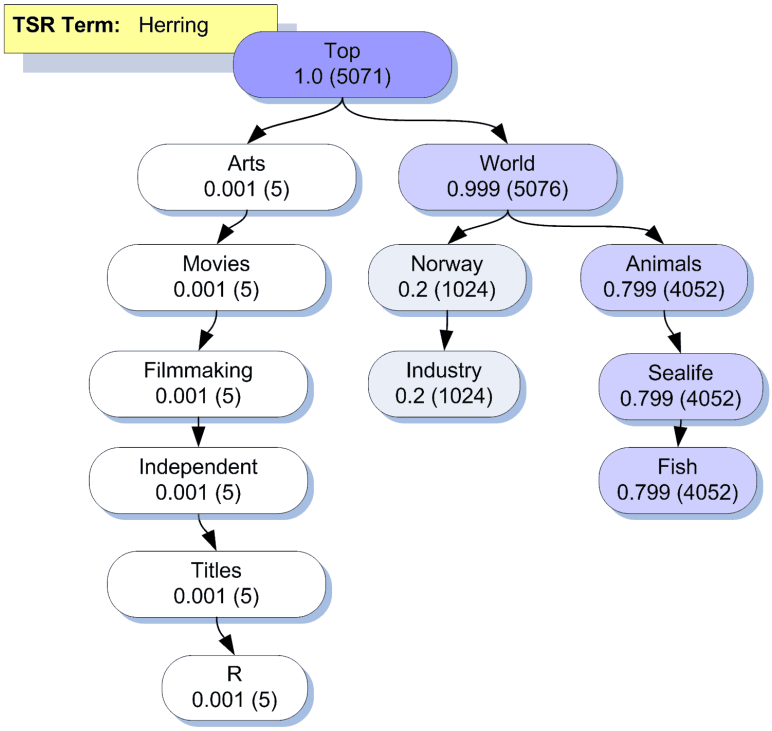

Abbildung 4.4: Hypothetisches TSR Beispiel *Herring*

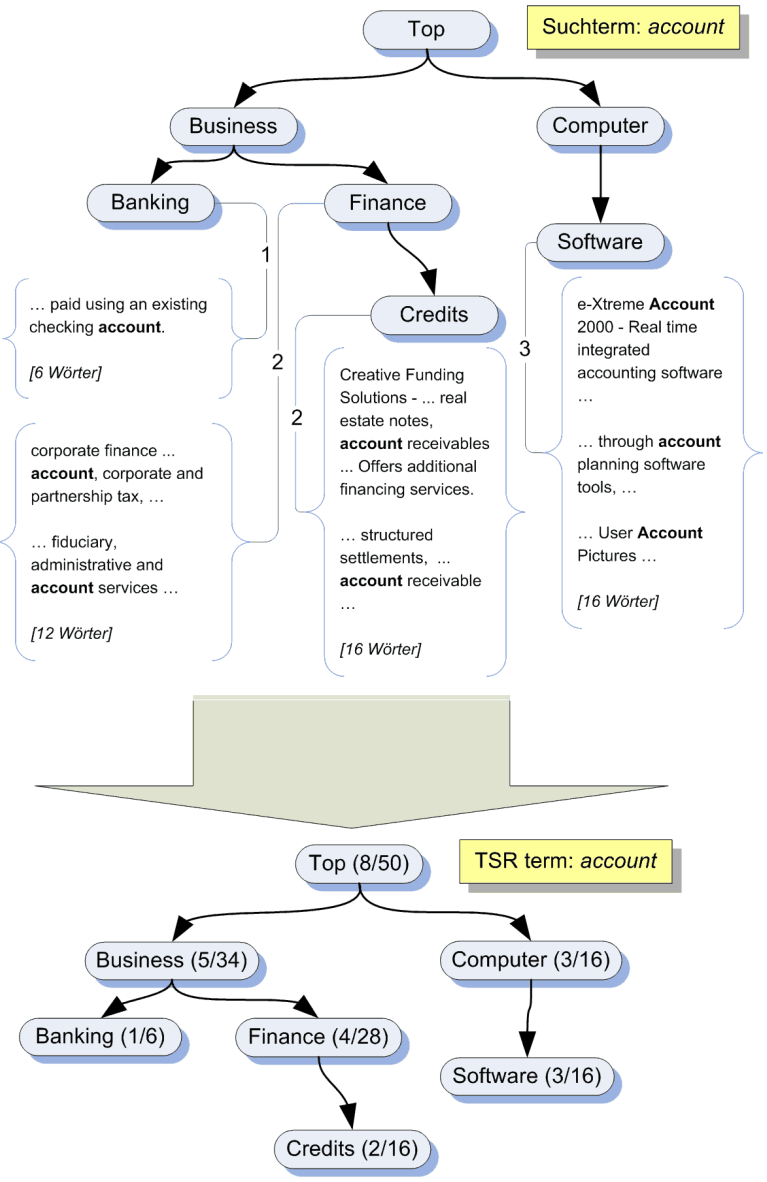

Abbildung 4.5: Übersicht / Beispiel der TSR Konstruktion

zur Reduktion der Größe der hiervon betroffenen TSRs. Die Anwendung dieses Verfahren ist gerechtfertigt, wenn keine semantisch-kategoriellen Informationen betroffen sind. Zum Beispiel sind die Unterelemente des Pfades */ Top / Arts / Movies / Titles* (vgl. Abbildung 4.6) in der technischen Realisierung zur besseren Übersichtlichkeit alphanumerisch gegliedert. Diese Ordnung kann aber ohne Verlust semantischer Information entfallen.

2. **Erzeugung von TSR-Bäumen** aus L:

Die im vorherigen Schritt erzeugte Wort-Pfad-Liste L wird nun in eine Menge von TSR-Bäumen überführt, indem jeweils alle l_k, die das Wort t (in der Instanz i) enthalten, zu einer Baumstruktur verschmolzen werden (ein hypothetischer Ausschnitt von L ist exemplarisch in Abbildung 4.7 dargestellt, zusätzlich zur o.a. List L ist noch die eindeutige Nummer des Pfades p_{t_i}angegeben).

Die Gewichtung der Knoten soll die Gewichtung der Terme in den einzelnen Baumknoten des verwendeten Internetverzeichnisses widerspiegeln. Sie folgt daher dem $TF \times IDF$-Maß, wobei nicht explizit in L repräsentierte Knoten interpoliert werden, indem ihr Gewicht aus der Summe der Gewichtungen ihrer Kindknoten gebildet wird - entsprechendes gilt auch für die Summe der Wörter aus d_{t_i}, $\sum|d_{t_i}|$. Abbildung 4.8 stellt die entsprechend aus dem Anfang von L entwickelte Liste $L*$ dar (Interpolationen sind mit dem Stern$*$gekennzeichnet). In dieser Abbildung ist zudem das Ergebnis der $g_{t_i} = TF \times IDF$-Berechnung enthalten, wobei für alle t_i durch Addition der Werte der Kindknoten gilt: $TF = w$ und $IDF = 1/(\sum|d_{t_i}|)$. .

w_{ij}	p_i	w_{ij}	$\|d_i\|$
10	Top/Arts/Movies/Titles/1/10_Rillington_Place	5	80
Rillington	Top/Arts/Movies/Titles/1/10_Rillington_Place	5	80
Place	Top/Arts/Movies/Titles/1/10_Rillington_Place	5	80
Film	Top/Arts/Movies/Titles/1/10_Rillington_Place	2	80
1971	Top/Arts/Movies/Titles/1/10_Rillington_Place	1	80
directed	Top/Arts/Movies/Titles/1/10_Rillington_Place	1	80
...	...		

Abbildung 4.6: Exzerpt Beispiel Wort-Pfad-Liste

t	p_{t_i}	$n(p_{t_i})$	w_{t_i}	$\|d_{t_i}\|$
Place	Top/Arts/Movies/Titles/1/10_Rillington_Place	56	5	80
Place	Top/Regional/UK	2345	9	98
Place	Top/Regional/Canada	67567	2	25
Place	Top/World/Visiting	34	1	12
Place	Top/Life/In-Places	3468	1	232
Place	Top/Computers/Conferences	983345	5	160
...	...			

Abbildung 4.7: Exzerpt Beispiel Wort-Pfad-Liste eines bestimmten Wortes

t	p_{t_i}	$n(p_{t_i})$	w_{t_i}	$\lvert d_{t_i}\rvert$	$\sum \lvert d_{t_i}\rvert$	g_{t_i}
∗Place	Top	0	17	5	652	0,0260
∗Place	Top/Arts	1	5	23	115	0,0434
∗Place	Top/Arts/Movies	3	5	2	92	0,0543
∗Place	Top/Arts/Movies/Titles	20	5	10	90	0,0555
∗Place	Top/.../Titles/1/	21	5	0	80	0,0625
Place	Top/.../1/10_Rillington_Place	56	5	80	80	0,0625
∗Place	Top/Regional	1998	11	58	181	0,0607
Place	Top/Regional/UK	2345	9	98	98	0,0918
Place	Top/Regional/Canada	67567	2	25	25	0,08
∗Place	Top/World	33	1	344	356	0,0028
Place	Top/World/Visiting	34	1	12	12	0,083
...	...					

Abbildung 4.8: Wort-Pfad-Liste L mit Interpolationen ($*$) und Gewichtungen

3. **Normalisierung der TSR-Knotenwerte g_{t_i}, Ableitung der TSR-Norm:**
   Durch den in Abbildung 4.9 vorgestellten Normalisierungsprozess soll die

```
procedure TSR.NORMALIZE(TSRa)
    max = 0
    for all v_a ∈ V_a do                    ▷ Iteration über alle Elemente aus TSR a
        x_a = f(v_a)
        if x_A > max then
            max = x_A
        end if
    end for
    SETWEIGHT(a, max)                       ▷ Setzt Gewicht von a auf max
    for all v_a ∈ V_a do
        x_a = f(v_a)
        y_a = x_a / max
        PUT(a, v_a, y_a)                    ▷ Überschreibe v_a mit Wert y_a in a
    end for
end procedure
```

Abbildung 4.9: Normalisierungsprozedur

Richtung der Relation zwischen Wort und Kategorie umgekehrt, d.h. die Bedeutung der Kategorie für das Wort festgestellt werden (Die genannte Relation wurde bereits auf Seite 101 anhand des Fish/Herring-Beispieles erläutert – dieses Beispiel wird zur Verdeutlichung hier wieder aufgegriffen): offenbar besitzt auch die Kategorie *Fish* für das Wort *Herring* eine starke Zentralität,

d.h. die Kategorie *Fish* und der Begriff *Herring* haben viele Merkmale gemein und treten vergleichsweise häufig in einem gemeinsamen Kontext auf, während die Kategorie *Movies* beispielsweise nur periphere Merkmale mit dem Begriff *Herring* teilt – nämlich die Tatsache, daß dieses Wort zum Titel eines Filmes gehört. Diese scheinbare Zufälligkeit tritt aber nicht störend in Erscheinung, sondern – im Gegenteil – verdeutlicht einen wichtigen Vorteil des TSR-Ansatzes, der eben in seiner pragmatischen Ausrichtung begründet ist: Viele Wörter können sehr schwache, subjektiv motivierte Assoziationen auslösen, die im alltäglichen Sprachverhalten eine Rolle spielen, durch eine konventionelle semantische Analyse üblicherweise aber nicht entdeckt werden. Über einen Schwellwertmechanismus (siehe Abschnitt 4.1.3.3) kann die diesbezügliche Auflösung der TSRs gesteuert werden (sehr schwache Assoziationen, d.h. solche, die unter einen bestimmtem Schwellwert liegen, können u.U. einfach gelöscht werden) und ein Verfahren somit gezielt auf nahe liegende oder weniger nahe liegende Verbindungen angesetzt werden.

Aus dem Konstruktionsverfahren wird ersichtlich, daß alle initialen TSRs durch Struktur und Inhalte des zugrunde gelegten Internetverzeichnisses vorgegeben sind. Die Menge der möglichen TSRs wird durch die Struktur des Internetverzeichnisses begrenzt - sie ist aber durch die Kombinatorik der Knoten außerordentlich hoch, und es kann erwartet werden, daß alle bekannten Wörter sich durch TSRs eindeutig darstellen lassen, d.h. derart, daß nahezu jedem Wort genau ein TSR entspricht – und umgekehrt. Durch die Endlichkeit der Menge der möglichen TSRs ist aber die Voraussetzung einer „closed world assumption“ gegeben, und es lassen sich, vergleichbar dem Prinzip der Widdow'schen Quantenlogik (vgl. [Widdows & Peters, 2003]), logische Junktoren über den TSRs angeben (auch die - in dieser Arbeit nicht definierte - Negation).

Abschließend sei festzustellen, daß die englische Sprache im Internet bei weitem dominierend ist, obgleich das ODP Internetverzeichnis Bereiche enthält, die in mehr als 75 verschiedenen Sprachen verfaßt sind. Diese Bereiche können wie folgt unterschieden werden: während englischsprachliche Inhalte direkt unter dem Wurzelknoten abgelegt sind, besitzen die anderen Sprachen ihre jeweils eigene Hierarchie unter dem Pfad */ Top / World /* $<$ *Sprache* $>$ – deutsche Inhalte lassen sich also in den Nachfolgerknoten von */ Top / World / Deutsch* finden[4]. Die im weiteren Text dieser Arbeit angegebenen Experimente und Ergebnisse legen aus Gründen der besseren Vergleichbarkeit zur Mehrheit von Ergebnissen anderer Autoren eine primäre Verwendung der englischen Sprache zugrunde. Nichtsdestoweniger ist der in diesem Kapitel beschriebene Ansatz sprachlich unabhängig und sein Erfolg hängt lediglich von der Menge der im ODP gespeicherten Inhalte einer bestimmten Sprache ab.

[4]Die Beschreibung der */ Top / World* -Kategorie lautet im Original: „*This category contains the non-English language versions of the Open Directory Project*“

4.1.3 Funktionale Eigenschaften von TSRs

Bisher wurde erläutert, welche strukturellen Eigenschaften den TSRs zugrunde liegen. In diesem Abschnitt soll dargestellt werden, welche Oberflächeneigenschaften TSRs zeigen, welche Vergleichsmöglichkeiten sie besitzen und welche Operationen auf ihnen definiert sein können.

Hierbei gilt, daß die vorgestellten Methoden lediglich aus Zeitgründen in ihrer Anzahl begrenzt sind: vorstellbar sind viele weitere Methoden und Funktionen, die jedoch Gegenstand zukünftiger Forschung sind.

4.1.3.1 Oberflächeneigenschaften

Bereits die Oberflächeneigenschaften eines TSRs – Baumtiefe, Breite, Norm – lassen einige (sehr grobe) Schlüsse auf den feststellbaren Gebrauch des mit dem TSR assoziierten Wortes oder Textfragmentes zu, allerdings besitzen die in den unten folgenden Punkten erläuterten Oberflächeneigenschaften der TSRs eine eher unzuverlässige semantische Aussagefähigkeit: häufig trifft die semantische Interpretation dieser Eigenschaften zu, manchmal allerdings nicht – in diesen Fällen liegen häufig besondere Umstände vor, die u.a. mit der Verwendung des WWW als Ausgangsdatenbasis zusammenhängen (Zum Beispiel besitzen computerbezogene Wörter eine deutlich höhere Gewichtung als in der „Alltagssprache“ und weisen demzufolge auch unerwartet hohe Oberflächenwerte auf).

Um die verschiedenen Oberflächeneigenschaften inhaltlich zu untermauern, wurde folgendes Experiment durchgeführt: es wurden 3 Korpora[5] mit jeweils ca. 700 Wörtern, extrahiert aus 3 Wörterbüchern erstellt. Bei diesen handelt es sich um einfache Wortlisten, in denen Wörter enthalten sind, die einer bestimmten Wissensdomäne zugeordnet werden können. Aus diesen Wortlisten wurden jeweils die ersten 700 Wörter herauskopiert und – aus technischen Gründen – in das SENSEVAL-2 Format überführt.

Die jeweiligen Eigenschaften dieser Wortzusammenstellungen werden in den Tabellen 4.3 und 4.4 aufgeführt (Zusammengesetzte Wörter sind in keinem Korpus enthalten).

[5] Wortlisten wie die hier referenzierten können als Korpora aufgefaßt werden, wenn man der Definition von Kilgariff und Grefenstette folgt und „Korpus“ als „Zusammenstellung von Texten als Objekt der Sprach- oder Literaturforschung“ definiert (vgl. [Kilgarriff & Grefenstette, 2003]).

Name	Beschreibung
Basic	**Dieses Korpus enthält ca. 700 Wörter eines umgangssprachlichen englischen Wörterbuches. Es sind Akronyme und Abkürzungen vorhanden, ansonsten nur Substantivformen von Wörtern.**
Med	**In diesem Korpus wurden ca. 700 Wörter eines auf ein medizinisches Umfeld spezialisierten Wörterbuches verwendet. Auch diese Zusammenstellung enthält nur Substantive.**
Comp	**Das Comp-Korpus enthält ca. 700 Wörter eines Wörterbuches aus der Domäne Computer- und IT-Begriffe**

Tabelle 4.3: Beschreibungen der Korpora

Korpus/Maß	SIZE	WEIGHT	DEPTH	WIDTH	DEPTH × WIDTH	DEPTH / WIDTH
„Basic“						
Arithm. Mittel	2.225	4.402	11	11	121	1
Median	430	268	11	12	132	0,92
Mittl. Abweichung	2.884	7.020	1	4	4	0,25
„Comp“						
Arithm. Mittel	745	1.031	9	7	63	1,29
Median	87	47	9	6	54	1,5
Mittl. Abweichung	1.043	1.616	2	4	8	0,5
„Med“						
Arithm. Mittel	580	923	8	5	40	1,6
Median	35	18	8	3	24	2,67
Mittl. Abweichung	919	1.593	2	4	8	0,5

Tabelle 4.4: Statistische Eigenschaften der Korpora

Tabelle 4.4 beinhaltet eine Übersicht der TSR-erzeugten Oberflächeneigenschaften der drei Korpora (Die enthaltenen Werte werden im Einzelnen in den folgenden Unterabschnitten erläutert und diskutiert werden). Die Tabelle ist wie folgt zu lesen:

- Spalte *Korpus/Maß* enthält den Namen der jeweils in den folgenden Zeilen beschriebenen Wortzusammenstellung, sowie die auf die Oberflächeneigen-

schaften angewandte statistische Funktion

- Die Spalten *Size*, *Weight*, *Depth* und *Width* enthalten die durchschnittlichen (nach dem arithmetischen Mittel bzw. Median) jeweiligen Oberflächenwerte der aus den einzelnen Korpora konstruierten per-Wort-TSRs bzw. deren mittlere Abweichung vom Mittel.

Im folgenden Text werden Hypothesen über die Aussagefähigkeit einzelner Maße diskutiert.

Funktionen WEIGHT und SIZE Die Funktion SIZE (Größe) ergibt die Anzahl der Knoten der Baumstruktur (deren Wert von Null verschieden ist) bzw. - aus „semantischer“ Sicht - die Anzahl der Bedeutungsfragmente: $SIZE = |V|$. Eine intuitive Interpretation besagt, daß dieser Wert als Maß der „Vielfalt“, der begrifflichen Komplexität des Ausdrucks in der Sprache verstanden werden kann, d.h. der Wert gibt an in wie vielen Sprachsituationen dieser Ausdruck verwendet werden kann. Es ist anzunehmen, daß Wörter mit einem kleinen Wert für SIZE in der Regel nicht so häufig verwendet werden, wie Wörter mit einem hohen Wert. Umgangssprachlich läßt sich dies im technischen Sinne etwa so formulieren: „je mehr Knoten ein Baum hat, in desto mehr thematischen Kontexten kann der Ausdruck üblicherweise sinnvoll eingesetzt werden“.

Eine weitere wichtige Aussage kann gewonnen werden, indem – ausgehend von der initialen Erstellung der TSRs – über die Funktion WEIGHT (Gewicht) die *Norm x* des TSR Wertevektors (vgl. Definition auf Seite 37) dazu verwendet wird, eine „Gewichtung“ des Wortes in der Hierarchie des Internetverzeichnisses festzustellen, ähnlich der Aussage des $TF \times IDF$-Gewichtes für Vektorraummodelle (vgl. Abschnitt 2.2.2.1 auf Seite 40). Vor der Normierung besteht der Sachverhalt $f(v_0) = x \in \mathbb{R}$, wobei x den Wert des Wurzelknotens bezeichnet.

Unterstützt werden diese Hypothesen durch die Angaben der Tabellen 4.5 und 4.6 in denen im Mittel deutlich höhere Size und Weight Werte für das „generische“ *Basic*-Korpus auftreten (für welches also die größte sprachliche Vielfalt angenommen werden kann) als für die anderen beiden Korpora, wobei das im Internet stärker repräsentierte *Comp*-Korpus ebenfalls höhere Werte aufweist als das recht spezifische *Med*-Korpus. Der große Unterschied zwischen Median und arithmetischem Mittel läßt sich durch die sehr großen Unterschiede zwischen der Mehrheit der Werte und den Extremwerten erklären, eine Erklärung, die durch die erhebliche mittlere Abweichung weiter gestützt wird.

Funktionen DEPTH und WIDTH Die Funktion DEPTH beschreibt die *Tiefe* der TSR-Baumstruktur (vgl. Abschnitt 3.4.1 auf Seite 77). Die Funktion WIDTH liefert dagegen die *Ordnung* des TSR-Baumes, d.h. die maximale Anzahl der Baumknoten auf einer Ebene des TSRs (vgl. Abschnitt 3.4.1 auf Seite 75).

Man kann diese Funktionen semantisch so deuten, daß ein Ausdruck umso spezialisierter verwendet wird, je tiefer die Baumstruktur ist und daß ein Ausdruck sprachlich umso „breiter“ verwendet wird (im Sinne einer *domänenabhängigen* Verwendungsvielfalt[6]), je höher der WIDTH-Wert ist. Diese Interpretation scheint stichhaltig, da ein langer Pfad im zugrundeliegenden Internetverzeichnis auf einen hohen Spezialisierungsgrad der Inhalte hindeutet und eine hohe Anzahl von Geschwisterknoten auf eine vielfältige Gebrauchsfähigkeit des Begriffes auf der entsprechenden Ebene.

Gegen eine solche Deutung spricht zunächst, daß in Bezug auf ihren Gebrauch in der Umgangssprache auch recht spezielle Themen auf vergleichsweise niedrigen Ebenen anzufinden sind (z.B. */ Top / Health / Medicine / Reference / Online Databases / Medline* – Tiefe 6 – für die medizinische Datenbank „Medline“), während umgangssprachliche Themen manchmal vergleichsweise tief strukturiert sind (z.B. */ Top / Regional / Europe / United Kingdom / Recreation and Sports / Sports / Equestrian / HorseRacing* – Tiefe 8 – für englische Pferderennen). Außerdem finden sich im Internetverzeichnis ODP verschiedene Knoten, in denen eine künstliche „Breite“ herbeigeführt wird, um eine einfachere Webrepräsentation der Daten zu erreichen, z.B. durch alphabetische Einordnung der eigentlichen Unterkategorien wie unter */ Top / Arts / Movies / Titles / A ; / Top / Arts / Movies / Titles / B ... / Top / Arts / Movies / Titles / Z.* Dies beeinflußt natürlich nicht die tatsächliche semantische „Vielfalt“ der Inhalte.

Diese Umstände schränken die Aussagefähigkeit der Werte DEPTH und WIDTH erheblich ein, dennoch lassen sich gewisse unscharfe Aussagen treffen, besonders, wenn die Werte untereinander oder in Kombination mit SIZE oder WEIGHT verwendet werden. In den o.a. Beispielen besitzt */ Top / Health / Medicine / Reference / Online Databases / Medline* eine Ordnung von 5, während */ Top / Regional / Europe*

[6] Als „Domäne“ soll in diesem Zusammenhang ein bestimmter Anwendungsbezug in der Welt bezeichnet werden, der durch die betreffende Kategorie im verwendeten Internet Verzeichnis charakterisiert wird.

Maß	**Basic**	**Comp**	**Med**
Arithm. Mittel	4.402	1.031	923
Median	268	47	18
Mittlere Abweichung	7.020	1.616	1.593

Tabelle 4.5: Exzerpt Statistik für WEIGHT

Maß	**Basic**	**Comp**	**Med**
Arithm. Mittel	2.225	745	580
Median	430	87	35
Mittlere Abweichung	2.884	1.043	919

Tabelle 4.6: Exzerpt Statistik für SIZE

/ United Kingdom / Recreation and Sports / Sports / Equestrian / HorseRacing eine Ordnung von 10 aufweist, damit offensichtlich eine größere Vielfalt besitzt und man hieraus schließen kann, daß die Wissensdomäne „englische Pferderennen“ breiter angelegt ist als die medizinische Datenbank Medline – was die Deutung der multiplizierten Werte wäre (Pferderennen: $6 \times 10 = 60$ gegen Medline: $8 \times 5 = 40$). Um die teilweise erheblichen Größenunterschiede zwischen einzelnen TSRs auszugleichen, sollte die TSR-Größe, d.h. der Size-Wert, als Normalisierungsfaktor eingesetzt werden, d.h. in der Form $\frac{Depth \times Width}{Size}$. Die Wirkung dieser Formel auf die drei Korpora ist in Tabelle 4.7zu beobachten – das „allgemeinste“ Korpus weist hier die größten Mittelwerte auf, zeigt also den umfassendsten sprachlichen Gebrauch, gefolgt vom etwas weniger umfassend vertretenen Comp-Korpus und das recht spezielle Med-Korpus wird in der Gebrauchsfähigkeit am geringsten bewertet, was sich so interpretieren läßt, daß die Begriffe dieses Korpus die geringste sprachliche Verbreitung erfahren, auch innerhalb der Wissensdomäne selbst.

Maß	**Basic**	**Comp**	**Med**
Arithm. Mittel	8	6	5
Median	7	5	4
Mittlere Abweichung	3	3	3

Tabelle 4.7: Exzerpt Statistik für DEPTH*WIDTH/SIZE

Wie in Tabelle 4.8 zu sehen ist, sind die Mittelwerte für DEPTH für das spezifische *Med*-Korpus kleiner als für das generische *Basic*-Korpus (wobei das *Med*-Korpus jedoch die größere mittlere Abweichung aufweist). Die o.a. Interpretation des WIDTH-Wertes wird dagegen durch die Werte von Tabelle 4.9 gestützt.

Maß	**Basic**	**Comp**	**Med**
Arithm. Mittel	11	9	8
Median	11	9	8
Mittlere Abweichung	1	2	2

Tabelle 4.8: Exzerpt Statistik für DEPTH

Maß	**Basic**	**Comp**	**Med**
Arithm. Mittel	11	7	5
Median	12	6	3
Mittlere Abweichung	4	4	4

Tabelle 4.9: Exzerpt Statistik für WIDTH

Tatsächlich sind jedoch diese Werte, wie vermutet, nicht unabhängig voneinander zu betrachten: Spezialisiertheit und Allgemeinheit eines Ausdruckes hängen be-

grifflich als Gegensatzpaar stark zusammen. Mit hoher Wahrscheinlichkeit ist demnach eine kombinierte Betrachtungsweise zielführender: Tabelle 4.10 zeigt demnach eine „Flächenbetrachtung“ der kombinierten Werte, in Tabelle 4.11 werden die Werte in ein Quotientenverhältnis (Tiefe pro Breite) gesetzt. Die zu beobachtenden starken Werteabweichungen können damit erklärt werden, daß eine tiefere Struktur in der Regel auch eine größere Wahrscheinlichkeit für eine hohe Breite in einer bestimmten Baumtiefe besitzt und damit auch eine größere „Fläche“.

Der Quotientenwert liefert hierbei die Angabe, ob die TSR-Bäume in der Regel tiefer als breit sind oder umgekehrt, d.h. welcher der beiden Faktoren dominiert. Damit wird die eingangs angeführte semantische Sichtweise durch die Werte von Tabelle 4.11 gestützt, in welcher das durch seine Spezialisierungseigenschaft anzunehmend „tiefere“ *Med*-Korpus einen zur Breite relativ höheren Tiefen-Wert besitzt als das generische *Basic*-Korpus.

Maß	**Basic**	**Comp**	**Med**
Arithm. Mittel	121	63	40
Median	132	54	24
Mittlere Abweichung	4	8	8

Tabelle 4.10: Exzerpt Statistik für DEPTH $\times$ WIDTH

Maß	**Basic**	**Comp**	**Med**
Arithm. Mittel	1	1,29	1,6
Median	0,92	1,5	2,67
Mittlere Abweichung	0,25	0,5	0,5

Tabelle 4.11: Exzerpt Statistik für DEPTH / WIDTH

Funktion VALUE Die Funktion VALUE liefert den Wert eines einzelnen Knotens innerhalb des TSR-Baumes bzw. einer einzelnen Position innerhalb des Wertevektors. Weil jeder Knoten eines TSRs eine Kategorie des Internetverzeichnisses repräsentiert, entspricht die Wertigkeit eines Knotens im TSR-Baum der relativen Gewichtung dieser Kategorie gegenüber den anderen, im TSR vertretenen Kategorien. Wenn zum Beispiel ein TSR den (fiktiven) Knoten */ Top / World / Dansk*$= 0,5$ und den (ebenfalls fiktiven) Knoten */ Top / World / Deutsch*$= 0,1$ enthält, dann läßt sich dieser Sachverhalt so interpretieren, daß für den TSR der Knoten */ Top / World / Dansk* fünfmal so bedeutend ist. Die Funktion VALUE liefert somit die TSR-lokale Gewichtung des Knotens.

Das aus technischen Gründen (Geschwindigkeitsverbesserung) manchmal wünschenswerte Verfahren der Vektorisierung (vgl. Verfahren auf Seite 99), d.h. die Erstellung eines TSR-Indexvektors unter Verlust der Strukturinformation, erhält

die Knotenwerte und die gespeicherten Oberflächeneigenschaftswerte in einer statischen Weise. „Statisch“ bedeutet in diesem Zusammenhang: bei Zusammenführung mehrerer TSRs können die Oberflächeneigenschaften nicht mehr präzise berechnet werden, sondern müssen über Heuristiken ermittelt werden. Algorithmen für die AND- und OR-Operationen (vgl. Abschnitt 4.1.3.3 auf Seite 118) werden – ohne Angabe der Operationen selbst – in Abbildung 4.10 vorgestellt. Hierbei gilt für die Eingabeparameter: *weight* ist das Gewicht, $size_a = |V|_{TSR(a)}$, $size_b = |V|_{TSR(b)}$ und für die OR-Heuristik selbst: Die OR-Funktion wirkt additiv, die dargestellte Heuristik setzt eine Annahme vollständig disjunkter Eingabe-TSRs voraus. Möglich wäre z.B. auch die Annahme einer z.B. 75%-igen Disjunktheit (Ähnliches gilt entsprechend für die AND-Funktion).

function TSR.ORHEURISTIC($weight, size_a, size_b$)
    $weight = \frac{\sum f(v_i)}{|V|}$ ▷ Summe der Knotenwerte ▷ durch Anzahl der Knoten
    $size = size_a + size_b$
    return $weight, size$
end function
function TSR.ANDHEURISTIC($weight, size_a, size_b$)
    $weight = \frac{\sum f(v_i)}{|V|}$
    $size = (size_a + size_b)/2$
    return $weight, size$
end function

Abbildung 4.10: Heuristiken zur Abschätzung von Oberflächenwerten bei der Vektorisierung

Anwendungsbeispiel multilinguale Algorithmen. Um einen möglichen Anwendungsbezug aufzuzeigen, soll die Nutzbarkeit der VALUE-Funktion exemplarisch am Beispiel der Sprachenerkennung (engl. language detection) aufgezeigt werden.

Ein einfacher TSR-basierter Ansatz hierfür ist der Folgende: es ist zu vermuten, daß die in einem beliebigen Textfragment verwendete Sprache durch Wertevergleich der Kindknoten (*Deutsch*, *Dansk*, *Suomi*, etc.) von */ Top / World* über die VALUE-Funktion festgestellt werden kann. Hierbei ist ausschlaggebend, welche Gewichtung der durch den TSR repräsentierte Ausdruck sowohl in der englischen Sprache, als auch in den weiteren Sprachen besitzt. Ein entsprechendes Experiment wurde für die Erkennung der Sprachen deutsch (DE), niederländisch (NL), italienisch (IT) und dänisch (DK) durchgeführt. Hierbei wurden Texte der jeweiligen Sprachen aus ihrem entsprechenden Bereich in Wikipedia extrahiert und in eine

Listing 4.1: Korpus Experiment Spracherkennung

```
<lexelt item="german">
  <instance id="1">
    <answer instance="1" senseid="de"/>
      Okanagan Lake ist ein langer, schmaler
      Binnensee in der kanadischen
      Provinz British Columbia. Der rund 135 km
      lange und 4–5 km breite See
...
```

Listing 4.2: Datei für die Spracherkennung, Liste der Sprachen

```
<lexelt item="german">
  <sense id="de" synset="164393" gloss="german" />
  <sense id="nl" synset="123690" gloss="dutch" />
  <sense id="it" synset="130034" gloss="italian" />
  <sense id="dk" synset="154382" gloss="danish" />
  <sense id="fi" synset="207193" gloss="finnish" />
</lexelt>
...
```

Form überführt, die dem SENSEVAL-2 Datenschema entspricht (Dies liegt auch den weiteren Experimenten dieser Arbeit zugrunde). Beispielhaft ist zur Verdeutlichung in Abbildung 4.1 ein Exzerpt angegeben. Anschließend wurde über ein TSR-basiertes System und eine manuell erstellte „Sprachen"-Datei (vgl. Abbildung 4.2; das *synset*-Attribut bezeichnet hierbei die technische ID des TSR-Knotens, z.B. kann auf den */ Top / World / Deutsch* –Knoten über die ID 164393 zugegriffen werden) eine Zuordnung zwischen den Sprachen und den einzelnen Texten vorgenommen. Zur Auswertung dieser Klassifizierung wurde das SENSEVAL-Programm *scorer2* verwendet.

Das Experiment lieferte trotz seiner Einfachheit aussagefähige Ergebnisse – sowohl für einzelne Wörter, als auch für kurze Textpassagen: Tabelle 4.12 zeigt die Accuracy-Werte für die genannten Sprachen in den Granularitätsstufen Wort, Satz und Text. Die Anzahl der Test-Instanztexte sind für jedes Experiment angegeben und die Anzahl der sprachbezogenen Kategorien für jede Sprache. Die Ergebnisse des Experimentes liegen zwar deutlich unter dem Stand der Technik (ca. 99%, vgl. [Kranig, 2005]), dennoch ist das Experiment insofern erfolgreich, als daß die Frage, ob eine TSR-basierte Spracherkennung möglich ist, positiv beantwortet werden kann. Das TSR-Experiment zeigt außerdem deutliche Unterschiede in der Erkennungsleistung verschiedener Sprachen, die aber bisher nicht erforscht wurden.

In der Literatur werden in der Regel Unterschiede in der Nutzung bestimmter Bedeutungsvarianten in unterschiedlichen Sprachen zur multilingualen Auflösung von Mehrdeutigkeiten verwendet. Multilinguale Verfahren begründen sich daher

	Text	Sätze	Wörter	Anzahl Kategorien
DE	100 %	100 %	100 %	55.808
NL	16 %	42 %	43 %	11.244
IT	99 %	100 %	100 %	23.285
DK	91 %	94 %	94 %	5.035
Summe	76 %	83 %	83 %	
Anz. Instanzen	1498	1174	1173	

Tabelle 4.12: Erkennung der verwendeten Sprache in Texten

auf dem Vorteil, relevante Bedeutungsvarianten durch die Verfügbarkeit mehrsprachlicher bzw. paralleler Korpora feststellen zu können. Ein paralleles Korpus enthält eine Menge von Texten in einer spezifischen Sprache und deren jeweilige Übersetzungen in einer anderen Sprache. Hierdurch ist es häufig möglich, sprachliche Unterschiede zur Auflösung von Mehrdeutigkeiten zu nutzen - beispielsweise unterscheiden sich die möglichen deutschen Übersetzungen von engl. *chair* erheblich: als *Stuhl*, bzw. als *Vorsitzender*.

Ein früher multilingualer Algorithms, der auf einem parallelen Korpus basiert, wurde von Brown et al. vorgestellt. Die Autoren benutzen den in [Brown et al., 1991] vorgestellten "Flip-Flop" Algorithmus zum automatischen "Corpus Alignment", d.h. zum Ausrichten des einen Korpus an einem anderen Korpus, das zudem in einer anderen Sprache gehalten ist - "Ausrichten" bedeutet in diesem Zusammenhang, daß Wörter und Ausdrücke auf der einen Seite mit entsprechenden Wörtern und Ausdrücken auf deren anderen Seite verknüpft werden, deren Übersetzung sie somit darstellen. Die Effektivität des Ansatzes besteht darin, daß einige Wörter im Rahmen des Übersetzungsprozesses disambiguiert werden können, da entweder das Wort in der Zielsprache nicht mehrdeutig ist, oder der Übersetzungsprozess nur eine einzige Interpretation zuläßt. Diesem Ansatz werden von Dagan und Itai lexikalische Constraints zugefügt, um statistische Modelle für mehrere Sprachen zu erstellen und damit Wörter zu disambiguieren (vgl. [Dagan & Itai, 1994]).

Eine etwas robustere Möglichkeit, WSD über ein multilinguales Verfahren herbeizuführen, stellten Diab und Resnik 2001 vor: hierbei werden Gruppen ähnlicher Übersetzungen gebildet, die mit dem jeweiligen Zielwort korrespondieren (vgl. [Diab & Resnik, 2001]). WSD erfolgt hier über die Auswahl der geeignetsten Übersetzungsmöglichkeit innerhalb einer Gruppe.

Die Nutzung mehrsprachlicher bzw. paralleler Korpora erscheint viel versprechend, allerdings zeigt hier der von Banko und Brill erläuterte Mangel geeigneter Datensammlungen (vgl. [Banko & Brill, 2001]) eine starke Wirkung, da reichhaltige parallele Korpora noch viel seltener sind als große monolinguale Korpora. Aktuelle Ansätze streben daher auch danach, weitere Datenquellen zu integrieren, z.B. monolinguale Korpora oder Wissensbasen wie WordNet, ein Beispiel hierfür ist das 2007 vorgestellte NUS-PT System von Chan et al. (vgl. [Chan et al., 2007]).

Der TSR-Ansatz ist inhärent mehrsprachlich, da Internetverzeichnisse nicht notwendigerweise auf eine einzige Sprache beschränkt sind (Das ODP als Datengrundlage beinhaltet Inhalte in über 70 verschiedenen Sprachen und kann somit selbst als multilinguales – wenn auch nicht paralleles – Korpus betrachtet werden). Aus diesem Grund erscheint die Verfügbarkeit multilingualer Korpora nicht in jedem Fall notwendig, sofern TSRs eingesetzt werden können. Bereiche unterschiedlicher Sprachen im Internetverzeichnis könnten beispielsweise im Gebrauch der Wörter in den Sprachen untereinander verglichen werden, wobei diese Bereiche im Fall des ODP-Internetverzeichnisses als Kindknoten des */ Top / World*-Pfades vertreten sind.

4.1.3.2 Verwendung von Ähnlichkeitsmaßen

Zur praktischen Nutzung der TSRs ist es notwendig, Vergleiche zwischen ihnen durchzuführen, um ihre relativen semantischen Eigenschaften und mögliche Assoziationen zwischen Ausdrücken aufzeigen zu können. Hierbei stellt sich die Frage, welcher Art ein solcher Vergleich sein kann bzw. was bewirkt werden soll.

Ein vielversprechender Ansatz ist die Prüfung auf *strukturelle Ähnlichkeit*. Hierbei wird ein paarweiser Vergleich der Werte von Knoten in zwei verschiedenen TSRs mit Summation der Unterschiede in einem singulären Ergebniswert durchgeführt. Die Erwartungshaltung ist hierbei die folgende:

> *Je ähnlicher sich zwei TSRs sind, desto ähnlicher werden die durch sie beschriebenen Ausdrücke in der Sprache verwendet.*

Dies bedeutet nicht, daß sich *notwendigerweise* die *Bedeutung* der beschriebenen Ausdrücke ähnelt! Beispielsweise besitzen Synonyme zwar denselben Bedeutungsumfang, werden aber nur sehr selten im selben Satz bzw. Textfragment gleichzeitig verwendet. Das Ähnlichkeitsmaß spiegelt hierbei den durch die TSRs beschriebenen Aspekt der Textbedeutung über den Wortgebrauch wider. Aus diesem Grunde erhalten tatsächlich bedeutungsähnliche Wörter selten einen hohen „Ähnlichkeitswert“. Andererseits werden so Assoziationen zwischen Ausdrücken sichtbar: Sobald die Kookkurrenz von Ausdrücken eine gewisse Signifikanz erreicht, kann dies mittels der TSR Methodik festgestellt werden – beispielsweise erhält der Ausdruck *James Bond* eine höhere Gewichtung als der vergleichbar komplexe Ausdruck *James Dog* (James Bond: $0,0016$, James Dog: $0,00036$).

Technisch betrachtet werden die Knoten zweier TSRs paarweise strukturgerecht miteinander verglichen, d.h. es werden nur Knoten miteinander verglichen, deren Beschriftungen übereinstimmen, wie in Tabelle 4.13 auf der nächsten Seite exemplarisch aufgeführt wird (In den TSRs nicht explizit enthaltene Knoten werden als Knoten mit dem Wert 0 angenommen).

Da TSRs sich - wie oben gezeigt - als Assoziation einer Baumstruktur mit einem Indexvektor darstellen lassen, liegt eine Nutzung bekannter Meßverfahren aus dem

Kategorie TSR_a	$f(v_a)$	**Vergleich**	**Kategorie** TSR_b	$f(v_b)$
`/Top`	$1,0$	$\longleftrightarrow$	`/Top`	$1,0$
`/Top/Adult`	$0,5$	$\longleftrightarrow$	`/Top/Adult`	$0,02$
`/Top/Adult/Arts`	$0,2$	$\longleftrightarrow$	`/Top/Adult/Arts`	$0,001$
`---`	(0)	$\nleftrightarrow$	`/Top/Arts/Animation`	$0,78$
`/Top/Arts/Contests`	$0,004$	$\nleftrightarrow$	`---`	(0)
`/Top/World/Català`	$0,02$	$\nleftrightarrow$	`---`	(0)

Tabelle 4.13: Vergleich zweier TSRs, schematisches Beispiel

Bereich Information Retrieval nahe: hier hat sich, wie in Abschnitt 2.2.2.1 erläutert, das Kosinusmaß bewährt[7] (der entsprechende Algorithmus wird in Abbildung 4.11 vorgestellt). Weitere Ähnlichkeitsmaße aus der Literatur, z.B. *Maximum Entropy*

function TSR.COSINE($TSRa, TSRb$)
  $score = 0$
  for all $v_a \in V_a, v_b \in V_b : v_a = v_b$ **do**
    $x_a = f(v_a)$
    $x_b = f(v_b)$
    if $x_b ! = 0$ **then**
      $score+ = x_a * x_b$
    end if
  end for
  $normlength = |V_a| \cdot |V_b|$
  $result = \frac{score}{normlength}$
  return *result*
end function

Abbildung 4.11: Algorithmus für das Ähnlichkeitsmaß *Cosinus*

(vgl. [Vázquez et al., 2004]) können vermutlich in ähnlicher Weise eingesetzt werden.

Es gilt:

- Sehr *ähnliche* Ausdrücke liefern einen hohen Wert zurück.
- Orthogonale, d.h. wie oben erläutert *unähnliche* Ausdrücke liefern einen sehr kleinen bzw. einen Nullwert zurück. Dies wird so interpretiert, daß es in der dem System bekannten Welt keine Assoziation zwischen beiden Ausdrücken gibt.

[7] Eine experimenteller Nachweis der hier behaupteten Sachverhalte erfolgt in Kapitel 5

Das Kosinus-Maß prädestiniert sich durch seine Einfachheit für eine effiziente Verarbeitung. Die Literatur bietet aber einige interessante Alternativen, die an dieser Stelle genannt werden sollen:

Ein neues Maß der Ähnlichkeit zwischen zwei Wörtern, die *semantische Kompaktheit* (engl. *semantic compactness*), wird von Mavrodeidis et al. im Rahmen der Vorstellung ihres WSD-Ansatzes eingeführt (vgl. [Mavroeidis et al., 2005]). Die semantische Kompaktheit ist ein Maß, welches über sog. Steiner-Trees gebildet wird: es wird eine Matrix erstellt, in welcher einzelne Konzepte über die erwähnten Steiner-Trees miteinander in einen Nähebezug (engl. *proximity*) gesetzt werden, wobei die Konzepte Bedeutungsvarianten aus WordNet sind. Diese Matrix dient als Basis eines *semantischen Kernels* (engl. *semantic smoothing kernel*) welcher über eine Kernelmaschine (GSVM) genutzt wird. Verschiedene Experimente über die Reuters-, SENSEVAL-2 und SENSEVAL-3 Datensets belegen nach Angabe der Autoren eine maximale Precision von ca. 80% bei einem Recallwert von 25% (der geringe Recallwert wird von den Autoren mit einer geringen Abdeckung des allgemeinen Sprachumfanges durch WordNet begründet). Der Ansatz ist für TSR-Bezüge vor allem deswegen interessant, weil sich das Maß der semantischen Kompaktheit, sowie die Verwendung eines semantischen Kernels möglicherweise gewinnbringend auf TSRs übertragen lassen, z.B. als Substitution für den Einsatz der $TF \cdot IDF$-Formel im Rahmen des TSR-Konstruktionsverfahrens.

Die in der Literatur ebenfalls behandelten Tree-Edit-Distance Verfahren sind dagegen *nicht* sinnvoll einsetzbar, da diese Verfahren nicht von einer statischen Knotenstruktur ausgehen (z.B. sind die Beschriftungen der Knoten in den TSR-Bäumen invariabel) und zudem in der Regel eine zu hohe Rechenkomplexität aufweisen (häufig exponentiell) und daher im Rahmen des TSR-Ansatzes (in welchem oft Knotenzahlen > 10.000 auftreten können) kaum einsetzbar sind. Einige der Tree-Edit-Distance Verfahren werden durch Yang diskutiert; er schlägt in [Yang et al., 2005] die Tree Edit Distance als universelles Ähnlichkeitsmaß vor, wobei er im Rahmen des von ihm entwickelten Algorithmus ebenfalls die von ihm analysierten Bäume in numerische Vektoren überführt.

Ein einfacher Test auf Wert-Gleichheit der Knotenwerte ist schließlich nicht geeignet, da er lediglich zeigen kann, daß ein TSR mit sich selbst identisch ist (TSRs besitzen in der Regel eine eindeutige Struktur bzw. eine eindeutige Wertebelegung der Knoten).

4.1.3.3 Operationen

Bisher wurden lediglich wortbasierte TSRs erläutert, jedoch reicht der TSR-Ansatz über die Granularität der Wortebene hinaus - tatsächlich lassen sich TSRs für nahezu beliebige Textfragmente bilden (sofern zumindest einige der Komponenten dieser Fragmente in der TSR-Datenbank enthalten, also dem System „bekannt“ sind).

Über verschiedene Operationen lassen sich TSRs aus bekannten TSRs ableiten, wobei auch wiederum abgeleitete TSRs als Basis weiterer Ableitungen dienen können. Ziel der Ableitung kann ein Wechsel der Granularität sein (z.B. Wechsel von der Wortebene zur Satzebene), die Erstellung eines neuen TSRs für ein bisher unbekanntes Wort oder auch die Änderung eines gegebenen TSRs.

Hierbei bestehen viele Möglichkeiten, eine derartige Ableitung durchzuführen, da sich TSRs i.A. nicht nur strukturell unterscheiden, sondern auch in ihren Knotenwerten. Die allgemeine Graphentheorie bietet hierzu zwar einige Ansätze, diese erweisen sich aber als in diesem Fall unnötig komplex und zu allgemein. Stattdessen werden einige grundlegende, der allgemeinen Mengenlehre entlehnte Methoden eingeführt, die um die Berechnung von Gewichten erweitert wurden.

Schnitt Die Schnittoperation (im weiteren als AND-Operation bezeichnet) dient zum Abbilden eines semantischen Äquivalentes der Schnittmengenbildung. Damit wird die allgemeine Anwendbarkeit von Begriffen durch die Kombination mit anderen Wörtern eingeschränkt.

Das Wort *Hund* kann beispielsweise sowohl das bekannte Haustier bezeichnen, als auch ein Hilfsgerät zum Gütertransport. Der komplexe Ausdruck *Der kurzhaarige Hund* schränkt diese Deutungsfähigkeit auf die Haustierinterpretation ein.

Die AND-Operation kann wie folgt formalisiert werden: Gegeben seien zwei TSRs A und B, und es wird vom Indexvektormodell ausgegangen (d.h. alle TSRs besitzen eine einheitliche Größe $size(TSR) = N$, es sind lediglich die meisten Werte mit 0 belegt) dann gilt der Zusammenhang aus Abbildung 4.12. d.h. das Ergebnis der

```
function TSR.AND(TSR a, TSR b)
    TSRC                                          ▷ Leerer TSR
    sum = 0
    for all v_a ∈ V_a, v_b ∈ V_b, v_a = v_b do    ▷ Iteration über alle Vektorpositionen
        x_a = f(v_a)                              ▷ Wert aus Vektor a
        x_b = f(v_b)                              ▷ Wert aus Vektor b
        if x_a != 0 then
            x_a = min(x_a, x_b)
            sum += x_a
            C.setPath(v_a, x_a)                   ▷ Wertebelegung C
        end if
    end for
    return TSRC                                   ▷ Neuer TSR enthält Vektordaten
end function
```

Abbildung 4.12: Kernprozedur AND

AND-Verknüpfung zweier TSRs ist wieder ein TSR, in welchem jeder Wert durch

das Minimum der entsprechenden Indices beider Ausgangsvektoren belegt wird. Ist einer der beiden Werte gleich Null, entfällt er in der Indexschreibweise für Vektoren. Damit kann der entstehende TSR höchstens so viele Werte beinhalten, wie die beiden Ausgangs-TSRs gemein haben. Bei Orthogonalität der Ausgangs-TSRs, d.h. wenn beide TSRs keinen gemeinsamen Knoten mit einer von 0 verschiedenen Gewichtung besitzen, ist der Ergebnis-TSR leer.

Die graphische Darstellung eines simplifizierten Beispiels findet sich in Abbildung 4.13:

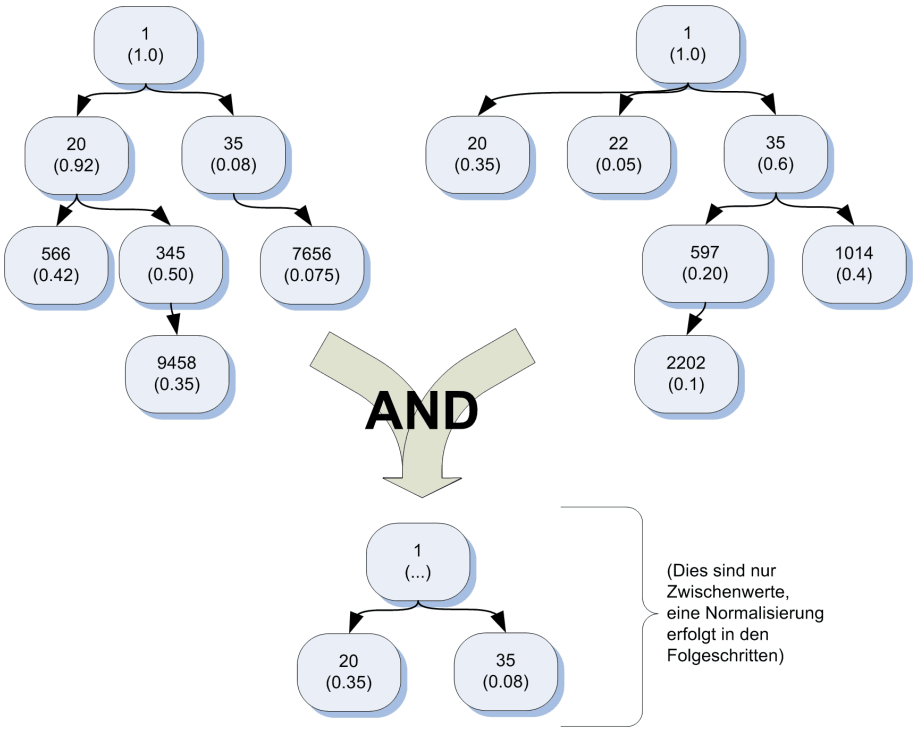

Abbildung 4.13: Beispiel Schnittbildung

Zusammenführung Die Zusammenführungs- (oder OR-) Operation für TSRs ist die zweite, wesentliche Operation und dient der Abbildung einer Vereinigungsmengenbildung. Hierdurch soll der Umstand verdeutlicht werden, daß die Verknüpfung von Ausdrücken unter Umständen eine eigene Bedeutung konstituieren kann.

Formalisiert können wir schreiben: Gegeben seien zwei TSRs A und B, dann gilt für die OR-Operation entsprechend der AND-Operation der in Abbildung 4.14 dargestellte Zusammenhang.

Demnach ist auch das Ergebnis der OR-Verknüpfung zweier TSRs wieder ein TSR, in welchem jeder Wert durch das Maximum der entsprechenden Indices in den

```
function TSR.OR(TSRa, TSRb)
    sum = 0
    for all v_a ∈ V_a, v_b ∈ V_b, v_a = v_b do        ▷ V_a : Knotenmenge von TSRa
        x_a = f(v_a)                                   ▷ Knotenwert von v_a
        x_b = f(v_b)                                   ▷ Knotenwert von v_b
        if x_a = 0 then          ▷ Wenn v_a nicht belegt ist, wähle Wert aus V_b
            sum+ = x_b
            SETPATH(V_b, v_b, x_b)
        else
            x_a = max(x_a, x_b)
            if WEIGHT(b)<WEIGHT(a) then
                x_a = x_b
            end if
            sum+ = x_a
            SETPATH(V_b, v_a, x_a)                     ▷ Setzen des Wertes aus A in B
        end if
    end for
    return b                                  ▷ TSRb enthält die verknüpften Knoten
end function
```

Abbildung 4.14: Kernprozedur OR

beiden Ausgangsvektoren belegt wird. Dies bringt mit sich, daß der entstehende TSR mindestens so viele Werte beinhaltet, wie der „größere“ Ausgangsvektor (d.h. derjenige mit den meisten von Null verschiedenen Werten) und maximal so viele Werte, wie beide Vektoren zusammen. Diese Umstände sind in Abbildung 4.15 zur Verdeutlichung dargestellt.

Beispielsweise können auch Definitionen neuer Wörter, ähnlich wie in Wörterbüchern, auf diese Art und Weise abgebildet werden. Untenstehend wird eine wörterbuchartige Definition des englischen Wortes „dog“ gegeben:

***dog**: a common animal with four legs, fur, and a tail.*

Gemäß der Semantik der OR-Operation kann ein TSR(*dog*) dann durch Zusammenführung von der TSRs für *common*, *animal*, *four*, *legs*, *fur* und *tail* erzeugt werden:

$$\begin{aligned} \textit{TSR(dog)} = \quad & \textit{TSR(}\text{common}\textit{)} \vee \textit{TSR(}\text{animal}\textit{)} \\ & \vee \textit{TSR(}\text{four}\textit{)} \vee \textit{TSR(}\text{legs}\textit{)} \\ & \vee \textit{TSR(}\text{fur}\textit{)} \vee \textit{TSR(}\text{tail}\textit{)} \end{aligned}$$

Äquivalent hierzu können wir sagen, daß sich der Sinn eines Satzes aus der Komposition seiner Komponenten ergibt, im Beispiel etwa:

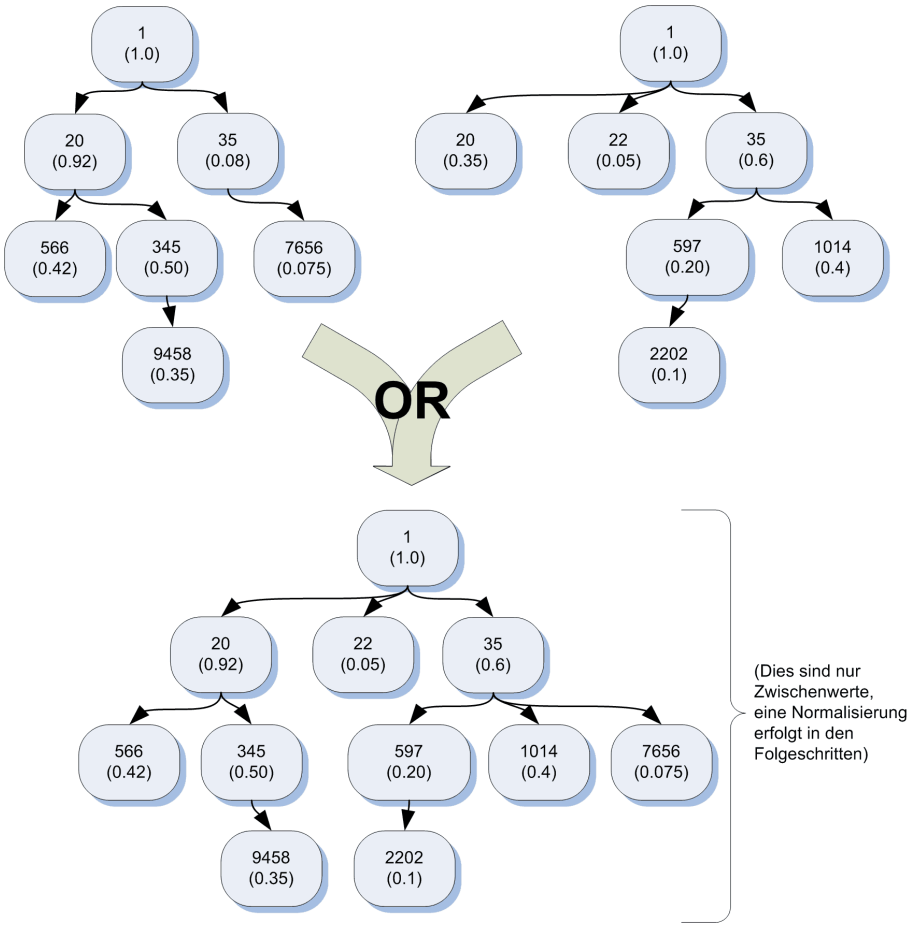

Abbildung 4.15: Beispiel Zusammenführung

$$\text{OR}(\ TSR(\textit{common animal four legs fur tail})\ ) = TSR(\text{common}) \vee TSR(\text{animal}) \vee TSR(\text{four}) \vee TSR(\text{legs}) \vee TSR(\text{fur}) \vee TSR(\text{tail})$$

Flache syntaktische Steuerung der TSR-Operationen Eine gezielte Anwendung der AND- und OR-Operationen anhand der vollständigen syntaktischen Struktur erfordert eine tiefe Vorverarbeitung des Textes.

Trivialerweise können aber auch sämtliche Wörter des Textes in einen Zusammenhang gebracht werden und per OR-Operation kombiniert werden. Diese Art der Kombination ist allerdings sehr grob und berücksichtigt nicht die vielfältigen Beziehungen der Wörter eines Textes untereinander, z.B. in Bezug auf ihr relatives Gewicht für die Bedeutung des Satzes in dem sie stehen. Außerdem ergibt sich ein Performanzproblem: die zu berechnenden TSRs können schnell sehr groß werden und ihre Berechnung erfordert dann sehr viel Rechenkapazität und Speicherplatz.

Das sogenannte *Online-Processing*, d.h. die aus Benutzersicht laufende Verarbeitung der Daten im Hintergrund, fällt dann schnell aus dem Rahmen der realistischen möglichen Anwendungsszenarien.

Eine Kompromißlösung liegt in der Anwendung einer *flachen* Vorverarbeitungsstufe: der zu verarbeitende Text wird über einen Chunker (vgl. [Abney, 1992]) in Textfragmente aufgespalten, wobei die Fragmente im Idealfall Phrasencharakter besitzen. Die Inhalte der Fragmente werden dann über eine andere TSR-Operation verarbeitet, als die Repräsentation der Fragmente selbst.

Da hierbei – ähnlich wie in der Aussagenlogik – kanonische Formen produziert werden, wollen wir zukünftig von einer *Disjunktiven Normalform* (*DNF*) sprechen, wenn ein TSR Ausdruck in der Form

$$TSR(A) = TSR(B_1) \vee TSR(B_2) \vee \ldots \vee TSR(B_n)$$

vorliegt, wobei jeder Ausdruck $TSR(B_i)$ der Form

$$TSR(B_i) = TSR(C_1) \wedge TSR(C_2) \wedge \ldots \wedge TSR(C_m)$$

unterliegt.

Ähnlich entsprechender Formalismen in der Aussagenlogik können demzufolge auch die TSR-Operationen AND und OR in komplexen Operationen zusammengefaßt werden. Zu beachten ist aber, daß für die TSR Methodik bisher keine weiteren Entsprechungen der logischen Junktoren wie Negation, Implikation, Äquivalenz etc. definiert wurden.

Im folgenden Beispiel wird ein (englischer) Satz anhand seiner Phrasenstruktur analysiert und - nach Streichung der Stopwörter „the“ und „over“ - in eine TSR-DNF Form transformiert:

Satz:

,,The quick brown fox jumps over the lazy dog‘‘

Phrasenstruktur:

```
(the quick brown fox/NP)
(jumps/VP)
(over/PP)
(the lazy dog/NP)
```

TSR-DNF Form:

```
TSR( TSR("quick") ∧ TSR("brown") ∧ TSR("fox") )
∨ TSR("jumps")
∨ TSR( TSR("lazy") ∧ TSR("dog") )
```

Äquivalent zur DNF-Form wird die *Konjunktive Normalform* (*CNF*) definiert, wobei lediglich $\vee$ und $\wedge$ vertauscht werden.

In der experimentellen Überprüfung des TSR-Ansatzes wurde - neben der Anwendung der OR-Operation auf den gesamten Text auch die Wirkungsweise der beiden Normalformen betrachtet (vgl. Abschnitt 5.3).

Kürzen und Beschneiden Nicht jede Information, die ein TSR enthält, ist in jeder Anwendungssituation gleichermaßen erwünscht: aus z.B. Performanzgründen kann es sinnvoll sein, bedeutende Merkmale zu selektieren und andere zu löschen. Auch eine Fokussierung auf eine bestimmte Domäne oder auf allgemein gebräuchliche Wortverwendungsweisen kann eine Löschung unerwünschter Merkmale voraussetzen, um Seiteneffekte (z.B. durch Fehler in der ODP-Datenbasis) zu minimieren.

In dem folgenden, hypothetischen Beispiel soll die Anwendung dieses Prinzips zur Verdeutlichung konkretisiert werden. In dem Satz:

> *I like movies and horse racing, i especially like „Dark Star“.*

soll geprüft werden, ob für den Ausdruck „Dark Star“ der Bereich „Filme“ dominiert, der z.B. durch die Kategorie */ Top / Arts / Movies* vertreten ist, oder z.B. die Kategorie */ Top / Sports / Equestrian* und damit der Bereich „Pferdesport“ eine besondere Gewichtung erfährt. In Tabelle 4.14 sind die wesentlichen, im TSR des o.a. Satzes enthaltenen Kategorien aufgeführt. Die Werte der Tabelle machen deutlich, daß beide Bereiche unterschiedlich stark enthalten sind (Gewichtung: $0,86$ und $0,04$ respektive), sich aber der Bereich „Filme“ stärker in verschiedene Kategorien aufteilt. Trotzdem würde bei Kürzung des TSRs mit einem Faktor $x = 0,3$, d.h. wenn alle Kategorien mit einem Gewicht von $< 0,3$ entfielen, der Bereich „Pferdesport“ aus dem TSR entfernt werden und lediglich der Aspekt „Film“ beibehalten werden.

Kategorie	**Gewicht**
$/Top/Arts/Movies$	$0,38$
$/Top/Arts/PerformingArts/Acting$	$0,45$
$/Top/Arts/Television/Programs/ScienceFiction$	$0,03$
	Summe : $0,86$
$/Top/Sports/Equestrian$	$0,04$
	Summe : $0,04$

Tabelle 4.14: Gewichtungsbeispiel TSR: Domäne Film contra Pferdesport

Die Pferde-Lesart würde nur dann dominieren, wenn ein entsprechender Kontext bestünde und in die Analyse einfließen würde - z.B. indem ein TSR für den gesamten Textabschnitt gebildet würde und mit den TSRs der einzelnen Sätze zusammengeführt würde. Eine andere Möglichkeit, diese Lesart hervorzuheben bestünde in der Änderung der ODP-Datenbasis, z.B. durch Einfügung einer geeigneten Menge an pferdesportbezogenen Knoten.

TSRs bieten zwei prinzipielle Ansätze, Knoten selektiv zu löschen:

1. Das Schwellwertverfahren (CUT-Verfahren) wird mit einem Schwellwert t parametrisiert und löscht sämtliche Knoten, deren Wert unter diesem Schwellwert liegt. Die Semantik ist hierbei, daß Knoten mit einer sehr geringen Bedeutung unter bestimmten Umständen (i.A. zur Komplexitätsreduktion) entfallen können; diese besitzen in der Regel auch einen geringen numerischen Wert. Die CUT-Operation ist in Abbildung 4.16 dargestellt – eine grafische Darstellung zur Verdeutlichung findet sich zusätzlich in Abbildung 4.17 auf der nächsten Seite.

```
procedure TSR.CUT(TSRa, Float t)          ▷ t : Schwellwert
    for v_a ∈ V_a do                      ▷ V : Knotenmenge von TSRa
        x = f(v_a)
        if x <= t then
            REMOVE(a, v_a)
        end if
    end for
end procedure
```

Abbildung 4.16: Kürzungsoperation CUT

2. Eine Fixmengenmethode, das TOP-Verfahren wird mit einem ganzzahligen Wert c parametrisiert. Lediglich die c höchstwertigen Knoten bleiben erhalten, die übrigen werden gelöscht. Dieses Verfahren kann benutzt werden, um die Baumgröße auf einen statischen Wert zu setzen - hierdurch werden strukturelle Gewichtungen stark priorisiert (ebenfalls zur Komplexitätsreduktion). Die TOP-Operation wird durch die Abbildungen 4.18 und 4.19 auf Seite 127 verdeutlicht (in den gewählten Beispielen entspricht ein CUT-Wert von $0,4$ zufälligerweise einem TOP-Wert von 4)

Der Effekt des Kürzens wurde bereits in [Winnemöller, 2005] publiziert, an dieser Stelle soll daher nur eine überblicksartige Darstellung des dort präsentierten Experimentes erfolgen: das in der genannten Quelle vorgestellte Klassifizierungs-Experiment hatte zum Ziel, die Wirkung verschiedener Kürzungsgrade auf die in den folgenden Punkten beschriebenen 3 artverschiedenen Korpora zu zeigen und in Relation zu einem traditionellen, wortvektorbasierten Algorithmus zu setzen:

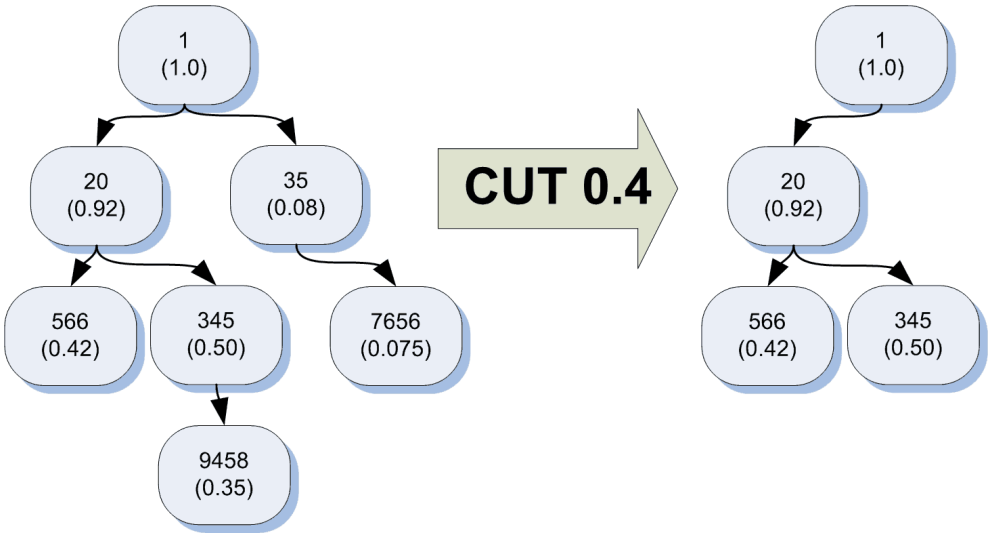

Abbildung 4.17: Beispiel CUT-Operation

function TSR.TOP(V, *Integer* c) ▷ V : geordnete Knotenmenge, c : gewünschte Zielknotenzahl
  SORT(V) ▷ Sortierung nach Werten
  $V_2 = \varnothing$ ▷ Leere Zielknotenmenge
  for $v_i \in V : 0 \leq i \leq c$ **do**
    PUT(v_i, V_2) ▷ v_i wird in V_2 kopiert
  end for
  return V_2
end function

Abbildung 4.18: Kürzungsoperation TOP

Das Korpus *small* beinhaltet ca. 150 Texte aus 5 Kategorien. Hierbei stammen die Texte aus 3 Internet-Newsgroups (alt.cooking, alt.business und sci.med), einer zufälligen Auswahl von Seiten der Java Development Kit Dokumentation und aus Exzerpten einer Online-Version der King James Bibel. Dieses Korpus enthält eine kleine Menge von „Off-Topic“ Texten (d.h. solchen Texten, die nicht dem Thema zugehörig sind, insbesondere sog. „Spam“ aus den eingesetzten Newsgroups)

Das Korpus *6newsgroups* besteht aus ca. 800 Nachrichten aus 6 verschiedenen Internet-Newsgroups: alt.atheism, comp.diagrams, rec.autos, sci.electronics, sci.med und sci.space. Dieses Korpus stellt eine Untermenge des im Information Retrieval Bereich häufig verwendeten sog. 20newsgroups Korpus dar (vgl. u.a. [McCallum et al., 1998]). Das Korpus besitzt - aufgrund der Natur seiner Herkunft - einen verhältnismäßig hohen Grad an Off-Topic Texten.

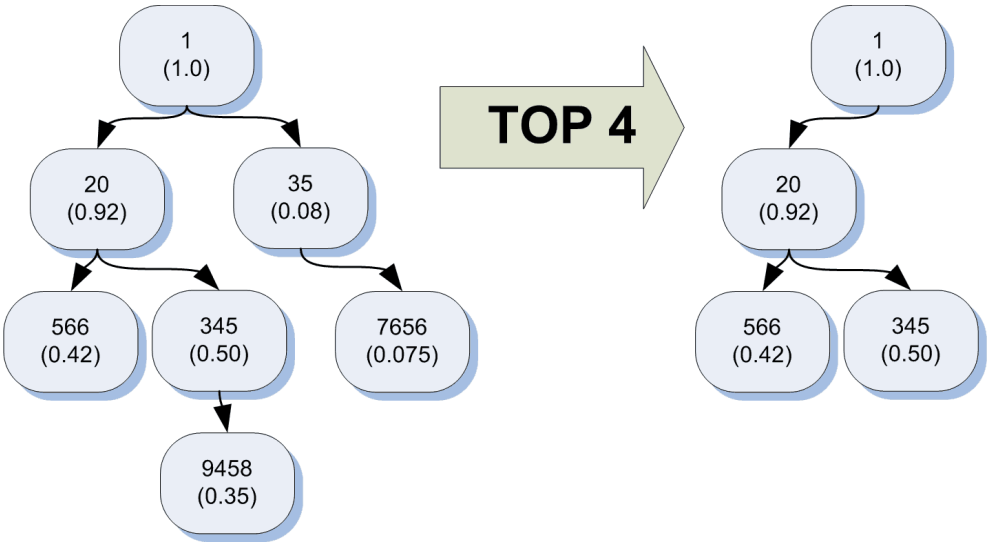

Abbildung 4.19: Beispiel *TOP*-Operation

Das Korpus *5abstracts* wurde aus ca. 1500 Kurzzusammenfassungen (engl. abstracts) von Artikeln aus 5 verschiedenen Wissenschaftsbereichen zusammengestellt: Wirtschaftswissenschaften (biz), Informatik (comp), Jura (law), Medizin (med) and Paläontologie (paleo). Dieses Korpus beinhaltet keine Off-Topic Texte.

Im Versuchsaufbau wurden zunächst TSRs für alle Textinstanzen der Korpora gebildet, indem aus jedem Text jeweils ein singulärer TSR konstruiert wurde (Anwendung der OR-Operation über alle n Wörter im Text, d.h. $TSRt = \mathbb{OR}(TSR(Wort_1), ..., TSR(Wort_n))$), welcher in die Vektordarstellung überführt wurde. Dann wurden im Rahmen einer 4-fachen Kreuzvalidierung drei Viertel der TSR-Vektorenmenge verwendet, um für jedes Korpus einen *decision tree* Klassifikator zu trainieren (ein maschinelles Lernverfahren, hier wurde die WEKA-Implementation verwendet, vgl. [Witten & Frank, 2000]) – es handelt sich daher um ein überwachtes Experiment. Das verbleibende Viertel der Textinstanz-TSR-Vektoren dienten anschließend als Test-Anwendungseingabedatenmenge der Klassifikatoren. Die Klassifizierungsaufgabe bestand in diesem Falle darin, die Menge dieser Testinstanzen den korrekten Ausgangskorpora zuzuweisen.

Parallel hierzu wurden die Textinstanzen außerdem einem üblichen Wortvektorverfahren zugeführt, welches die Klassifikatoren über Term-Dokument-Matrizen trainierte – entsprechende Beispiele und Beschreibungen der Algorithmen finden sich bei Witten und Frank (vgl. [Witten & Frank, 2000]). Auch hier wurde eine 4-fache Kreuzvalidierung angewendet.

Vor der Trainingsphase der Klassifikatoren wurden die TSR-Vektoren verschiedenen Kürzungsszenarien unterworfen, die in Tabelle 4.15 auf der nächsten Seite

dargestellt sind (TSR_{20} steht hierbei für eine Kürzung der TSR-Knotenmenge auf maximal 20 Knoten, TSR_{40} auf entsprechend 40 Knoten, TSR_{∞} bedeutet keine Kürzung und WV repräsentiert den Wortvektorversuchsaufbau).

Korpus	Typ	Volumen	Anzahl Knoten	Zeit	Accuracy
small	TSR_{20}	64 K	123	0.38	92 %
small	TSR_{40}	136 K	278	0.22	89 %
small	TSR_{∞}	1,4 M	3315	2.73	86 %
small	WV	1,1 M	3407	2.78	86 %
6newsgroups	TSR_{20}	659 K	352	5.08	64 %
6newsgroups	TSR_{40}	1,5 M	849	16.2	62 %
6newsgroups	TSR_{∞}	18,9 M	10842	275	64 %
6newsgroups	WV	21,9 M	14041	326	72 %
5abstracts	TSR_{20}	1,4 M	388	8.42	85 %
5abstracts	TSR_{40}	3,8 M	1089	25.5	86 %
5abstracts	TSR_{∞}	36,5 M	10427	213	92 %
5abstracts	WV	45,9 M	14847	273	96 %

Tabelle 4.15: Experiment: Kürzen von TSRs

Zu sehen ist, daß im gemischten *small*-Korpus das Verfahren mit der stärksten Kürzung das beste Resultat zeigt, während es in den beiden anderen Versuchen jeweils die Wortvektorvarianten sind. Eine Diskussion dieses Ergebnisses erfolgte bereits in [Winnemöller, 2005] und ist hier von minderem Interesse, beachtenswerter ist aber die Tatsache, daß der effektive Unterschied zwischen dem schlechtesten und dem besten Ergebnis eines Versuches ungefähr 7 Prozent beträgt, der Lastunterschied jedoch erheblich ist. Dies läßt die Deutung zu, daß hohe Gewinne in Begriffen von Rechengeschwindigkeit und benötigtem Speichervolumen durch vergleichsweise kleine Minderungen der zu erwartenden Accuracy möglich sind.

Einsatzmöglichkeiten der Informationsbegrenzung sind prinzipiell auch in der Literatur vertreten: einen diesbezüglich etwas ungewöhnlichen Ansatz, in welchem nicht (wie sonst eher üblich) Texte oder Dokumentverweise analysiert werden, sondern Bedeutungsvarianten, stellte Chen 1998 vor (vgl. [Chen & Chang, 1998]). Dies ist ein für den Bereich sense discrimination interessantes Verfahren, da es feine Unterscheidungen zwischen Bedeutungsvarianten eliminiert und damit zur Verbesserung der Performanz dienen kann, es ist aber nur wenig hilfreich, wenn Bedeutungsvarianten - wie in den SENSEVAL-Evaluationen (vgl. Kapitel 5) - fest vorgegeben werden. Der Vorschlag von Chen, kleine Bedeutungsunterschiede zu eliminieren, wird prinzipiell durch das TSR-Kürzungsverfahren umgesetzt.

Ableitung von TSRs. Als komplexes Beispiel für den Einsatz der Verknüpfungs- und Kürzungsoperationen dient die Konstruktion abgeleiteter TSRs

(*Teaching*[8]).

Die aus dem ODP erstellten initialen TSRs beinhalten in der Regel neben relevanten Informationen auch eine Menge irrelevanter Informationen, das sog. „Rauschen“. Beispielsweise wird die Kategorie *Freizeit* durch „Alles rund um Freizeitaktivitäten und Hobbys.“ beschrieben, wobei *Freizeitaktivitäten* und *Hobbys* offensichtlich relevante Informationen darstellen, das Wort *rund* jedoch im Rahmen einer Redewendung gebraucht wird und an dieser Stelle zum Rauschen gehört[9].

Um das ungewollte Rauschen zu dämpfen, kann es sinnvoll sein, das TSR-basierte Wissen bestimmter Wörter zu konsolidieren, indem man sie mit relevanten Wörtern verknüpft, also per OR-Operation mit dem Primärwort zusammenführt. (Hierbei ist allerdings Vorsicht geboten, da in realen Textsituationen das „Rauschen“ häufig akzeptiert und benötigt wird, z.B. in Redewendungen). Ein wesentliches Ergebnis der Ableitung besteht darin, „ähnliche“ Begriffe - und Textfragmente - näher zusammenzubringen (im Sinne der Wittgenstein’schen Familienähnlichkeit), im technischen Sinne also den TSR-Distance Wert zwischen ihnen zu minimieren, um eine bessere Abgrenzung gegenüber unähnlichen Begriffen zu bewirken. Ein anderes Ziel der Ableitung liegt in der Bestimmung bisher unbekannter Begriffen, d.h. solchen, die bisher nicht in der TSR-Datenbank enthalten sind.

Um dieses Prinzip zu verdeutlichen, soll es an einem kleinen Beispiel erläutert werden: Es sollen verschiedene Begriffe für Farben (*colour, blue, red, yellow*), einige für Nicht-Farben (*clock*, *browser*) und ein nicht in der TSR-Datenbasis enthaltenes Kunstwort (*mycolour*) verwendet werden, um die verschiedenen Wirkungsweisen der Ableitungsprozedur aufzuzeigen. Der die Wissensdomäne charakterisierende Begriff *„colour*“ soll hierbei als „Sekundärterm“ bezeichnet werden, die eigentlichen Inhaltsbegriffe werden „Primärterme“ genannt.

Tabelle 4.16 zeigt übersichtshalber die TSR-Größenparameter der genannten Begriffe.

In Tabelle 4.17 werden die TSRs der einzelnen Primärterme mit dem hier untersuchten Sekundärterm *colour* verglichen. Es zeigt sich eine charakteristische Werteverteilung, der sich entnehmen läßt, daß die Farb-Begriffe *blue*, *red* und *yellow* dem Begriff *colour* näher sind als die Begriffe *clock* und *browser* (da sie die geringere normalisierte Distanz aufweisen). Der Begriff *mycolour* zeigt definitionsgemäß eine „unendliche Unähnlichkeit“.

In der eigentlichen Ableitungsphase wird nun der Sekundärbegriff mit den Primärbegriffen verknüpft:

[8] da der eigentlich semantisch passendere Begriff „Training“ im NLP-Kontext üblicherweise durch das „trainieren“ statistischer maschineller Lernverfahren vorbelegt ist, wird an dieser Stelle der Begriff „Teaching“ eingeführt

[9] Die Wörter *Alles, um* und *und* werden an dieser Stelle fallengelassen

Term	SIZE	WEIGHT
colour	531	216,74
blue	9729	7191.63
red	7996	8070.64
yellow	1854	988.14
clock	777	404.44
browser	856	564.89
mycolour	0	0.0

Tabelle 4.16: Teaching - Anwendungsbeispiel Farben

Primärterm/ Sekundärterm	**Size**	**Size A/B**	**Distance**	**Distance/ (A/B)**
blue/colour	9729	18.32	0.85	0,05
red/colour	7996	15.05	0.85	0,06
yellow/colour	1854	3.49	0.83	0,24
clock/colour	777	1.46	0.84	0,57
browser/colour	856	1.61	0.89	0,56
mycolour/colour	0	0	1,0	∞

Tabelle 4.17: Teaching - Initialwerte

$$TSR(colour\ blue\ red\ yellow) = TSR(colour) \vee TSR(blue) \vee TSR(red) \vee TSR(yellow)$$

Äquivalent hierzu wird der „neue“ Begriff *mycolour* über die Begriffe blue, red, yellow definiert, so daß sich die folgenden neuen TSR-Größenparameter ergeben:

Term	SIZE	WEIGHT
colour blue red yellow	16494	0.005
blue red yellow	16273	0.005

Um die erfolgten Änderungen untersuchen zu können, wird der Vergleich zwischen Primär- und Sekundärtermen aus Tabelle 4.17 wiederholt (der Begriff *mycolour* wird nun ebenfalls untersucht), die diesbezüglichen Ergebnisse für *colour* und *mycolour* sind in Tabelle 4.18 auf der nächsten Seite abgebildet. Deutlich zu sehen ist hierbei, daß sich die Werteverteilung stärker ausgeprägt hat. Außerdem wird die Ähnlichkeit der Werteverteilung zwischen dem geänderten Begriff *colour*

und dem neu definierten Begriff *mycolour* deutlich, sowohl im Bezug zu den anderen Begriffen, als auch in der großen Ähnlichkeit zueinander (durch den sehr kleinen Distanzwert festzustellen). Gleichzeitig entfernen sich unähnliche Begriffe (mit einem höheren Distanzwert, im Beispiel am Wert von *clock* zu erkennen) voneinander – dies wirkt sich als Minderung des „Rauschens“ aus.

Primärterm/ Sekundärterm	**Size**	**Size A/B**	**Distance**	**Distance/ (A/B)**
blue/colour	9729	0.58	0.75	1.28
red/colour	7996	0.48	0.81	1.68
yellow/colour	1854	0.11	0.74	6.61
clock/colour	777	0.04	0.88	18.81
browser/colour	856	0.05	0.93	18.04
blue/mycolour	9729	0.59	0.64	1.07
red/mycolour	7996	0.49	0.73	1.49
yellow/mycolour	1854	0.11	0.62	5.46
clock/mycolour	777	0.04	0.85	17.94
browser/mycolour	856	0.05	0.92	17.50
colour/mycolour	16494	1.01	0.32	0.31

Tabelle 4.18: Teaching - neue Werte

Ein Experiment mit einem einzelnen Datensatz kann zwar schwerlich als Nachweis für die Wirksamkeit eines Verfahrens dienen, eine belastbare Untersuchung fällt jedoch aus dem Rahmen dieser Einführung und soll Gegenstand zukünftiger Forschung sein, insgesamt wird aber gezeigt, daß der TSR-Ansatz über die Methode der Erzeugung abgeleiteter TSRs zu verschiedenen Zwecken (Konsolidierung von bestehenden TSRs oder Erzeugung neuer TSRs) in Bezug auf das enthaltene begriffliche Volumen beliebig erweiterbar ist.

Interessant ist die Einordnung des TSR-Ansatzes in die Systematik des Kapitels 2: Kollokationen sind durch TSRs über die OR-Operation gut darstellbar - sofern eine hinreichende empirische Basis besteht; ein Zusammenhang von syntaktischer und semantischer Wortassoziation könnte über Familienähnlichkeit und die hierarchische Struktur der TSRs untersucht werden. Außerdem können über die

TSR-Gewichtungsfunktion Präferenzen abgebildet werden. Damit würde das TSR-Verfahren als Ansatz im Bereich von Ontologien und Korpora klassifiziert werden.

4.2 Prototypische Realisierung

Zum Nachweis der Funktionsfähigkeit der in diesem Kapitel erläuterten Prinzipien und Prozeduren wurde ein prototypisches System entwickelt, welches auch als Basis für die in Kapitel 5 auf Seite 139 beschriebenen Versuchsaufbauten diente. In den folgenden Abschnitten werden die Architektur und die Systemeigenschaften dieses Prototypen behandelt.

Inhalt dieses Abschnittes ist die im Zusammenhang mit der vorliegenden Arbeit entwickelte prototypische Systemarchitektur (Abschnitt 4.2.1) und die Erläuterung des hierauf basierenden realisierten Prototypen (Abschnitt 4.2.2)[10].

4.2.1 Architektur

Für den Entwurf einer geeigneten Architektur wurde das Client-Server Modell gewählt, da hierdurch verschiedene Funktionen auf unterschiedlichen Hosts ausführbar werden. Dies ist aus pragmatischen Gründen wünschenswert, um das benötigte Speichervolumen und die aufzuwendende Rechenzeit zu minimieren. Zudem können Konsolidierungen der Datenbasis an eine große Menge an Clients weitergereicht werden. Eine Verteilung der Rechenlast scheint besonders für die Online-Nutzung von TSR-basierten Systemen ratsam, z.B. bei der denkbaren automatischen Kategorisierung von eMails bei Abholung der Mails vom Server.

Die prototypische Architektur basiert auf der Erweiterung eines Standard SQL-Datenbankservers um NLP- und TSR-spezifische Funktionalitäten, wie in Abbildung 4.20 auf der nächsten Seite skizziert.

Hierdurch ist es möglich, auch ohne Verwendung einer speziellen proprietären Schnittstelle über standardisierte Kanäle mit dem Server zu kommunizieren. Zudem können TSR-Anwendungen bestimmte Eigenschaften des DBMS nutzen (z.B. Transaktionen, Sicherheit, Backupstrategien, Clustering, denkbare Einbettungen in weitere Anwendungen, etc.). Abgesehen von der Möglichkeit der reinen SQL-Konnektivität wurde zusammen mit dem Server für Anwendungsentwickler auch eine entsprechende Programmierschnittstelle (engl. API, Abk. für Application Programming Interface) entwickelt.

Der Server liefert die grundlegenden Funktionalitäten für TSRs, d.h. es wird die Möglichkeit geboten, aus Wörtern und Textfragmenten TSRs zu erzeugen, diese durch die definierten Operationen zu verschmelzen und ferner Informationen über einzelne TSRs zu erhalten (DEPTH, WIDTH, etc.). Darüber hinaus ist die Erzeugung der initialen Datenstrukturen aus den Daten des ODP-Internetverzeichnisses ebenfalls möglich. Außerdem wird eine NLP Komponente angeboten, die eine flache syntaktische Vorverarbeitung von Texten bieten soll (Satzerkennung, Tokenization und PoS-Tagging).

[10] Obwohl es sich intentionsgemäß um einen Prototypen handelt, ist jedoch aufgrund der guten Performanz des Systemes ein Produktionseinsatz nicht unrealistisch.

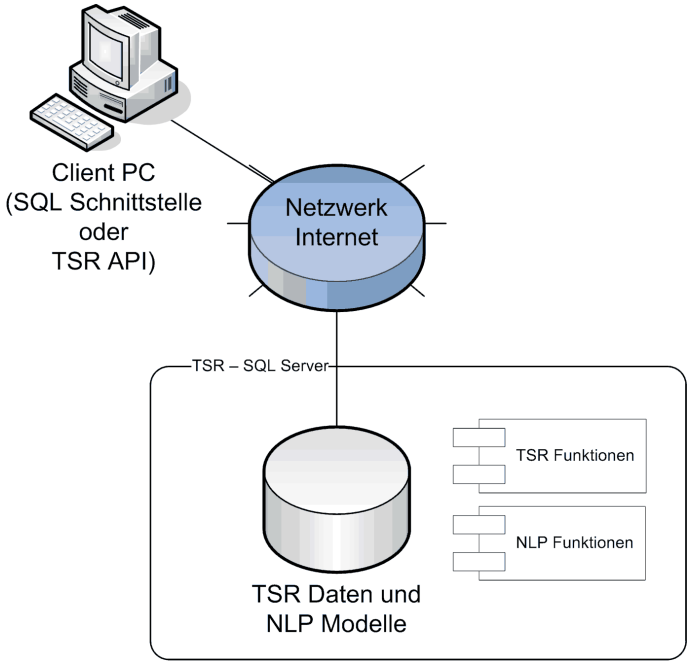

Abbildung 4.20: TSR Server Architektur

Die prinzipielle Funktionsweise soll anhand des Szenarios von Abbildung 4.21 verdeutlicht werden:

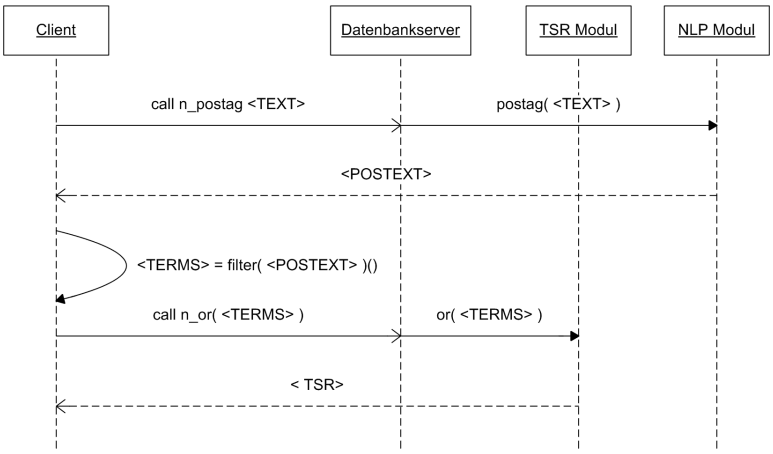

Abbildung 4.21: Nutzungsszenario TSR

1. Client benötigt PoS Information, sendet eine Anfrage an den NLP Modul über SQL Schnittstelle; erhält annotierten Text

2. Client verarbeitet die PoS Information, um eine Liste relevanter Wörter

(Termliste) zu erstellen.

3. Client sendet Anfrage per SQL an den TSR Modul, um einen TSR (per OR Operation aus den einzelnen Wörtern gebildet) zu erhalten

4. Clientseitige Weiterverarbeitung des TSRs (Vergleich oder Verschmelzung mit anderen TSRs, etc.)

Die Nutzung der TSR-Informationen erfolgt dann innerhalb des Anwendungsprogramms in Bezug auf die jeweilige Problemstellung.

4.2.2 Implementation

Die in Abbildung 4.20 auf der vorherigen Seite skizzierte Architektur wurde in Hinblick auf einen späteren produktiven Einsatz realisiert, d.h. es wurden – unabhängig von qualitativen Aspekten der Implementation (Funktionalität) – auch quantitative Aspekte beachtet (Durchsatz, Effizienz der Umsetzung, Leistung der Anwendung, etc.).

Aus diesem Grunde wurde die Programmiersprache Java als Anwendungsplattform gewählt. Ein externes javabasiertes RDBMS – das so genannte H2 Datenbanksystem (vgl. [Mueller, 2006]) – wurde verwendet, um eine an einer schnellen Produktivsetzung orientierten Serverumgebung schaffen zu können. Die Wahl einer schnellen SQL Datenbank hat zudem gegenüber vielen anderen Ansätzen (z.B. HTTP-basierte Methoden) den Performanz- und Sicherheitsvorteil einer verbindungsorientierten Kommunikation.

Durch die extensive Verwendung von Bausteinen aus dritter Hand konnte der Entwicklungsaufwand im Wesentlichen auf die Erstellung der TSR Funktionalität, und die der Komponentenbindungen reduziert werden.

4.2.2.1 Erstellung der Datenbank

Zur Erstellung der Datenbasis wurde das in Abschnitt 4.1.2 erläuterte Verfahren umgesetzt. Innerhalb der Datenbank wurde eine Tabelle für ODP Pfadangaben (Tabelle *Structure*) definiert, sowie eine Tabelle für TSRs (Tabelle *TSR*).

Die Konstruktionsprozedur wird in zwei Stufen durchlaufen[11]:

1. Die erste Stufe dient zur Erstellung einer Liste der auftretenden Wörter aus den ODP-XML Datendateien. Diese Wörter werden in Kombination mit Angaben zur Frequenz des jeweiligen Wortes in der jeweiligen ODP Kategorie

[11] Das Verfahren wurde bereits in Abschnitt 4.1.2 auf Seite 101 erläutert, die hier beschriebenen Details beziehen sich eher auf die tatsächliche technische Umsetzung.

Listing 4.3: Erstellung der TSR Datenbank: Exzerpt Stufe 1

```
top          top              1.0   1.0
level        top/reference    1.0  61.0
study        top/reference    1.0  61.0
open         top/reference    1.0  61.0
topics       top/reference    1.0  61.0
directories  top/reference    2.0  61.0
access       top/reference    1.0  61.0
under        top/reference    1.0  61.0
categories   top/reference    1.0  61.0
```

gespeichert und mit einer Angabe des Eintragspfades innerhalb der ODP Hierarchie versehen. Ein Exzerpt der entstehenden Datei ist in Abbildung 4.3 abgebildet (Zeileninhalte sind: Term, ODP-Pfad, Auftreten des Terms im angegebenen Pfad, Anzahl aller Terme im angegebenen Pfad). Weiterhin werden die Pfadangaben eindeutig in der Tabelle „Structure“ der Datenbank gespeichert.

2. Die zweite Stufe erzeugt aus der zuvor angelegten Liste - nach vorheriger externer alphanumerischer Sortierung - die Einträge der TSR Tabelle.

Der Gesamtvorgang benötigt auf einem Pentium IV Rechner (1 GB RAM, 1,2 GHz) ca. 8 Stunden. Die aktuelle Datenbank enthielt zum Zeitpunkt ihrer Erstellung (April 2006) 678.998 ODP Pfadangaben und 1.273.797 TSRs für sämtliche, im ODP enthaltenen Wörter.

4.2.2.2 Realisierung des Datenbankzugriffes

Der Zugriff auf NLP- und TSR-Funktionen wurde über SQL „Call“ Statements realisiert, zurückgegeben wird in jedem Fall eine sog. „SQL ResultSet“, d.h. eine tabellarisch aufbereitete Ergebnismenge. Die für die TSR-Verarbeitung relevanten Statements werden in Tabelle 4.19 vorgestellt.

Während die TSR Funktionalität durch den Autor dieser Arbeit selbst entwickelt wurde, wurde zur Nutzung von NLP Funktionen zur Vorverarbeitung das sog. OpenNLP-Toolkit (vgl. [Baldridge & Morton, 2006]) integriert, insbesondere die Wortartenanalyse wird durch dieses Toolkit ermöglicht.

Die Nutzung der Funktionen erfolgt über Standard-Datenbankmechanismen; d.h. über eine JDBC (Java) oder ODBC (Windows, Unix, Mac) Schnittstelle.

SQL call	Modul	Beschreibung
n_nlp	NLP	Erzeugt Analyse eines Textes im XML-Format (PoS, Chunking)
n_postag	NLP	Erzeugt PoS Analyse eines Textes (Format: „<Wort>/<PoS>“)
n_stats	TSR	Ausgabe bestimmter Datenbankstatistiken (Anzahl gespeicherter ODP-Pfade, Anzahl gespeicherter TSRs)
n_and	TSR	Verknüpfung der dem vorgegebenen Text (d.h. den in diesem enthaltenen Worten) entsprechenden einzelnen TSRs per AND-Operation.
n_or	TSR	Verknüpfung der dem vorgegebenen Text (d.h. den in diesem enthaltenen Worten) entsprechenden einzelnen TSRs per OR-Operation.

Tabelle 4.19: Funktionen für TSR Verarbeitung

4.3 Zwischenergebnis

In diesem Kapitel wurde ein Vorschlag zur Umsetzung der in Kapitel 3 erarbeiteten theoretischen Grundlagen vorgestellt.

Hierzu wird, wie bereits dargestellt, die Menge der initialen TSRs, die als Grundlage für spätere produktive Operationen dient, in mehreren Phasen aus einem Internetverzeichnis erzeugt. Der Erzeugungsvorgang extrahiert zu jedem Wort, das im Internetverzeichnis enthalten ist, einen Teilbaum des Internetverzeichnisses, in dessen Knoten das jeweilige Wort auftritt. Die Okkurrenzinformation wird zusammen mit einer knotenweisen Gewichtung im TSR-Baum gespeichert. Hierbei ist zu beachten, daß die Anzahl der Elemente, aus denen die TSRs konstruiert werden, zwar sehr groß (initial etwa 680.000) und ggf. erweiterbar, aber endlich ist.

TSRs für weitere Wörter können durch die auf den TSRs definierten Operationen durch Ableitung bestehender TSRs konstruiert werden, so daß zu jedem Zeitpunkt zwar nur eine endliche Menge von TSRs in einem System bekannt ist, diese aber prinzipiell beliebig erweiterbar ist. Das TSR Instrumentarium umfaßt zurzeit eine kleine Anzahl einfacher Algorithmen der Analysis und der Graphentheorie, die in der Literatur auf vielen Ebenen von NLP-Systeme tatsächlich eingesetzt werden. TSRs lassen sich daher über eine Transformation in TSR-Indexvektoren leicht anstelle von Wortvektoren in bestehende Machine Learning und Information Retrieval Anwendungen usw. einbinden.

Um die Wirksamkeit des TSR-Ansatzes auch empirisch belegen zu können, wurde

eine Architektur und ein prototypisches System, welches auf dieser Architektur basiert, entwickelt.

Der erstellte Software-Prototyp wurde auf Basis eines relationalen Datenbankmanagementsystem (RDBMS) erstellt. Verschiedene TSR-relevante Funktionen, sowie Funktionen für eine natürlichsprachliche Vorverarbeitung wurden in Form einer Java-Klassenbibliothek realisiert, die über interne Datenbankaufrufe eine Entwicklerschnittstelle für externe Anwendungen zur Verfügung stellt. Dieses Vorgehen ist durch ein gutes Performanzverhalten im Verbindungsaufbau und im Datendurchsatz, sowie durch die Verwendbarkeit der verschiedenen RDBMS Produktivmittel (Standard-Abfragesprache, Sicherheitskonzept, Skalierbarkeit durch Clustering, etc.) gerechtfertigt.

Kapitel 5

Evaluation des Ansatzes

Während im vorhergehenden Kapitel TSRs als reichhaltige Informationsstrukturen präsentiert wurden, soll in diesem Kapitel aufgezeigt werden, warum die Verwendung dieser Strukturen besser ist, als zueinander in Beziehung gesetzte atomare Terme bzw. Wörter – und was „besser“ in diesem Zusammenhang bedeutet.

Inhalt dieses Kapitels ist daher die empirische Stützung des TSR-Ansatzes durch entsprechende experimentelle Ergebnisse.

Um allgemein Probleme der WSD-bezogenen Textverarbeitung untersuchen zu können, wurde zunächst ein prototypisches Evaluationssystem entwickelt, welches in der Lage ist, den TSR-Ansatz funktional umzusetzen. Dieser Prototyp wurde auf Basis des in Abschnitt 4.2.2 auf Seite 135 vorgestellten Systems entwickelt und ist in der Lage, die verschiedenen, in dieser Arbeit vorgestellten Experimente durchzuführen.

Die Zielsetzung der WSD-bezogenen Experimente werden in Abschnitt 5.1 erläutert, wobei auf besondere Aspekte der Evaluation von Systemen des WSD-Umfeldes im Rahmen der SENSEVAL-Wettbewerbe in Abschnitt 5.2 auf Seite 143 eingegangen wird. Im zweiten Teil dieses Abschnittes wird der Versuchsaufbau der TSR-Experimente im Detail dargestellt.

Die Ergebnisse der Untersuchung werden in Abschnitt 5.3 auf Seite 148 diskutiert.

In Abschnitt 5.4 auf Seite 151 wird schließlich ein Zwischenergebnis angegeben.

5.1 Ziel der Untersuchung

Zweck der Evaluation ist es, festzustellen, ob durch die TSR-Methodik im Bereich der Auflösung von Wortmehrdeutigkeiten signifikant bessere Ergebnisse erbracht werden können als über vergleichbare herkömmliche Methoden. Hierbei wurde

die Verwendung von Wortvektoren als Repräsentant konventioneller Methoden gewählt, da sie eine Grundlage der meisten im SENSEVAL English Lexical Sample Wettbewerb bewerteten Systeme darstellt.

Aufgrund der Gefahr eines Mißverständnisses soll dieser Punkt noch einmal herausgestellt werden: es ist nicht das Ziel der Untersuchung, ein besonders leistungsfähiges WSD-System zu implementieren, sondern es geht darum, die Wirksamkeit einer anders gearteten (d.h. einer nicht konventionellen merkmalsbasierten) Semantik zu belegen. Effektiv soll damit gleichzeitig die positive Auswirkung einer entsprechenden semantisch-pragmatischen Theorie im Vergleich zu einer rein lexikalisch-syntaktischen theoretischen Basis gezeigt werden.

Da die Zielsetzung der TSR Methode auf die *bedeutungsgerechte* Verarbeitung von Texten gerichtet ist, wird entsprechend nach folgenden Grundsätzen evaluiert:

1. **Nicht-überwachte Verarbeitung**: Die in den Experimenten zum Einsatz kommenden Modelle sollen nicht durch Einsatz von Trainingsdaten trainiert bzw. konfiguriert werden.

   Hierdurch soll auf die Wirkung des Einsatzes von Weltwissen und einer Repräsentation der Textbedeutung fokussiert werden, anstelle auf (üblicherweise opake) statistische Relationen, welche nicht notwendigerweise eine semantische Dimension besitzen. Beispiele derartiger Relationen finden sich in der Literatur über Maschinenlernen (Einen Überblick bietet beispielsweise [Witten & Frank, 2005]).

   Ein wesentlicher Grund für diese Entscheidung besteht darin das die Qualität der derzeit für WSD verwendeten ML-Verfahren in großem Maße von der Verfügbarkeit großer Trainingskorpora abhängt , wie in einem prototypischen Experiment von Banko und Brill belegt wird (vgl. [Banko & Brill, 2001]). Resnik und Yarowsky bestätigen dieses Faktum ebenfalls in ihren „Observationen“ (vgl. [Resnik & Yarowsky, 1997]). Yarowsky (vgl. [Yarowsky, 1993]) und Schütze (vgl. [Schütze, 1995], in: [Leacock et al., 1998]) entwickelten zwar alternative Ansätze, in denen Pseudo-Wörter aus Paaren "echter" Wörter gebildet werden - hierdurch ist es sehr einfach möglich, größere Trainings- und Testdaten zu erzeugen - allerdings weisen die Autoren auch auf gewisse Verfälschungen der Ergebnisse durch die künstliche Erzeugungsweise der Daten hin.

2. **Syntaktische Vorverarbeitung**: Die Zuhilfenahme einer syntaktischen Vorverarbeitungsstufe soll zulässig sein, sofern keine Daten manipuliert oder hinzugefügt, sondern zur Effzienzsteigerung Stördaten entfernt werden.

   Eine (flache) syntaktische Analyse ist für eine effiziente Steuerung der TSR Methode wünschenswert. Syntaktische Attribute werden aber für die hier vorgestellte Evaluation nicht genutzt, um die Kernaussage der Versuche nicht unnötig zu schwächen.

3. **Nicht-iterative Berechnung**: Eine iterative Wiederverwendung von bereits berechneten Ergebnissen, z.B. durch Clustering- oder Bootstrappingverfahren, soll nicht eingesetzt werden[1].

   Inkrementell-iterative Verfahren werden nicht verwendet, da sie zwar die Qualität der Ergebnisse erhöhen können, aber nicht zur experimentalen Grundaussage beitragen.

4. **Keine Nutzung von a-priori Wissen**: (Einbeziehung von Domänenwissen, z.B. in Form von Heuristiken) Der TSR-Ansatz wird zwar durch ein spezifisches, standardisiertes Experiment im WSD-Umfeld überprüft, jedoch ist das Verfahren selbst auf eine generische Anwendbarkeit im Bereich der automatisierten Textverarbeitung ausgerichtet. Aus diesem Grund wird die Allgemeingültigkeit der Anwendung gefordert.

   Ein Beispiel für eine nicht-allgemeingültige Heuristik ist die sog. *First Sense Heuristik*. Damit wird der Umstand beschrieben, daß in Lexika häufig die am weitesten verbreitete Wortbedeutung zuerst genannt wird. Die Nutzung der First Sense Heuristik alleine bedeutet einen erheblichen Vorteil im Rahmen einer WSD-Aufgabenstellung, trägt aber nur in eingeschränktem Maße zur Lösung bei (da der textuelle Kontext der Instanzdatensätze nicht betrachtet wird).

Diese Punkte sollen helfen, störende Fremdeinflüsse auf die Anwendung der im folgenden Abschnitt erläuterte Bewertungsmethodik zu minimieren.

Die meisten Systeme, die in der English Lexical Sample Aufgabe eingesetzt wurden, erfüllen die in Abschnitt 5.2 angeführten Rahmenbedingungen nicht. Um aber dennoch eine Vorstellung der experimentellen Ergebnisse zu ermöglichen, werden in Tabelle 5.1 auf der nächsten Seite die 9 Systeme, die nicht-überwacht (*unsupervised*) arbeiten, aufgeführt und kurz beschrieben[2].

Die in Abschnitt 5.1 definierten Rahmenbedingungen lassen einen direkten Vergleich der in Tabelle 5.1 aufgeführten Systeme ebenfalls nicht zu:

- entweder werden Ergebnisse untergeordneter Algorithmen durch maschinelle Lernverfahren aufbereitet und damit die für die Zwecke dieser Arbeit relevanten Erkenntnisse verdeckt, z.B. im Falle von Cymfony (vgl. [Niu et al., 2004]) und Prob0 (vgl. [Preiss, 2004])

- oder Fremddaten werden über a-priori Heuristiken unzulässig genutzt, z.B. durch wsdiit (vgl. [Ramakrishnan et al., 2004a]) und KUNLP (vgl. [Seo

[1] Derartige Verfahren wurden durch Yang et al im Rahmen Information-Retrieval untersucht, vgl. [Yang et al., 2003]

[2] Die Beschreibung entspricht hierbei derjenigen von Mihalcea et al. und ist in diesem Sinne ein vom Autor dieses Werkes übersetzer Exzerpt von [Mihalcea et al., 2004]

System / Referenz	Kurzbeschreibung	Precision	Recall
wsdiit / [Ramakrishnan et al., 2004a]	Lesk-ähnlicher Algorithmus, sowie Nutzung der WordNet-Hypernymrelation; verschiedene Parameter	66,1	65,7
Cymfony / [Niu et al., 2004]	Clustering von Kontextinformation aus Test- und Trainingsdaten der SENSEVAL-3 Korpora	56,3	56,3
Prob0 / [Preiss, 2004]	Kombination zweier nicht-überwachter Algorithmen (über PoS und Wort-Frequenzdaten aus WordNet-Bedeutungsvarianten)	54,7	54,7
clr04-ls / [Litkowski, 2004]	MRD-basierter Algorithmus; Nutzung syntaktischer, semantischer und Diskursinformatioen	45,0	45,0
CIAOSENSO / [Buscaldi et al., 2004]	Conceptual Density Algorithmus, Domäneninformation (WordNet Domains)	50,1	41,7
KUNLP / [Seo et al., 2004]	Nutzung von Wort-Kookurrenzen in verschiedenen WordNet-Relationen	40,4	40,4
Duluth-SenseRelate / [Pedersen, 2004]	Nachbarschaften von Wörtern in WordNet-Glossen	40,3	38,5
DFA-LS-Unsup / [Férnandez-Amorós, 2004]	First Sense Heuristik; Vergleich von Wort-Kookurrenzen und Synonymen; Syntaktische Muster	23,4	23,4
DLSI-UA-LS-NOSU / [Vázquez et al., 2004]	Domänenbasierte Assoziation zu WordNet Glossareinträge	19,7	11,7

Tabelle 5.1: Ergebnisse Aufgabe English lexical Sample (nicht überwachte Systeme)

et al., 2004]). In einigen Fällen werden Heuristiken wie die erwähnte First-Sense-Heuristik verwendet, z.B. durch DFA-LS-Unsup (vgl. [Férnandez-Amorós, 2004]).

Eine Einbettung des TSR-Ansatzes in Algorithmen der beteiligten Systeme wäre ein geeigneter Test für die Leistung des TSR-Ansatzes in Hinblick auf realistische Szenarien, wie sie in Tabelle 5.1 dokumentiert werden. Dies liegt allerdings jenseits des Rahmens der hier zu erbringenden Erkenntnisse (Leistungsmessung mit minimalen Abhängigkeiten) und soll in zukünftigen Untersuchungen analysiert werden.

5.2 Bewertungsmethodik und Testszenarien

In diesem Abschnitt werden die Umstände erläutert, unter denen der TSR-Ansatz überprüft wird. Um einem anerkannten Standard zu folgen, wurden – wie in Abschnitt 2.3.2 auf Seite 59 dargelegt – als Rahmen der Auswertung die Gegebenheiten des SENSEVAL-3 Workshops zugrunde gelegt, insbesondere die speziellen Rahmenbedingungen der Aufgabe „English Lexical Sample“, die im Folgenden erläutert werden .

Da es sich beim TSR Testsystem um ein nicht-überwachtes System handelt, und da bei der Überprüfung des TSR-Ansatzes das SENSEVAL-3 basierende Testverfahren nach den eben angeführten Maßgaben eingesetzt werden soll, können sowohl die Test- als auch die Trainingsdatensätze[3] der Aufgabe SENSEVAL-3 English Lexical Sample Task für eine Evaluation genutzt werden (Prinzipiell wäre aber auch eine Verwendung von anderssprachlichen Datensätzen aus dem SENSEVAL-Umfeld möglich).

Diese bestehen auf Seite der Bedeutungsvarianten aus entsprechenden WordNet-Einträgen, auf Seite der Test-Instanzdaten aus verschiedenen Texten des British National Corpus (BNC). Die Daten werden in Form der in den folgenden Punkten beschriebenen Dateien zur Verfügung gestellt[4]:

EnglishLS.test.key Angabe der korrekten Ergebnisse für die Testdatensätze. Wie in Abbildung 5.1 dargestellt, enthält diese Datei Angaben über das Wort („Lexelt“), die Identifikationsangabe der jeweiligen Instanz („Instance“) und die Identifikation der korrekten Bedeutungsvariante („Sense“).

Lexelt	Instance	Sense
activate.v	activate.v.bnc.00008457	38201
activate.v	activate.v.bnc.00061340	38201
activate.v	activate.v.bnc.00061975	38201
activate.v	activate.v.bnc.00062654	38201

Abbildung 5.1: Exzerpt Datei EnglishLS.test.key

EnglishLS.train.key Angabe der korrekten Ergebnisse für die Trainingsdatensätze (vergleichbar mit EnglishLS.test.key).

EnglishLS.answers Angabe der durch das Evaluationssystem ermittelten Resultate (vergleichbar mit EnglishLS.test.key). Zusätzlich wird ein Programm zur Auswertung der Testergebnisse mitgeliefert, *scorer2*. Dieses Programm

[3] Bei diesen wird einfach die Information der korrekten Antwort ignoriert.

[4] Die Datei *EnglishLS.test* wurde hierbei um leichte XML-Formatfehler korrigiert und als *EnglishLS.test.xml* gespeichert. Entsprechend wurde die Datei *EnglishLS.train* in *EnglishLS.train.xml* umbenannt und die Datei *EnglishLS.dictionary* in *EnglishLS.dictionary.xml*.

Listing 5.1: Exzerpt Datei EnglishLS.dictionary.xml

```
<lexelt item="activate.v">
      <sense id="38201" source="ws"
         synset="activate actuate energize start stimulate
            "
         gloss="to initiate action in; make active."
      />
      <sense id="38202" source="ws"
         synset="activate"
         gloss="in chemistry, to make more reactive,
                as by heating."
      />
      ...
    </lexelt>
```

vergleicht die Resultatdatei mit den zu erwartenden Ergebnissen (d.h. mit der Datei EnglishLS.test.key bzw. EnglishLS.train.key, je nachdem welche Daten als Testgrundlage gewählt wurden). Diese Datei wird nicht mit den anderen Dateien ausgeliefert, sondern durch das Evaluationsverfahren ermittelt!

EnglishLS.dictionary.xml Informationen zu den zugrunde gelegten Bedeutungsvarianten aus der WordNet-Datenbank (Ausschnittweise in 5.1 enthalten). Wenn im weiteren Text von Bedeutungsvariantendaten gesprochen wird, sind Datensätze aus dieser Datei gemeint.

EnglishLS.test.xml Texte des BNC, in welchen pro Einheit jeweils ein zu disambiguierendes Wort enthalten ist (Inhalt des „head“-Elementes, wie in Abbildung 5.2 auf der nächsten Seite dargestellt). Diese Datei ist Basis des eigentlichen Tests. Auf einzelne Datensätze dieser Datei wird über die Formulierung „Testinstanzen“ Bezug genommen.

EnglishLS.train.xml Vergleichbar der Datei EnglishLS.test.xml, kann zu Trainingszwecken verwendet werden - äquivalent zur o.a. Bezeichnungsweise werden Datensätze aus dieser Datei „Trainingsinstanzen“ genannt.

Um die Rolle und die Relevanz einer lexikalisch-syntaktischen Vorverarbeitung besser bewerten zu können, stehen fünf Testszenarien mit unterschiedlichem Grad der Vorverarbeitung zu Verfügung. Alle Szenarien basieren auf demselben Basisaufbau, der in Abbildung 5.2 auf der nächsten Seite dargestellt ist:

Die einzelnen Schritte der Versuchsanordnung sind dann wie folgt:

1. Für jedes Szenario wird ein getrennter Testlauf durchgeführt. Zunächst werden die Instanzdaten und die Bedeutungsvarianten lexemweise aus den je-

Listing 5.2: Exzerpt Datei „EnglishLS.test“

```
<lexelt item="activate.v">
   <instance id="activate.v.bnc.00061340" docsrc="BNC">
    <context>
     Avoid fitting one in the kitchen ,
     as fumes from cooking are often enough
     to <head>activate</head> the alarm .
     The one illustrated is by First Alert : like most
     types it can be simply screwed into a ceiling .
    </context>
   </instance>
    ...
  </lexelt>
```

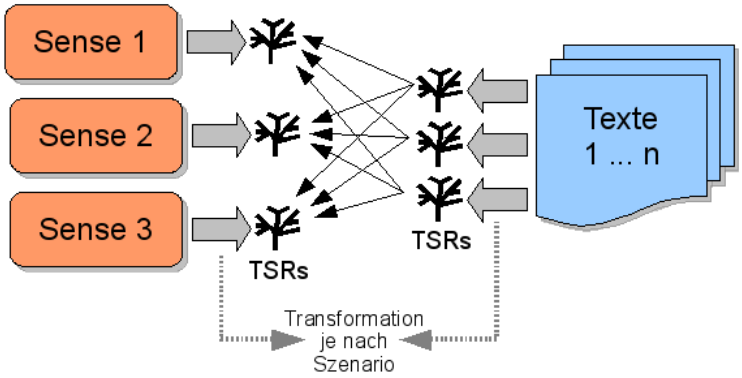

Abbildung 5.2: Basis Setup

weiligen Dateien eingelesen und für jeden Datensatz eine Speicherstruktur erzeugt:

(a) Sowohl der textuelle Inhalt der Bedeutungsvarianten, als auch derjenige der eigentlichen Test-Textinstanzen wird – je nach Vorgabe des jeweiligen Testszenarios – lexikalisch-syntaktisch vorverarbeitet und in genau einen TSR pro Bedeutungsvariante bzw. Textinstanz transformiert.

(b) Um relevante Wortvektoren für den äquivalenten Wortvektor-Versuch pro Szenario zu erhalten, werden die Wortvektoren aus den Termen der in den oben angegebenen Schritten erzeugten TSRs extrahiert, und mithilfe der Werkzeuge des WEKA-Toolkits für maschinelles Lernen (vgl. [Witten & Frank, 2005]) in eine angemessene Matrixdarstellung überführt. Dies führt dazu, daß exakt dieselben Terme zur Weiterverarbeitung genutzt werden, wie im TSR-basierten Verfahren und sich lediglich die interne Datensemantik unterscheidet.

(c) Um vergleichbare Experimentalbedingungen zu schaffen, werden die o.a. TSRs ebenfalls vektorisiert (durch Bildung von Indexvektoren), so daß die folgenden Schritte unabhängig vom Szenario oder Typ (TSR-basiert oder Wortvektorbasiert) durchführbar sind. Damit bestehen auch in der internen Repräsentation der Daten keine Unterschiede mehr zum TSR-basierten Verfahren.

2. Nun wird mit einer Schleife zunächst über die Menge der jeweilig zu untersuchenden Lexeme iteriert und dann in einer inneren Schleife über die Menge der Instanzen, die dem jeweiligen Lexem zugeordnet sind:

   (a) In jedem Iterationsschritt wird die aktuelle Testinstanz-Struktur paarweise mit den Strukturen der dem Lexem zugeordneten Bedeutungsvarianten verglichen, wobei der höchstwertige Vergleich die Auswahl der Bedeutungsvariante bestimmt, wie in Abbildung 5.3 im Pseudocode dargestellt ist[5].

```
function SELECT(TSR[] senses, TSR instance)        ▷ senses : Menge von
Bedeutungsvarianten-TSRs,
                                                   ▷ instance : Textinstanz-TSR
    maxscore = 0
    maxsense                                       ▷ Bestbewertete Bedeutungsvariante
    for all s ∈ senses do
        score =TSR.COSINE(instance, s)
        if score > maxscore then
            maxscore = score
            maxsense = sense
        end if
    end for
    return maxscore, maxsense;
end function
```

Abbildung 5.3: Auswahl der Bedeutungsvariante (Pseudocode)

   (b) Der Bezeichner der ausgewählten Bedeutungsvariante wird zusammen mit dem Bezeichner der aktuell untersuchten Testinstanz in die Ausgabedatei *EnglishLS.answers* zur späteren Berechnung der Meßdaten geschrieben.

3. Die Ausgabe der tatsächlichen Leistungsdaten erfolgt für jeden Testlauf über die Berechnung der Meßwerte Precision und Recall durch das externe,

[5] Der im Algorithmus angegebene Wert *MAXFLOAT* ist der technisch maximale Wert des Datentyps float

durch die Veranstalter von SENSEVAL-3 zur Verfügung gestellte Evaluationswerkzeug „Scorer2“ (Dieses Werkzeug bewertet die Antwortdatei *EnglishLS.answers* zeilenweise gegen die Wertevorgaben aus der Datei *EnglishLS.test.key*). Diese Vorgehensweise wurde gewählt, um interne Fehlerquellen der Evaluation weitestmöglich auszuschließen und hierin den Vorgaben von SENSEVAL-3 zu folgen.

Im Rahmen der o.a. Versuchsanordnung wurden die folgenden 5 Szenarien evaluiert:

Volltext Dieses Szenario dient als grundlegendes Setup ohne besondere Vorverarbeitung. Sämtliche Terme aus den jeweiligen Testinstanzen werden über eine einmalige Anwendung der OR-Operation in einen TSR (bzw. in einen Wortvektor) überführt, wobei lediglich sog. Stopwörter entfernt werden. Diese Variante besitzt den größten Informationsumfang, aber ebenso auch das größte, der Berechnung zugrundeliegende Datenvolumen. Dem Szenario liegt die Erwartungshaltung zugrunde, daß zwar die Effektivität der Verfahren hiermit maximal ist, aber die Effizienz nur unterdurchschnittlich.

Nomen Grundlage der Verarbeitung sind hierbei lediglich die in den Bedeutungsvarianten und den Instanztexten vorhandenen Nomen. TSRs und Wortvektoren werden – jenseits dieser Filterung anderer Wortarten – gebildet wie im Volltext-Szenario. Erwartungsgrundlage ist, daß Nomen den größten semantischen Beitrag leisten, da viele Verfahren in Bezug auf Nomen offenbar die beste Leistung zeigen (vgl. [Mihalcea & Faruque, 2004, Artiles et al., 2004, Pedersen, 2004], etc.).

DNF Hierbei werden Bedeutungsvarianten und Instanztexte phrasenstrukturell (vgl. [Abney, 1992]) in „chunks“ (Textfragmenten) untersucht, und infolgedessen TSRs aus *DNF* Formen (vgl. Abschnitt 4.1.3.3) - wie in Beispiel 4.15 auf Seite 122 angegeben - erstellt, wobei aber lediglich die Sätze, die das zu untersuchende Lexem enthalten, verwendet werden. Anhand eines kleinen Beispieles soll dieses Vorgehen verdeutlicht werden:

1. Text der Instanz (Lexem ist *activate*, ID ist „activate.v.bnc.00379850“):

   „Precisely . That 's what I do and that 's why I hide . I know how to effectively program the software that <*head*>**activates**</*head*> the red Goddess . Tammuz you are mad . What am I doing here in bed with you ?“

2. Fokussierung auf den Kernsatz, d.h. den Satz, der das disambiguierende Wort enthält:

   „I know how to effectively program the software that <*head*>**activates**</*head*> the red Goddess“

3. Konstruktion der DNF-Funktionsform (Junktoren werden hervorgehoben dargestellt) anhand der chunks:

   (know) OR (effectively AND program) OR (software) OR (**activates**) OR (red AND Goddess)

Dieses Szenario wurde durch die Vermutung motiviert, eine bessere oder weitgehend gleichbleibende Leistung bei geringerem Text-Datenvolumen zu erhalten – wobei die konkrete Datensituation durch flache syntaktische Merkmale gesteuert werden soll.

CNF Das *CNF*-Szenario ist nahezu identisch zum *DNF*-Szenario, lediglich werden konjunktive Normalformen anstelle von disjunktiven Normalformen aus den Testtexten konstruiert. Die hierbei zugrundeliegende Motivation ist ebenfalls vergleichbar mit derjenigen, die zum Entwurf des DNF-Szenarios führte.

Schließlich werden die Leistungsdaten von TSRs und Wortvektoren miteinander verglichen - die Darstellung und Diskussion dieser Ergebnisse sind Inhalt des folgenden Abschnittes.

5.3 Resultate und Diskussion

Die wesentlichen Ergebnisse der Auswertung der in Abschnitt 5.2 vorgestellten Szenarien liefert Tabelle 5.2[6]. Im Durchschnitt erbringen die Wortvektor-basierten

Szenario	**Wortvektoren**	**TSRs (Indexvektoren)**
DNF	23%	30%
CNF	19%	27%
Nomen	19%	23%
Volltext	29%	34%
Micro-average K	20%	20%

Tabelle 5.2: Ergebnisse der Experimente (jeweils 3944 Instanzen)

Systeme demnach lediglich 79% der entsprechenden TSR-basierten Systeme, wobei die Proportionen zwischen 70% (CNF) und 85% (Volltext) liegen und keines der Wortvektorsysteme besser ist als das entsprechende TSR System. Die Verwendung von TSRs zeigt also in dem in Abschnitt 5.1 gesetzten Rahmen eine bessere

[6]Sämtliche Experimente besitzen eine *Coverage* von 100%, d.h. die Testdatensätze werden vollständig verarbeitet.Dies führt zu einer zahlenmäßigen Übereinstimmung von precision und recall, daher wird in der Tabelle nur ein Wert notiert. Aufgrund eines Fehlers des scorer-Programms ist außerdem lediglich die Auswertung des *Fine*(*Micro*)-Wertes möglich. Zum Vergleich wird in der letzten Tabellenzeile zudem noch einmal der Wert für das „Agreement by chance“ aufgeführt.

Leistung bezüglich precision und recall als entsprechende, Wortvektor-basierte Ansätze. Zur weiteren Prüfung wurden die eigentlichen SENSEVAL-3 Trainingsdaten ebenfalls dem DNF-Szenario unterworfen: hier zeigte sich ein Ergebnis von 31,1% Precision für den TSR-basierten Ansatz und 23,9% Precision für den Wortvektoransatz – bei insgesamt 7860 Testinstanzen.

Tabelle 5.4 beinhaltet über die oben angeführten Resultate hinaus eine Aufschlüsselung der Werte nach der Wortart, wobei die Verteilung der Wortarten auf das Testkorpus in Tabelle 5.3 angegeben wird[7]. Die Ergebnisse nach Wortarten sind

Wortart	**Anzahl (prozentual)**	**Anzahl (absolut)**
Nomen	45,82%	1807
Verben	50,15%	1978
Adjektive	4,03%	159
Summe	*100%*	*3944*

Tabelle 5.3: Verteilung der Textinstanzen nach Wortarten

hierbei wie folgt zu lesen:

Wortart	**Wortvektoren**		**TSR Indexvektoren**	
	Precision	**Recall**	**Precision**	**Recall**
Nomen	29,7%	13,6%	34,4%	15,7%
	(536)		(621)	
Verben	29,0%	14,6%	31,7%	15,9%
	(574)		(627)	
Adjektive	23,3%	0,9%	48,0%	2,0%
	(37)		(77)	
Ergebnis	*29,1%*	*29,1%*	*33,6%*	*33,6%*
	(1147)		*(1325)*	

Tabelle 5.4: Ergebnisse nach Wortarten-Information

1. Die Spaltenwerte für *recall* addieren sich zum Gesamtergebnis für recall, da der recall-Meßwert die Vollständigkeit der korrekt ermittelten Instanzen bemißt[8] (vgl. Definition auf Seite 58). Dieser Sachverhalt läßt sich auch formelhaft wie folgt darstellen::

   Wortvektoren:

$$\frac{(a_{Nomen} + a_{Verben} + a_{Adjektive})}{(a_{Gesamt} + c_{Gesamt})} = \frac{(536 + 574 + 37)}{(1147 + 2797)} = \sim 0,291$$

[7] Diese Ergebnisse wurden bereits 2008 in [Winnemöller, 2008] publiziert

[8] Die gerundeten prozentualen Verteilungen stehen hierbei über den in Klammern angegebenen absoluten Anzahlen der Textinstanzen – dies gilt auch für die Angabe der *precision*-Werte.

TSR-Indexvektoren:

$$\frac{(a_{Nomen}+a_{Verben}+a_{Adjektive})}{(a_{Gesamt}+c_{Gesamt})}=\frac{(621+627+159)}{(1325+2619)}=\sim 0,336$$

2. Die Resutate der Spalte *precision* ergeben sich entsprechend nach dem *precision*-Wert (vgl. Definition auf Seite 58):

   Wortvektoren:

   $$\frac{(a_{Nomen}+a_{Verben}+a_{Adjektive})}{(a_{Gesamt}+b_{Gesamt})}=\frac{(536+574+37)}{(1147+2797)}=\sim 0,291$$

   wobei gilt:

   $$\frac{a_{Nomen}}{(a_{Nomen}+b_{Nomen)}}=\frac{536}{(536+1269)}=\sim 0,297$$

   $$\frac{a_{Verben}}{(a_{Verben}+b_{Verben)}}=\frac{574}{(574+1406)}=\sim 0,29$$

   $$\frac{a_{Adjektive}}{(a_{Adjektive}+b_{Adjektive)}}=\frac{37}{(37+122)}=\sim 0,233$$

   TSR-Indexvektoren:

   $$\frac{(a_{Nomen}+a_{Verben}+a_{Adjektive})}{(a_{Gesamt}+b_{Gesamt})}=\frac{(621+627+159)}{(1325+2619)}=\sim 0,336$$

   wobei gilt:

   $$\frac{a_{Nomen}}{(a_{Nomen}+b_{Nomen)}}=\frac{621}{(621+1184)}=\sim 0,344$$

   $$\frac{a_{Verben}}{(a_{Verben}+b_{Verben)}}=\frac{627}{(627+1351)}=\sim 0,317$$

   $$\frac{a_{Adjektive}}{(a_{Adjektive}+b_{Adjektive)}}=\frac{77}{(77+844)}=\sim 0,48$$

Das TSR-Volltext System erbringt zudem bessere Resultate als die beiden (letztplazierten) SENSEVAL-3 Systeme *DFA-LS-Unsup* und *DLSI-UA-LS-NOSU*. Insgesamt liegen die Ergebnisse der TSR Ansätze zudem signifikant höher als der bezüglich SENSEVAL-3 angegebene entsprechende Wert für Agreement by Chance.

Wie zu vermuten war, schneiden die meisten SENSEVAL-3 Systeme im Ergebnis besser ab als der reine TSR-Ansatz. Allerdings handelt es sich bei diesen Systemen in der Regel um Hybridsysteme im Sinne von Abschnitt 2.2.4, d.h. es werden kombinierte Methoden und externe Daten (in der Majorität Lesk-ähnliche bzw. MRD basierte Algorithmen über WordNet-Daten) durch die implementierten Systeme

genutzt. Derartige Kombinationssysteme erzielen meistens bessere Ergebnisse als „single-Idea“ Systeme (vgl. hierzu Abschnitt 2.2.4), eine Zuordnung der Leistung bzw. des Erfolges einzelner Komponenten ist hierbei aber sehr viel schwieriger: welche Komponente in welcher Weise zum Ergebnis beiträgt und wie sich das Gesamtsystem in Bezug zur Einzelkomponente verhält sind weitgehend ungeklärte Fragen.

Da die Eingabewerte des Evaluationssystemes lediglich in der Herkunft der Vektordaten differieren (TSR-Indexvektoren versus Wortvektoren), und zudem das offizielle SENSEVAL-3 Evaluationsprogramm *scorer* verwendet wurde (um Seiteneffekte und mögliche Programmierfehler bei der eigentlichen Auswertung auszuschließen) kann von der Gültigkeit der Ergebnisse ausgegangen werden.

Zwar ist anzunehmen, daß die Performanzsteigerung im Zusammenhang mit Ensemble-Methoden für den TSR-Ansatz anders ist als für die wortvektorbasierten Ansätze, die Untersuchung dieser Phänomene übersteigt jedoch den Rahmen der vorliegenden Arbeit.

Zum realen Zeitverhalten ist schließlich anzumerken, daß die TSR basierten DNF und CNF Experimente auf einem Pentium4 (2,7 GHz) Notebook mit 512 MB Hauptspeicher unter Verwendung der Java VM 1.4 jeweils in ca. 45 Minuten durchliefen und das Volltext-Experiment etwa die zehnfache Zeit benötigte.

5.4 Zwischenergebnis

Die in diesem Kapitel vorgestellten Experimente zeigen die Effektivität der TSR--basierten WSD-Lösung gegenüber einem konventionellen wortvektorbasierten Verfahren im Rahmen eines anerkannten Evaluationsschemas. Hierzu wurde der im vorherigen Kapitel vorgestellte Prototyp verwendet, der als Referenzimplementation des TSR-Ansatzes dient.

Gegenüber den SENSEVAL-3 Teilnehmersystemen belegt das TSR-basierte Verfahren zwar nur den 8. Platz (von insgesamt 11), allerdings ist dieses das einzige Nicht-Hybridsystem. Es ist anzunehmen, daß sich das TSR Verfahren durch die strukturelle Ähnlichkeit mit konventionellen Wortvektoren gut mit weiteren Methoden - wie in SENSEVAL-3 verwendet - kombinieren läßt, wodurch für die hieraus entstehenden Hybridsysteme eine insgesamt bessere Leistung zu erwarten ist.

Kapitel 6

Schluss

In diesem Kapitel soll zunächst im Fazit eine kritische Betrachtung der erarbeiteten Inhalte erfolgen (Abschnitt 6.1). Abschließend werden in Abschnitt 6.2 mögliche zukünftige Entwicklungspfade und -optionen aufgezeigt. Da der TSR-Ansatz erst am Anfang seiner Entwicklung steht, soll der Ausblick nicht nur einige unmittelbare Konsequenzen der möglichen Entwicklung aufzeigen, sondern einen breiteren Überblick über mögliche Richtungsoptionen bieten.

6.1 Fazit

In dieser Arbeit wurde ein Vorschlag für eine Betrachtungsweise bestimmter Aspekte der Textbedeutung präsentiert, die sich aus der Theorie der Familienähnlichkeit zwischen bestimmten Wörtern, der sich hieraus ergebenden Kategorienbildung und dem Gebrauch der Wörter ergibt. Diese Aspekte verbinden semantische und pragmatische Aspekte durch ihren Bezug zum Weltwissen.

Zusätzlich wurde eine effektive und effiziente Methodik entwickelt, diese Betrachtungsweise der Textbedeutung durch eine praktisch orientierte komplexe Repräsentationssystematik zu unterstützen - dem sogenannten TSR-Ansatz. Dieser liefert nicht nur einen Bezug zum genannten theoretischen Hintergrund, sondern bietet auch praktische Anwendungsmöglichkeiten im Bereich der automatischen Textverarbeitung, insbesondere mit Bezug zur WSD-Problematik.

Die zentralen Objekte des TSR-Ansatzes sind die sog. TSRs, spezielle Baumstrukturen, die miteinander verglichen und über verschiedene Operationen kombiniert und manipuliert werden können. Eine Menge von initialen TSRs wird aus einem Internet-Verzeichnis erzeugt, es können aber zudem weitere TSRs aus bereits verfügbaren TSRs abgeleitet werden, um beispielsweise bisher unbekannte Wörter repräsentieren zu können, die Repräsentation bekannter Wörter zu verbessern oder um die Granularitätsebene der Repräsentation zu wechseln (z.B. um Sätze darstellen zu können). Für TSRs sind verschiedene Operationen und Funktionen definiert

(und realisiert), mit denen sie analysiert und manipuliert werden können, außerdem bestehen Möglichkeiten, TSR-Daten in bestehende vektorbasierte Systeme zu integrieren - damit wurde eine Kombination bisheriger Verfahrensparadigmen erreicht, die deren jeweilige Vorteile (Robustheit aufgrund eines hohen Hintergrundinformationsvolumens und wissensbasierte, regelhafte Datenverarbeitung) in hohem Maße beibehält. Durch das Offenlegen der Ergebnisse semantisch-pragmatischer Datenmanipulation über die TSR-Baumstrukturen und aus dem Systemverhalten angesichts der verschiedenen Testszenarien sind ferner Schlüsse auf semantische Sprachvorgänge möglich, die ohne den Einsatz der TSR-Methodik bisher kaum denkbar wären.

Einen Einblick in die tatsächliche Wirksamkeit des TSR-Ansatzes wurde über ein prototypisches System auf Basis eines relationalen Datenbanksystemes in einem Standard-Experiment im Bereich der nicht-überwachten Auflösung von Wortmehrdeutigkeiten gewonnen. Hier zeigte ein TSR-basiertes System eine signifikant bessere Performanz gegenüber der herkömmlichen, Wortvektorraum-basierten Methodik – zwar hätte der Prototyp unter den in SENSEVAL-3 getesteten Systemen nur eine Positionierung im unteren Bereich erreicht; dies ist aber angesichts des Vorteils, den die SENSEVAL-3 Systeme durch ihre Hybridstruktur erhalten, erklärbar. Für die Erfolgsmessung wurden die für die WSD-Forschung üblichen Meßzahlen *precision* und *recall* genutzt. Die Verwendung in Experimenten mit andersgearterter Zielrichtung (z.B. Sprachenerkennung) zeigt die Generalisierbarkeit der Grundsätze des TSR-Ansatz über die Anwendung im WSD-Rahmen hinaus.

Ergänzend lassen sich einige prominente Eigenschaften des TSR-Ansatzes feststellen, die in den folgenden Punkten diskutiert werden (eine Wiederholung der kapitelbezogenen Zwischenergebnisse soll aber unterbleiben):

1. TSRs bieten gegenüber der Darstellung durch rein konventionelle Methoden eine bessere Leistung in der Auflösung von Wortmehrdeutigkeiten. Dies gilt besonders in Situationen, in denen nur sehr wenige Textdaten verfügbar sind bzw. nur sehr wenige Informationen aus den verfügbaren Texten gewonnen werden können. Dadurch, daß TSRs über die Verwendung von internetbasierten Textdaten eine *große Menge* hierarchisch organisierter Daten aufnehmen und *einem* bestimmten Term oder Textfragment assoziativ zur Seite stellen können (vgl. Abschnitt 4.1.2), sind sie auch in der Lage, in relativ hohem Maße weitere Informationen, z.B. Verarbeitungsinformationen anderer Verfahren zu speichern, selbst zu nutzen und weiteren Komponenten zur Verfügung zu stellen, wie bereits in der Einleitung gefordert wird. Damit wird das Informationsverhältnis zwischen Term und Text effektiv umgekehrt: nun müssen Algorithmen im Idealfall nicht länger große Textkorpora verarbeiten, um bestimmte Schlüsse ziehen zu können, sondern es genügt in diesen Fällen die dezidierte Analyse des Verhaltens der beteiligten TSRs.

2. TSRs wirken durch die direkte Anwendbarkeit des Kompositionalitätsprinzipes auf einer beliebigen textuellen Granularitätsebene: es gibt Wort-TSRs,

Chunk-TSRs, Satz-TSRs, Text-TSRs, Fragment-TSRs, etc. Dieser Wechsel der Ebene kann durch verschiedene Operationen (OR, AND, vgl. Abschnitte 4.1.3.3 auf Seite 120 und 4.1.3.3 auf Seite 119) gesteuert werden. In Abschnitt 4.1.3.1 auf Seite 113 wird aber gezeigt, daß ein Wechsel der Granularitätsebene über die TSR-Methodik nicht in allen Anwendungsbereichen sinnvoll ist, bzw. nur einen geringen Effekt zeigt. Der Vorteil dieses Aspektes liegt darin, daß somit Objekte verschiedener Granularität miteinander verglichen werden können, z.B. Wörter mit Texten – hiervon könnte beispielsweise eine Suchmaschine profitieren.

3. Über eine Transformation in eine Vektordarstellung (vgl. Abschnitt 4.1.1) bestehen gute Integrationsmöglichkeiten in bestehende Verfahren. Hierdurch können TSRs auch in vielen anderen Bereichen der automatischen Textverarbeitung eingesetzt werden, in denen üblicherweise Wortvektoren zum Einsatz kommen. Dieser Vorteil wird allerdings mit dem Verlust der Hierarchieinformation erkauft, die durch die flache Vektor-Repräsentation verloren geht, auch wenn die Operationen weiterhin verfügbar bleiben (da die Vektorelemente die Knotenreferenz beinhalten, bleibt dieser Anteil der Strukturinformation erhalten, lediglich die Oberflächenparameter verschwinden; vgl. 4.1.1 auf Seite 99).

4. Die in einem Text verwendete Sprache kann mit hoher Wahrscheinlichkeit festgestellt werden (vgl. Abbildung 4.12 auf Seite 115) - das dieser Funktionalität zugrundeliegende Prinzip kann unter Ausnutzung lokaler Information innerhalb des ODP generalisiert werden: immer dort, wo sich Domänenwissen im ODP konzentriert, läßt sich dieses Domänenwissen explizit über die Auswertung von TSR-Teilbäumen nutzen. Diese Behauptung wurde bisher aber nur durch das besagte Sprachenerkennungs-Experiment gestützt.

5. Nicht zuletzt ist die Erstellung der TSRs durch das Recycling von Internet-Datenbasen (vgl. Abschnitt 4.1.2) vergleichsweise effizient, da der eigentliche Aufwand einer semantischen Vorverarbeitung bereits von Dritten (zu einem Zweck, der jenseits der TSR-Methodik liegt) betrieben wurde. Der Nachweis einer Gütebewertung der ODP-Datenbasis in Relation zu anderen möglichen Datengrundlagen wurde bisher aber noch nicht erbracht, daher besteht eine gewisse Wahrscheinlichkeit, daß andere Datenbasen besser geeignet sind, als das verwendete Internetverzeichnis.

6. Die Verwendung einer großen Datenbank für wissensbasierte Verarbeitung bringt immer den Nachteil eines gewissen Zeit- und Rechenaufwandes mit sich. In Abschnitt 4.1.3.3 auf Seite 124 wurde die Möglichkeit erörtert, diesen Aufwand durch verschiedene Kürzungsoperationen zu minimieren, allerdings um den möglichen Preis einer Leistungseinbuße. Ob diese Option akzeptabel ist, hängt ultimativ von der realen Anwendungsumgebung ab. Wie

oben angeführt, sind bestimmte Anwendungen dennoch realistisch online zu betreiben.

7. Inferenz- und Deduktionsmechanismen sind nicht unmittelbarer Bestandteil der TSR-Methodik, weil die theoretische Grundlage der TSRs dies nicht beinhaltet (vgl. Abschnitt 3.1). Hierdurch treffen viele Nachteile flacher Verfahren ebenfalls auf den TSR-Ansatz zu (z.B. Ignoranz der Negation - damit wird ein Sachverhalt in der Regel nicht von dem negierten Sachverhalt unterschieden). Diese Eigenschaft kann jedoch vermutlich durch die Kombination mit anderen Methoden im Rahmen eines Ensemble-Verfahrens abgeschwächt werden, oder durch eine assoziative Integration flacher Inferenzmechanismen.

8. Aufgrund des vagen Charakters der TSRs können diese zwar robust eingesetzt werden, es besteht jedoch immer eine gewisse Ungenauigkeit der Ergebnisse. Zwar könnte u.U. die Qualität der Ausgangsdaten (d.h. des verwendeten Internet-Verzeichnisses) verbessert werden, dies könnte aber ebenso zu einer ungewollten Gewichtung der Bedeutungsinterpretation führen.

Die oben angeführten Punkte stellen sicherlich keine erschöpfende Diskussion der Ergebnisse und Möglichkeiten des TSR-Ansatzes dar, es handelt sich eher um wesentliche Feststellungen, die sich aus der in der Einleitung vorgegebenen Zielsetzung der vorliegenden Arbeit ergeben.

Die in der Einleitung formulierten Anforderungen an den TSR-Ansatz wurden in ihren wesentlichen Grundzügen erfüllt.

6.2 Ausblick

Die TSR-Methodik ist in ihrer theoretischen und praktischen Ausprägung noch vergleichsweise neu, erste Ansätze hierzu wurden in [Winnemöller, 2001] vorgestellt, verschiedene konkrete Thesen wurden aber erst 2004 in [Winnemöller, 2004] publiziert. Aufgrund dieser Novität gibt es eine große Vielfalt denkbarer zukünftiger Vorgehensweisen, Erweiterungen und Verbesserungen. Einige mögliche Richtungen sollen in diesem Abschnitt vorgestellt werden.

Wünschenswert wäre beispielsweise eine Reihe von Erweiterungen des ursprünglichen TSR-Ansatzes, wie in den folgenden Punkten exemplarisch aufgeführt:

- Denkbar wäre eine Nutzung des sog. *Stemming*, also des Rückführens von Wortformen auf eine (hypothetische) Grundform (vgl. [Porter, 1980]) – damit wäre zusätzlich die Speicherung und Nutzung von Wortarten-Informationen im TSR möglich. Der Vorteil dieser Variante wäre zweifach: einerseits würde der Gebrauch vieler Wortformen zusammengeführt werden,

was eine konsistentere – und damit vermutlich effektivere – Strukturierung der Grundform-TSRs bewirken würde und andererseits wäre eine grundlegende Unterscheidung nach PoS-Information möglich, so daß Bedeutungsvarianten mit derart grober Unterscheidbarkeit auch nicht zusammengeführt würden.

- Eine syntaxgesteuerte Anwendung von TSR-Operationen wäre über ein musterbasiertes Verfahren denkbar, z.B. derart, daß bestimmte syntaktische Strukturen vorgegebene Abfolgen von TSR-Operationen auslösen. Derzeit kommt durch Realisierung der DNF- bzw. CNF-Szenarien lediglich eine einfache und flache Variante dieser Idee zur Anwendung, jedoch würden elaboriertere und situierte Syntaxmuster vermutlich bessere Ergebnisse zeigen. Hierbei stellt sich jedoch in erhöhtem Maße die Frage, wie TSR-Operationen und lexikalisch-syntaktische Textmerkmale zusammenhängen.

- Umgekehrt ist eine Kopplung von Syntaxmustern mit TSRs denkbar, um TSRs regelbasiert für Parsing einzusetzen - ähnlich des Prinzips der *Tree Adjoining Grammars* (*TAG*s, vgl. [Abeille, 1988]). Dies könnte beispielsweise erfolgen, indem Referenzen auf TAGs, die in einer gesonderten Datenbasis gespeichert sind, in einem speziellen Arm der TSR-Bäume gespeichert werden, wie in Abbildung 6.1 auf der nächsten Seite beispielhaft dargestellt wird. Hierbei würde der TSR-Datenraum um syntax-relevante Pfade erweitert. Beispielsweise könnten Pfade wie */top/syntax/pos/jj* und */top/syntax/chunk/np* eingefügt werden, die den lokalen syntaktischen und lexikalischen Kontext des TSRs beschreiben. Damit würden sich TSR-Instanzen auch bezüglich ihres syntaktischen Kontextes unterscheiden lassen. Kombinationen von TSRs würden dann flache syntaktisch-lexikalische Informationen enthalten, die z.B. im WSD-Umfeld genutzt werden könnten. Denkbar wäre damit auch eine Unterstützung von Parsing-Verfahren, etc., wobei der Vorteil einer solchen Vorgehensweise in der uniformen Repräsentation von Sprach- und Weltwissen liegen würde. Es hinge in diesem Fall lediglich von der Nutzung der Daten ab, inwiefern ggf. Verbindungen zwischen diesen Bereichen explizit hergestellt werden, etwa in Form von ebenenübergreifenden Regeln (z.B. in der Art von „*Wenn X im Gebrauchszusammenhang zu Y steht und als Nominalform vorliegt, dann tue dieses oder jenes*“). Ein geeigneter Wertebereich für die Baumknoten müßte in diesem Falle allerdings noch ermittelt werden.

  Wie auf Seite 156 bereits angedeutet, könnten auch ganz allgemein Objekt-Referenzen im TSR-Baum gespeichert werden. Die referenzierten Objekte können verschiedener Art sein:

  1. Sprachliche Objekte: Beispielsweise können die bereits erwähnten TAGs referenziert werden und damit zur Struktur des TSRs beitragen. In ähnlicher Weise könnten auch lexikalische Informationen in-

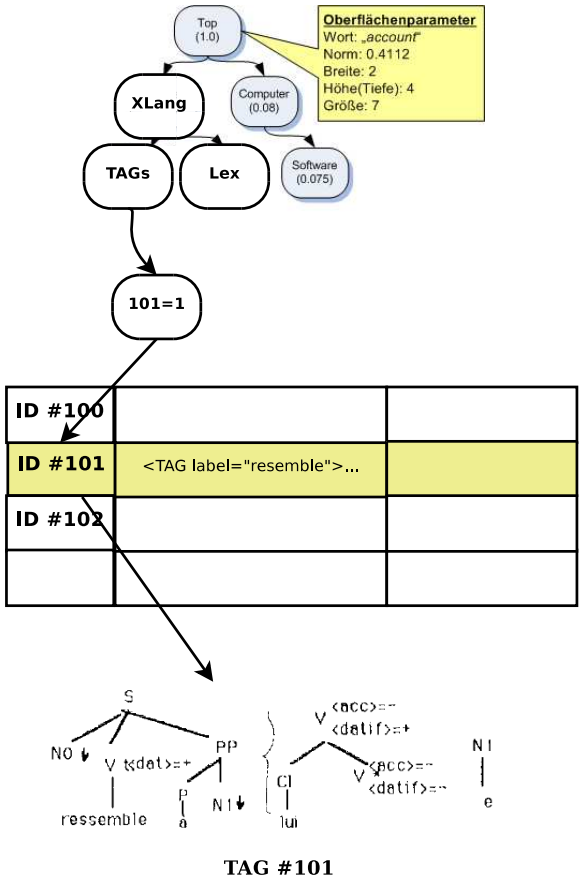

Abbildung 6.1: Einbettung von TAG-Referenzen

tegriert werden, z.B. Wortarteninformationen. Der folgende hypothetische Pfad eines TSRs für „Horse“ wäre ein Beispiel hierfür:

TSR(„Horse“) : */ Top / XLang / Lex / POS / NN*= 1.0

2. Ähnlich könnten weitere Referenzen von Objekten anderer Wissensdomänen gespeichert werden, z.B. wäre die Speicherung visueller Primitive eine denkbare Möglichkeit, um in TSR-Bäumen auch Bildinformationen aufzunehmen – beispielsweise so (Die Semantik der numerischen Information bleibt hierbei offen):

TSR(„Sun“) : */ Top / XVisPrim / Yellow / Bright / Circle*= 0.5

Insgesamt könnte eine derartige lose Informations-Kopplung „höhere“ Anwendungen direkt unterstützen; für die Anwendung des automatischen Erstellens von Zusammenfassungen (engl. *Summarization*) soll daher an dieser Stelle ein Beispielszenario zur Verdeutlichung angegeben werden: Ein wissenschaftlicher Mitarbeiter möchte eine Literaturdatenbank mit wissenschaftlichen Artikeln nach Quellenmaterial zu einem bestimmten Thema

durchsuchen. Leider erhält er auf seine Anfrage sehr viele Antworten, möchte aber den Suchbereich nicht einschränken. Obwohl die Möglichkeit besteht, inhaltliche Übersichten der gefundenen Artikel erzeugen zu lassen, sind diese „Zusammenfassungen“ nicht sehr hilfreich, da sich der Inhalt häufig nicht aus diesen Exzerpten erschließen läßt. Eine weitere Möglichkeit, fremdsprachliche Artikel in seine Muttersprache übersetzen zu lassen, produziert ebenfalls überwiegend unbrauchbare Ergebnisse.

Hier könnte über den TSR-Ansatz ein Kürzen der ursprünglichen Texte auf eine überschaubare Menge von relevanten Exzerpten erfolgen, wobei die Einschätzung der „Relevanz“ aus der TSR-Bewertung einzelner Sätze in Bezug auf die Suchanfrage resultiert. Der Zusammenhang zu ähnlichen Artikeln in anderen Sprachen könnte durch denselben Mechanismus, aber unter Einbeziehung von Spracheninformationen in den TSRs – wie beispielsweise in Abschnitt 4.1.3.1 exemplarisch demonstriert wird – hergestellt werden.

- Der Einsatz einer internen Suchmaschine zum Auffinden „verwandter“ TSRs könnte eine andere Verwendungsweise bewirken: nicht nur die Ähnlichkeit bekannter TSRs könnte festgestellt werden, sondern es könnten auch TSRs mit einem bestimmten Grad an Ähnlichkeit in der Datenbank gesucht und gefunden werden. Ein solches Verfahren könnte genutzt werden, um bestehende Verfahren um eine robustere Methodik zu ergänzen, beispielsweise um in Suchmaschinen nach Begriffen suchen zu können, die nicht in einem direkten Bedeutungszusammenhang stehen (z.B. als Synonyme oder Antonyme), aber ähnlich gebraucht werden. Hierzu müßten die TSRs vektorisiert und die entstehenden Vektoren in die Index-Datenbank einer Suchmaschinen-Engine (z.B. Apache Lucene, vgl. [Bialecki et al., 2008]) eingefügt werden.

  Moderne Suchmaschinen nutzen vielfältige Heuristiken und statistische Methoden um möglichst präzise, aber umfassende Ergebnisse für beliebige Suchanfragen zu erbringen - aber dennoch werden häufig sehr große Mengen irrelevanter Antworten produziert. TSRs könnten helfen, die Menge der Antworten weiter zu reduzieren. In entgegengesetzter Weise muß das Wissen, welches vielen (regelbasierten) eMail-Spamfiltern zugrunde liegt, oft aufwändig von Hand eingepflegt werden. Trotz dieses Aufwandes arbeiten viele derartiger Filter noch mit einer vergleichsweise hohen Fehlerquote. An dieser Stelle könnten TSRs eingesetzt werden, um bestimmte Inhalte und typische Formulierungen robust zu filtern (dies betrifft nur die semantischen Filter – Filter, die auf lexikalischen Muster u.ä. basieren, können über die TSR-Methodik derzeit nicht realisiert werden).

- Die Integration lexikalischer und ontologischer Datenbasen wäre sinnvoll, z.B. von WordNet, um manuell erzeugte Assoziationen von inhärent besserer Qualität (und damit deutlicherer Aussagekraft) nutzen zu können. Die Internetverzeichnis-basierten Baumstrukturen lassen sich vergleichswei-

se einfach um weitere Wissensquellen erweitern, z.B. könnten lexikalische Datenbanken wie WordNet eingebunden werden, um ontologisches Wissen ebenso repräsentieren und nutzen zu können. Zudem wäre es möglich, den Kanten zwischen den TSR-Knoten einen „Typ“ zuzuweisen, der beispielsweise eine ontologische oder lexikalische Dimension spiegelt - dies würde eine zusätzliche intensionale Sichtweise von Textbedeutung in den TSR-Ansatz integrieren[1].

Prinzipiell ist auch eine Vielzahl von funktionalen Erweiterungen der TSR-Methodik denkbar. Diese müssen zwar Gegenstand zukünftiger Forschung bleiben, an dieser Stelle soll aber dennoch eine Variante beispielhaft genannt werden:

Hierbei läßt sich die Tatsache, daß einige Begriffe von anderen „dominiert“ werden, durch TSRs abbilden indem die bekannten Operationen OR und AND angemessen modifiziert werden. Zum Beispiel wird in der Phrase „der kurzhaarige Hund“ das Wort „kurzhaarig“ von dem Wort „Hund“ dominiert, bzw. das Wort „Hund“ von „kurzhaarig“ modifiziert (Welcher Begriff hierbei der dominante ist, wird in diesem Fall von der lexikalischen Ausprägung definiert – Substantive dominieren Adjektive). Für eine dominierte Variante der OR-Operation gilt entsprechend: die Knoten werden kombiniert, wie für die OR-Operation definiert, im Falle gleicher Knoten im TSR wird aber der Wert des „dominanten“ TSRs verwendet. Ein möglicher rekursiver Algorithmus, der auf TSR-Bäumen anstelle von TSR-Indexvektoren arbeitet, wird in Abbildung 6.3 auf der nächsten Seite vorgestellt.

Abbildung 6.2 auf der nächsten Seite exemplifiziert diesen Algorithmus - der Knoten mit der Nummer 35 zeigt hierbei die Wirkung der Dominanz.

Auch die AND-Operation läßt sich durch eine dominierte Variante ergänzen. Die Dominanz-Operationen sollen bewirken, daß bei Vergleichen zwischen TSRs die Dominanz sichtbar wird. Damit würde beispielsweise der dominierte Vergleich von „kurzhaarige Hund“ und „Hund“ eine viel geringere Distanz als der undominierte Vergleich zeigen – der diesbezügliche Nachweis muß jedoch in zukünftigen Experimenten erbracht werden.

Über weitere mögliche Operationen könnte eine Berechnung der Mittelwerte von Knoten erfolgen, oder bestimmte Operationen könnten ähnlich der von Widdows vorgeschlagenen Quantenlogik formuliert werden (vgl. [Widdows & Peters, 2003]). Hiermit wäre eine Erhöhung der Aussagefähigkeit verbunden, deren Eigenschaften jedoch noch zu bestimmen wären.

Abschließend läßt sich feststellen, daß die TSR-Methodik einen neuartigen und viel versprechenden Ansatz zur bedeutungsbezogenen Textverarbeitung darstellt, der eine Vielfalt von Einsatzmöglichkeiten bietet. Diese gilt es in Zukunft weiter zu erforschen.

[1] Dieser Gedanke wurde bereits anhand einiger externer Projekte in Abschnitt 3.4 diskutiert

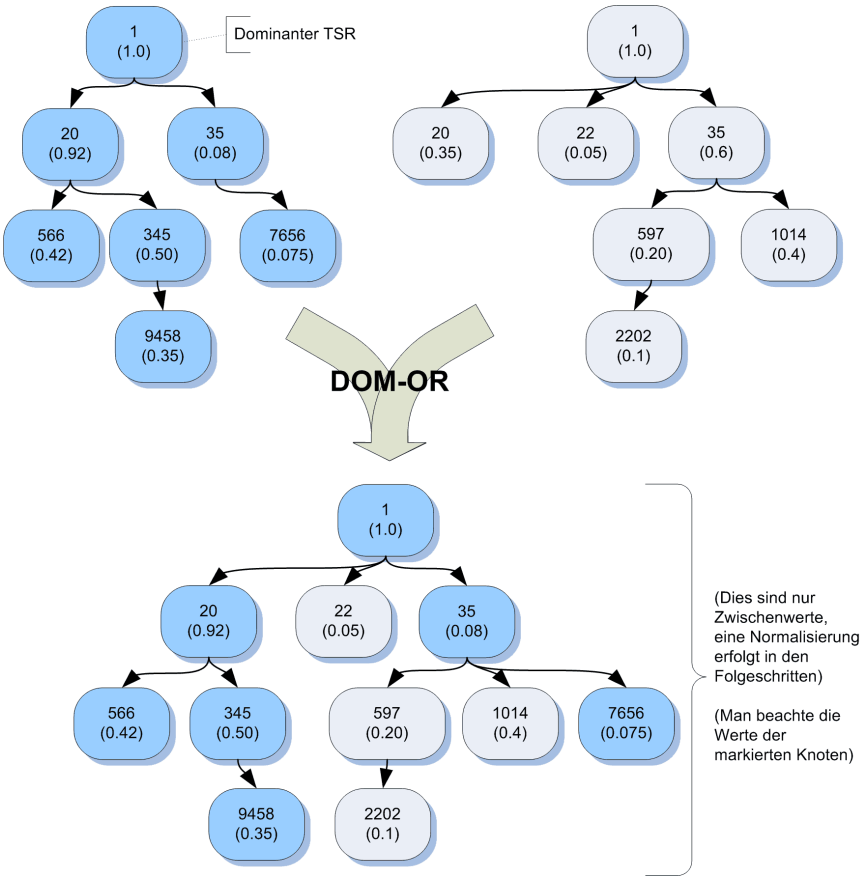

Abbildung 6.2: Beispiel „dominierte Zusammenführung“

function TSR.DOMOR($TSRa, TSRb$)
  for all $v_a \in V_a, v_b \in V_b, v_a = v_b$ **do** ▷ V_a : Knotenmenge von $TSRa$
    $x_a = f(v_a)$ ▷ Knotenwert von v_a
    $x_b = f(v_b)$ ▷ Knotenwert von v_b
    if $x_b \neq 0$ **then**
      $x_a = x_b$
    end if
    SETPATH(V_b, v_a, x_a) ▷ Setzen des Wertes aus a in b
  end for
  return b ▷ $TSRb$ enthält die verknüpften Knoten
end function

Abbildung 6.3: Prozedurvorschlag DOMOR

Glossar

Algorithmus Eine präzise Handlungsvorschrift zur Lösung eines definierten Problems. Die Formulierung eines Algorithmus im Bereich der Informatik erfolgt oft durch Angabe eines Programms einer Programmiersprache oder einer Pseudo-Programmiersprache (d.h. mit eindeutigen, aber umgangssprachlich formulierten Elementen).

Arithmetisches Mittel (auch Durchschnittswert) Ein rechnerisch bestimmter Mittelwert, der oft eine geeignete Schätzung für den Erwartungswert einer Verteilung darstellt, aus der eine Stichprobe stammt. Das arithmetische Mittel ist vergleichsweise einfach zu berechnen, allerdings empfindlich gegenüber Extremwerten.

Auflösung von Wortmehrdeutigkeiten (engl. Word Sense Disambiguation, WSD) Auflösung von Mehrdeutigkeiten, die durch die Nutzung von gleich lautenden Wörtern mit unterschiedlicher Bedeutung (Homonyme) oder verwandter, voneinander abgeleiteter Bedeutung (Polynome) entstehen.

Baseline Die Baseline eines Auswertungsverfahrens stellt einen unteren standardisierten Referenzpunkt dar. Im Allgemeinen wird von teilnehmenden Systemen erwartet, daß sie in ihrer Leistung die Baseline übertreffen. Die Baseline hängt dabei in starkem Maße vom Auswertungsverfahren selbst ab, und von dem, was durch die Evaluation gezeigt werden soll.

Bedeutungsvariante (engl. *Sense*) Ein Lexem besitzt häufig mehrere Bedeutungen bzw, Bedeutungsvarianten. In einem Wörterbuch werden üblicherweise die Bedeutungsvarianten eines Wortes unter diesem subsumiert. In Bezug auf die lexikalische Datenbank WordNet werden Bedeutungsvarianten auch als „Synonymgruppen“ (engl. synsets) bezeichnet. Beispielsweise besitzt das Lexem *Bank* eine Bedeutungsvariante als *Sitzmöbel*, eine als *finanzielle Institution* etc.

BOW (*Bag-of-Words*) Eine Darstellungsweise für Texte, in welcher oft Worthäufigkeiten über Elemente in Vektoren dargestellt werden. Es handelt sich damit um eine Sammlung von Wörtern (und Werten) unter Weglassung der Strukturinformation (Wird auch als Vektorraum-Repräsentation bezeichnet).

British National Corpus (Abk. BNC) Sammlung gesprochener und geschriebener Sprache der britisch-englischen Sprache. Der in dieser Arbeit genutzte Teil des BNC besteht überwiegend aus britischen Nachrichtenmeldungen, Journalen und Büchern der zweiten Hälfte des 20. Jahrhunderts. Texte aus dem BNC dienen dem SENSEVAL-3 Verfahren als Test- und Trainingsdatengrundlage.

Chunking Aufteilen eines Textes in verschiedene Abschnitte (engl. *chunks*).

Clustering Ballungsanalyse, unüberwachtes statistisches Verfahren zur Ermittlung von Gruppen von Objekten, wobei jeweils unterschiedliche Algorithmen zur Anwendung kommen können.

Disambiguierung (engl. *Disambiguation*) Auflösung von Mehrdeutigkeiten.

Evaluation Auswertung, i.d.R. im Rahmen eines Experimentes. Schließt die Bewertung des Resultates mit ein und setzt die Beschreibung eines Prozesses oder Verfahrens voraus.

Extension Umfang eines Begriffes: die Menge der Objekte, die durch diesen Begriff bezeichnet werden. Die Extension kann nur empirisch festgestellt werden.

Granularität Genauigkeit einer sprachlichen Einheit; z.B. besitzt die Einheit „Satz“ eine andere Granularität als die Einheit „Wort“. Natürlichsprachliche Verfahren arbeiten häufig auf einer festgelegten Granularitätsebene, d.h. sie verarbeiten beispielsweise nur vollständige Sätze.

Heuristik auch, umgangssprachlich: Faustregel. Heuristiken werden üblicherweise eingesetzt, wenn zur Lösung eines Problems kein bestimmt erfolgreicher Algorithmus bekannt ist.

Homonymie Eigenschaft eines Wortes, mehrdeutig zu sein, wobei (vormals unterschiedliche) Wörter im Laufe der Zeit gleichlautend wurden (vgl. Polysemie).

Implementation Wechsel von einer abstrakten Ebene auf eine konkretere Ebene. Beispielsweise wird hierunter die Ausprogrammierung einer Softwarespezifikation verstanden.

Inferenz auch: Schlußfolgerung.

Inkrementell-iterativ Vorgehensweise, i.d.R. zur schrittweisen Verbesserung des Ergebnisses eines Prozesses, wobei der Prozeß häufig ohne ein definiertes Ende vorgegeben ist.

Intension Inhalte eines Begriffes: die Menge der Merkmale, die ein Objekt aufweist. Aufgrund der begrifflichen Unschärfe wird die Intension per Konvention vereinbart.

Inversion Im Zusammenhang dieser Arbeit: Regelumkehrung.

Isomorphie auch: Strukturelle Gleichheit

Java Eine von der Firma SUN entwickelte, objektorientierte Programmiersprache.

Junktor In der Aussagenlogik bezeichnet der Begriff "Junktor" eine Verknüpfung zwischen zwei Aussagen.

Kanonische Form vgl. Normalform.

Kappa-Maß Ein 1960 von Cohen vorgeschlagenes Maß der Übereinstimmung von üblicherweise zwei Beurteilern.

Kognition Bezeichnung für bestimmte Arten menschlicher mentaler Prozesse, z.B. Denken, Wissen, Verstehen, Lernen, etc. Die Kognition wird oft gegenüber emotionalen Prozessen abgegrenzt.

Kollokation Gemeinsam auftretende Wörter, die durch ihre Bedeutung verbunden sind, z.B. "Baby wickeln".

Konkatenation In dieser Arbeit: sequentielle Verknüpfung von textuellen Repräsentationen. Zum Beispiel werden $/Top$ und $/world$ zu $/top/world$ konkateniert.

Kontext, textueller Der textuelle Kontext eines Wortes besteht aus den das Wort umgebenden Wörtern. Häufig wird ein Textfenster für den Kontext angegeben, also ein Bereich von x Worten, der dem Kontext zugerechnet wird.

Korpus Nach Kilgarriff und Grefenstette eine "Zusammenstellung von Texten als Objekt der Sprach- oder Literaturforschung".

Kreuzvalidierung Validierungsmethode von Experimenten, häufig im Bereich der überwachten Klassifikationsversuche genutzt. Man spricht von einer k-fachen Kreuzvalidierung, wenn die Menge der Testdaten in k Partitionen aufgeteilt wird und in k Experimenten die i-te Partition die Testmenge darstellt, während die anderen (k-1) Partitionen als Trainingsdatenmenge genutzt werden. Aus den Einzelergebnissen wird dann das Gesamtergebnis berechnet. Man versucht durch diese Methodik den Effekt von Ungleichheiten in der Gesamtdatenmenge zu minimieren.

Lexem Sprachliche Einheiten, die dieselbe Bedeutung, aber unterschiedliche morphosyntaktische Eigenschaften besitzen. z.B. sind *fliegen*, *flog*, *geflogen* Wortformen des Lexems *fliegen*.

LSA (engl. *Latent Semantic Analysis*) Verfahren zur statistischen Untersuchung von Textkorpora; 1990 von S. Deerwester, S. Dumais, T. Landauer, G. Furnas und R. Harshman entwickelt.

LSI (engl. *Latent Semantic Indexing*) Anwendung des LSA-Verfahrens zum Zweck des Information Retrieval

Machine Learning Erzeugung von Wissen aus Beispieldaten durch ein automatisiertes Verfahren, i.d.R. durch adaptive Konfiguration mit Hilfe eines generischen Algorithmus.

Median (auch: Zentralwert) Grenze zwischen zwei Hälften. Der Median ist in der Regel durch Aufzählen der Elemente einer Menge zu bestimmen und hierdurch relativ unempfindlich gegenüber Extremwerten. Dies prädestiniert seinen Einsatz im Falle von nicht-normalverteilten Mengen.

Merkmal (engl. *Feature*) Attribut eines Textfragmentes. Mögliche Merkmale sind z.B. Wortform, Tempus, Genus von Wörtern, aber auch: Auftreten von Wörtern in einem Textfragment. Merkmale werden häufig numerisch kodiert und als Elemente in Vektoren eingesetzt.

Natürlichsprachliche Texte Im Sinne dieser Arbeit sollen hierunter unrestringierte Texte verstanden werden, ohne Anforderungen an z.B. ihre Situiertheit zu stellen.

NLP Vgl. Textverarbeitung

Normalform A priori bestimmte (und üblicherweise vereinheitlichende) Form für logische oder mathematische Formeln.

Oberflächeneigenschaften von Bäumen Eigenschaften, die sich nicht aus strukturellen Informationen oder Eigenschaften der Baumelemente ableiten lassen, z.B. Tiefe oder Breite eines Baumes, Anzahl der Knoten etc.

ODP Abkürzung für „Open Directory Project“ (http://dmoz.org). Es handelt sich hierbei um ein kooperativ erstelltes Internet Verzeichnis, dessen Datenbasis frei verfügbar ist.

Opak “Unsichtbarkeit” interner Zusammenhänge. Ein *opaker* Mechanismus ist z.B. eine Maschine, deren Wirkungsweise lediglich über ihr Eingabe- und Ausgabeverhalten bestimmbar ist.

Orthogonal In dieser Arbeit sollen Vektoren, deren inneres Produkt 0 ergibt, als “orthogonal” bezeichnet werden.

In Annäherung hieran kann unter Orthogonalität die (paarweise) Unabhängigkeit von Konzepten oder Elementen (in einer Menge) voneinander verstanden werden. Beispielsweise sind sich die Konzepte “Bank” und “Pferd” im allgemeinen Sprachgebrauch so unähnlich, daß sie als “orthogonal” verstanden werden können.

Overfitting Übergenaue Anpassung eines Modells an eine bestimmte Datenmenge. Problematisch ist hierbei, daß neu hinzukommende Daten häufig durch das Modell nicht mehr korrekt erfaßt werden.

Paradigma Erkenntnistheoretischer Begriff, der bestimmte (wissenschaftlich orientierte) Mengen von Denkmustern voneinander abgrenzt. Zum Beispiel unterscheidet man in der praktischen Informatik das *Paradigma der objektorientierten Programmierung* vom *Paradigma der funktionalen Programmierung*.

Parsing (Im Zusammenhang der vorliegenden Arbeit ist ausschließlich das Parsing von natürlichen Sprachen gemeint) Das Parsing bezeichnet einen Übersetzungsvorgang, in welchem ein vorliegender Text in eine "interne" Form umgewandelt wird, welche ein Computer weiterverarbeiten kann. In der Regel wird die Struktur der sprachlichen Äußerung (hier: der Eingabetexte) durch den Parser ermittelt.

Partition Unterteilung einer Menge in mehrere disjunkte Untermengen

Performanz auch: Leistung.

Polysemie Form der Mehrdeutigkeit von Wörtern, bei der sich im Laufe der Zeit unterschiedliche Bedeutungen von der Ursprungsbedeutung abgespalten haben (z.B. bezeichnet "Bank" sowohl die Finanzinstitution als auch das diese beheimatende Gebäude).

PoS-Tagging (PoS steht für *Part-of-Speech*) Das PoS-Tagging bestimmt die jeweilige Wortart für Wörter oder Terme innerhalb eines Satzes bzw. Textes.

Pragmatik Wissen um den Sprachgebrauch. Interpretation von Sprachelementen in einem situativen Kontext.

Priorisierung Ablaufgewichtung. Beispielsweise können bestimmte Verarbeitungsschritte priorisiert werden und hieraufhin anderen Schritten vorgezogen werden.

Rauschen In der Physik: Störung eines Signales. In der Informatik wird hierunter häufig in ähnlicher Weise der Einfluß von Störgrößen verstanden.

RDBMS (Abk. für *Relational Database Management System*) Datenbanksystem, welches die Prinzipien der relationalen Datenverarbeitung umsetzt. SQL-Datenbanken, welche die Majorität der eingesetzten Systeme darstellen, sind relationale Datenbanken.

Sequentielle Datenverarbeitung Folgegerechte Verarbeitung. Hierunter wird eine Form der Datenverarbeitung verstanden, bei der jeder einzelne Schritt erst durchgeführt wird, sobald die Ergebnisse seines Vorgängers vorliegen (Im Gegensatz zur parallelen Datenverabeitung).

SemCor Mit WordNet-Bedeutungsvarianten (→WordNet) annotierter Korpus.

Semantik Wissen um die Bedeutung von Sprachelementen (ohne Kontextbezug).

Single-Idea Im Bereich der WSD-Systeme werden Systeme als "Single-Idea-Realisierung" bezeichnet, die lediglich eine einzige Idee umsetzen, d.h. einen einzelnen Kernalgorithmus implementieren. Fast alle der nicht-überwachten Systeme, die an der SENSEVAL-3 Evaluierung teilgenommen haben, sind als "Multiple-Idea-Realisierungen" ausgeführt - kombinieren also mehrere Verfahren.

Standardabweichung In der Stochastik ein Maß für die Streuung der Werte einer Zufallsvariablen um ihren Mittelwert. Das arithmetische Mittel und die Standardabweichung sind die zwei wichtigsten Maßzahlen zur Beschreibung der Eigenschaften einer Wertereihe.

Stopwörter Als "Stopwörter" werden häufig solche Wörter bezeichnet, die voraussichtlich keine eigenständige relevante Bedeutung mit sich führen, z.B. Artikel oder Präpositionen. Diese Wörter werden aus Gründen der Effizienz oft vor der Weiterverarbeitung aus Texten entfernt, z.B. wenn die Texte in den Index einer Suchmaschine importiert werden sollen.

SVM (Support Vector Machine) Klassifikatorverfahren aus dem Bereich des maschinellen Lernens. Eine Menge von Objekten wird i.d.R. durch eine Hyperebene separiert. Diese Hyperebene dient zur Einordnung später hinzugefügter Objekte und somit zur Klassifikation derselben.

Syntax Vgl. Satzbau.

Taxonomie hierarchische Klassifikation sprachlicher Begriffe

Textverarbeitung Im Gegensatz zum "Word Processing", d.h. zur formatierten Eingabe von Texten durch einen Benutzer, häufig im Zusammenhang mit sogenannten WYSIWYG ("What You See Is What You Get)-Editoren, bezeichnet der Begriff "Textverarbeitung" im Kontext der vorliegenden Arbeit die automatisierte Verarbeitung von Texten. Textverarbeitungsanwendungen sind demzufolge Programme, die z.B. automatische Zusammenfassungen von Texten erstellen, Texte aus Datenbankinhalten generieren, Einträge für Suchmaschinen gestalten etc. In englischsprachlichen Texten wird die Umschreibung "text based Natural Language Processing" (text based NLP) verwendet.

Tokenisierung Erzeugung von Termen aus einem Text.

Traversieren Beim Traversieren eines Baumes werden alle Elemente des Baumes besucht, um eine bestimmte Operation auf ihnen durchzuführen.

Tupel Geordnete Zusammenstellung von Elementen, d.h. eine Zusammenstellung, bei der es auf die Reihenfolge der Elemente ankommt. Damit unterscheidet sich z.B. das 2-Tupel $< 4, 8 >$ vom 2-Tupel $< 8, 4 >$

Varianz Statistisches Streuungsmaß, d.h. ein Maß für die Abweichung einer Zufallsvariable von ihrem Erwartungswert

WordNet Lexikalische Datenbank, oft genutzt als Datenbasis für Versuche und Verfahren im Bereich der WSD-Forschung.

WSD Abkürzung für: Word Sense Disambiguation, engl. für „Auflösung von Wortmehrdeutigkeiten“.

Abbildungsverzeichnis

Tabellenverzeichnis

Literaturverzeichnis

[Aas & Eikvil, 1999] Aas, K. & Eikvil, L. (1999). *Text Categorization: a survey.* Technical Report 941, Norwegian Computing Center.

[Abeille, 1988] Abeille, A. (1988). Parsing french with tree adjoining grammar: some linguistic accounts. *COLING-88*, (pp. 7–12).

[Abney, 1992] Abney, S. (1992). Prosodic structure, performance structure and phrase structure. In *Speech and Natural Language Workshop* San Mateo, CA: Morgan Kaufmann Publishers.

[Abney, 1996] Abney, S. (1996). Partial parsing via finite-state cascades. In *Proceedings of the ESSLLI '96 Robust Parsing Workshop*.

[Agirre et al., 2000] Agirre, E., Ansa, O., Hovy, E. H., & Martínez, D. (2000). Enriching very large ontologies using the www. In *ECAI Workshop on Ontology Learning*.

[Agirre & Martinez, 2001a] Agirre, E. & Martinez, D. (2001a). Knowledge sources for word sense disambiguation. In V. Matousek, P. Mautner, R. Moucek, & K. Tauser (Eds.), *Proceedings of International Conference on Text, Speech and Dialogue* Plzen (Pilsen), Czech Republic.

[Agirre & Martinez, 2001b] Agirre, E. & Martinez, D. (2001b). Learning class-to-class selectional preferences. In *Proceedings of the ACL CONLL Workshop* Toulouse, France.

[Agirre & Martínez, 2004] Agirre, E. & Martínez, D. (2004). The basque country university system: English and basque tasks. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 44–48). Barcelona, Spain: Association for Computational Linguistics.

[Agirre & Martinez, 2004] Agirre, E. & Martinez, D. (2004). Unsupervised wsd based on automatically retrieved examples: The importance of bias. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* Barcelona, Spain.

[Agirre & Rigau, 1996] Agirre, E. & Rigau, G. (1996). Word sense disambiguation using conceptual density. In *Proceedings of COLING'96* (pp. 16–22). Copenhagen, Danmark.

[Al-Halimi et al., 1998] Al-Halimi, R., Berwick, R. C., Burg, J. F. M., Chodorow, M., Fellbaum, C., Grabowski, J., Harabagiu, S., Hearst, M. A., Hirst, G., Jones, D. A., Kazman, R., Kohl, K. T., Landes, S., Leacock, C., Miller, G. A., Miller, K. J., Moldovan, D., Nomura, N., Priss, U., Resnik, P., St-Onge, D., Tengi, R., van de Riet, R. P., & Voorhees., E. (1998). *WordNet An Electronic Lexical Database*. The MIT Press.

[Allen, 1995] Allen, J. (1995). *Natural Language Understanding*. Benjaming/-Cummings Publish. Corp, CA, 2 edition.

[Altintas & Coskun, 2005] Altintas, E. & Coskun, E. K. V. (2005). A new semantic measure evaluated in word sense disambiguation. In *NODALIDA 2005 (15th Nordic Conference of Computational Linguistics)*.

[Anderson, 1980] Anderson, J. R. (1980). Concepts, propositions, and schemata: What are the cognitive units? In J. Flowers (Ed.), *Nebraska Symposium on Motivation*. Lincoln, Nebraska: University of Nebraska Press.

[Anderson, 1996] Anderson, J. R. (1996). Act: A simple theory of complex cognition. *American Psychologist*, (pp. 355–365).

[Anderson et al., 1977] Anderson, J. R., Kline, P. J., & Lewis, C. H. (1977). *Cognitive Processes in Comprehension*, chapter A production system model for language processing, (pp. 271–311). Lawrence Erlbaum Associates: Hillsdale, NJ.

[Artiles et al., 2004] Artiles, J., Penas, A., & Verdejo, F. (2004). Word sense disambiguation based on term to term similarity in a context space. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 58–63). Barcelona, Spain: Association for Computational Linguistics.

[Atserias et al., 2004] Atserias, J., Villarejo, L., Rigau, G., Agirre, E., Carroll, J., Magnini, B., & Vossen, P. (2004). The meaning multilingual central repository. In *Proceedings of the Second International WordNet Conference* http://www.lsi.upc.es/ nlp/meaning/meaning.html.

[Austin, 1968] Austin, J. L. (1968). Performative und konstatierende äußerung. In *Sprache und Analysis* (pp. 140 – 153). G???ttingen: R. Bubner.

[Baeza-Yates & Ribeiro-Neto, 1999] Baeza-Yates, R. A. & Ribeiro-Neto, B. A. (1999). *Modern Information Retrieval*. ACM Press / Addison-Wesley.

[Baker et al., 1998] Baker, C. F., Fillmore, C. J., & Lowe, J. B. (1998). The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics* (pp. 86–90). Morristown, NJ, USA: Association for Computational Linguistics.

[Baldridge & Morton, 2006] Baldridge, J. & Morton, T. (2006). Opennlp. http://opennlp.sourceforge.net/.

[Baldwin et al., 2008] Baldwin, T., Kim Su Nam, Francis Bond, S. F. D. M., & Tanaka, T. (2008). Mrd-based word sense disambiguation: Further extending lesk. In *Proceedings of the Third International Joint Conference on Natural Language Processing (IJCNLP 2008)* (pp. 775–780). Hyderabad, India.

[Banerjee & Pedersen, 2003] Banerjee, S. & Pedersen, T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the 18th IJCAI* (pp. 805–810). Acapulco.

[Banko & Brill, 2001] Banko, M. & Brill, E. (2001). Scaling to very very large corpora for natural language disambiguation. In *ACL '01: Proceedings of the 39th Annual Meeting on Association for Computational Linguistics* (pp. 26–33). Morristown, NJ, USA: Association for Computational Linguistics.

[Bar-Hillel, 1960] Bar-Hillel, Y. (1960). The present status of automatic translations of languages. *Advances in Computers*, I, 91–163.

[Bärenfänger, 2002] Bärenfänger, O. (2002). Merkmals- und prototypensemantik: Einige grundsätzliche überlegungen. *Linguistik online*, 12.

[Basile et al., 2007] Basile, P., de Gemmis, M., Gentile, A. L., Lops, P., & Semeraro, G. (2007). Uniba: Jigsaw algorithm for word sense disambiguation. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)* (pp. 398–401). Prague, Czech Republic: Association for Computational Linguistics.

[Beale et al., 1995] Beale, S., Nirenburg, S., & Mahesh, K. (1995). Semantic analysis in the mikrokosmos machine translation project. In *Proceedings of the 2nd Symposium on Natural Language Processing* (pp. 297–307). Bangkok, Thailand.

[Bhandari, 1999] Bhandari, D. (1999). *Extraction of Web Information Using W4F Wrapper Factory and XML-QL Query Language*. Department of Computer and Information Science Technical Report MS-CIS-99-20, University of Pennsylvania.

[Bialecki et al., 2008] Bialecki, A., Cutting, D., Ganyo, S., Goller, C., Gospodnetic, O., Harwood, M., Hatcher, E., Hostetter, C., Ingersoll, G., McCandless, M., Naber, D., Seeley, Y., & Siren, S. (2008). Apache lucene project. http://lucene.apache.org.

[Bille, 2005] Bille, P. (2005). A survey on tree edit distance and related problems. *Theor. Comput. Sci.*, 337(1-3), 217–239.

[Blutner, 1995] Blutner, R. (1995). *Die Ordnung der Wörter. Kognitive und lexikalische Strukturen*, chapter Prototypen und Kognitive Semantik. de Gruyter: Berlin, New York.

[Boyd-Graber & Blei, 2007] Boyd-Graber, J. & Blei, D. (2007). Putop: Turning predominant senses into a topic model for word sense disambiguation. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)* (pp. 277–281). Prague, Czech Republic: Association for Computational Linguistics.

[Brody et al., 2006] Brody, S., Navigli, R., & Lapata, M. (2006). Ensemble methods for unsupervised wsd. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Assocaition for Computational Linguistics* (pp. 97–104). Sydney, Australia.

[Brown et al., 1991] Brown, P. F., Pietra, S. A. D., Pietra, V. J. D., & Mercer, R. L. (1991). Word-sense disambiguation using statistical methods. In *Proceedings of the 29th annual meeting on Association for Computational Linguistics* (pp. 264–270). Morristown, NJ, USA: Association for Computational Linguistics.

[Bruce et al., 1992] Bruce, R., Wilks, Y., Guthrie, L., Slator, B., & Dunning, T. (1992). *NounSense - A Disambiguated Noun Taxonomy with a Sense of Humour*. Research Report MCCS-92-246, Computing Research Laboratory, New Mexico State University.

[Buscaldi et al., 2004] Buscaldi, D., Rosso, P., & Masulli, F. (2004). The upv-unige-ciaosenso wsd system. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 77–82). Barcelona, Spain: Association for Computational Linguistics.

[Cabezas et al., 2004] Cabezas, C., Bhattacharya, I., & Resnik, P. (2004). The university of maryland senseval-3 system descriptions. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 83–87). Barcelona, Spain: Association for Computational Linguistics.

[Cai et al., 2007] Cai, J. F., Lee, W. S., & Teh, Y. W. (2007). Nus-ml:improving word sense disambiguation using topic features. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)* (pp. 249–252). Prague, Czech Republic: Association for Computational Linguistics.

[Canas et al., 2003] Canas, A., Valerio, A., Lalinde-Pulido, J., Carvalho, M., & Arguedas, M. (2003). Using wordnet for word sense disambiguation to support

concept map construction. In *10th International Symposium on String Processing and Information Retrieval* Manaus, Brazil.

[Carpuat & Wu, 2007] Carpuat, M. & Wu, D. (2007). Improving statistical machine translation using word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)* (pp. 61–72).

[Chan et al., 2007] Chan, Y. S., Ng, H. T., & Zhong, Z. (2007). Nus-pt: Exploiting parallel texts for word sense disambiguation in the english all-words tasks. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)* (pp. 253–256). Prague, Czech Republic: Association for Computational Linguistics.

[Chanen & Patrick, 2004] Chanen, A. & Patrick, J. (2004). Complex, corpus-driven, syntactic features for word sense disambiguation. In *Australasian Language Technology Workshop*.

[Chen & Chang, 1998] Chen, J. N. & Chang, J. S. (1998). Topical clustering of MRD senses based on information retrieval techniques. *Computational Linguistics*, 24(1), 61–95.

[Chin et al., 2006] Chin, O. S., Kulathuramaiyer, N., & Yeo, A. (2006). Automatic discovery of concepts from text. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*.

[Chklovski & Mihalcea, 2002] Chklovski, T. & Mihalcea, R. (2002). Building a sense tagged corpus with open mind word expert. In *Proceedings of the ACL 2002 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions* Philadelphia.

[Chomsky, 1957] Chomsky, N. (1957). *Syntactic Structures*. Mouton & Co.

[Chute et al., 2006] Chute, C., Pedersen, T., Maclin, R., Joshi, M., & Pakhomov, S. (2006). An end-to-end supervised target-word sense disambiguation system. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence* (pp. 1941–1942).

[Ciaramita & Johnson, 2004] Ciaramita, M. & Johnson, M. (2004). Multi-component word sense disambiguation. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 97–100). Barcelona, Spain: Association for Computational Linguistics.

[Cohen, 1960] Cohen, J. (1960). A coefficient of agreement for nominal scale. *Educational and Psychological Measurement*, 20, 37–46.

[Cohn, 2003] Cohn, T. (2003). Performance metrics for word sense disambiguation. In *Australasian Language Technology Workshop* (pp. 49–56).

[Crestan, 2004] Crestan, E. (2004). Contextual semantics for wsd. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 101–104). Barcelona, Spain: Association for Computational Linguistics.

[Crysmann et al., 2001] Crysmann, B., Frank, A., Kiefer, B., Müller, S., Neumann, G., Piskorski, J., Schäfer, U., Siegel, M., Uszkoreit, H., Xu, F., Becker, M., & Krieger, H.-U. (2001). An integrated architecture for shallow and deep processing. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (pp. 441–448). Morristown, NJ, USA: Association for Computational Linguistics.

[Cuadros et al., 2005] Cuadros, M., Padro, L., & Rigau, G. (2005). Comparing methods for authomatic acquisition of topic signatures. In *5th International Conference on Recent Advances in Natural Language Processing* Borovets, Bulgaria.

[Cunningham, 2000] Cunningham, H. (2000). *Software Architecture for Language Engineering*. PhD thesis, University of Sheffield. `http://gate.ac.uk/sale/thesis/`.

[Dagan & Itai, 1994] Dagan, I. & Itai, A. (1994). Word sense disambiguation using a second language monolingual corpus. *Comput. Linguist.*, 20(4), 563–596.

[Dahlgren, 1992] Dahlgren, K. (1992). The autonomy of shallow lexical knowledge. In *Proceedings of the First SIGLEX Workshop on Lexical Semantics and Knowledge Representation* (pp. 255–267). London, UK: Springer-Verlag.

[Davoudi, 2005] Davoudi, M. (2005). Inference generation skill and text comprehension. *The Reading Matrix*, 5(1).

[Decadt et al., 2004] Decadt, B., Hoste, V., Daelemans, W., & Van den Bosch, A. (2004). Gambl, genetic algorithm optimization of memory-based wsd. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 108–112). Barcelona, Spain: Association for Computational Linguistics.

[Deerwester et al., 1990] Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., & Harshman, R. A. (1990). Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6), 391–407.

[Diab & Resnik, 2001] Diab, M. & Resnik, P. (2001). An unsupervised method for word sense tagging using parallel corpora. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (pp. 255–262). Morristown, NJ, USA: Association for Computational Linguistics.

[Dietterich, 1997] Dietterich, T. (1997). Machine learning research: Four current directions. *AI Magazine*, 18(4), 97–136.

[Edmonds, 2002] Edmonds, P. (2002). Senseval: The evaluation of word sense disambiguation systems. *ELRA Newsletter*, 7(3).

[Escudero et al., 2004] Escudero, G., Màrquez, L., & Rigau, G. (2004). Talp system for the english lexical sample task. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 113–116). Barcelona, Spain: Association for Computational Linguistics.

[Férnandez-Amorós, 2004] Férnandez-Amorós, D. (2004). Wsd based on mutual information and syntactic patterns. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 117–120). Barcelona, Spain: Association for Computational Linguistics.

[Florian & Yarowsky, 2002] Florian, R. & Yarowsky, D. (2002). Modeling consensus: classifier combination for word sense disambiguation. In *EMNLP '02: Proceedings of the ACL-02 conference on Empirical methods in natural language processing* (pp. 25–32). Morristown, NJ, USA: Association for Computational Linguistics.

[Frege, 1892] Frege, G. W. (1892). über sinn und bedeutung. *Zeitschrift für Philosophie und philosophische Kritik*, (pp. 25–50). translated Ön sense and meaning.", Reprinted in McGuinness, Brian (ed.) Collected Papers on Mathematics, Logic and Philosophy, 157–177, Oxford: Basil Blackwell (1984).

[Galley & McKeown, 2003] Galley, M. & McKeown, K. (2003). Improving word sense disambiguation in lexical chaining. In *Proceedings of the 18th IJCAI* (pp. 1486–1488). Acapulco.

[García-Vega et al., 2004] García-Vega, M., García-Cumbreras, M., Martín-Valdivia, M. T., & Urena-López, L. A. (2004). The university of jaén word sense disambiguation system. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 121–124). Barcelona, Spain: Association for Computational Linguistics.

[Gelbukh et al., 2005] Gelbukh, A., Sidorov, G., & Han, S. (2005). On some optimization heuristics for lesk-like wsd algorithms. In *10th International Conference on Applications of Natural Languages to Information Systems* (pp. 402–405). Alicante, Spain: Springer-Verlag.

[Gellert et al., 1971] Gellert, W., Kästner, H., Hellwich, M., & Kästner, H. (1971). *Kleine Enzyklopädie Mathematik*. VEB Bibliographisches Institut Leipzig, 13. edition.

[Ginter et al., 2005] Ginter, F., Pyysalo, S., & Salakoski, T. (2005). Document classification using semantic networks with an adaptive similarity measure. In *5th International Conference on Recent Advances in Natural Language Processing* Borovets, Bulgaria: Bulgarian Academy of Sciences.

[Gomez, 2001] Gomez, F. (2001). An algorithm for aspects of semantic interpretation using an enhanced wordnet. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics, NAACL-2001* CMU, Pittsburgh.

[Graca et al., 2006] Graca, J., Mamede, N. J., & Pereira, J. D. (2006). Framework for integrating natural language tools. *Lecture Notes in Computer Science*, (3960), 110–119.

[Grishman, 1997] Grishman, R. (1997). Information extraction: Techniques and challenges. In *SCIE* (pp. 10–27).

[Gruber, 1993] Gruber, T. R. (1993). A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2), 199–220.

[Halpin, 2006] Halpin, H. (2006). Automatic evaluation and composition of nlp pipelines with web services. In *Proceedings of Language Resources and Evaluation of Corpora Conference (LREC 2006)*.

[Hassel, 2005] Hassel, M. (2005). Word sense disambiguation using co-occurrence statistics on random labels. In *5th International Conference on Recent Advances in Natural Language Processing*: Bulgarian Academy of Sciences.

[Helbig, 2001] Helbig, H. (2001). *Die Semantische Struktur Natürlicher Sprache: Wissensrepräsentation Mit MultiNet*. Berlin: Springer-Verlag.

[Hutchins, 1999] Hutchins, J. (1999). Warren weaver memorandum, july 1949. *MT News International*, (99).

[Ide & Veronis, 1998] Ide, N. & Veronis, J. (1998). Introduction to the special issue on word sense disambiguation: the state of the art. *Comput. Linguist.*, 24(1), 2–40.

[Joachims, 1998] Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the European Conference on Machine Learning*.

[Kanerva et al., 2000] Kanerva, P., Kristofersson, J., & Holst, A. (2000). Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society* (pp. 1036).: Erlbaum.

[Kaplan, 1989] Kaplan, R. M. (1989). The formal architecture of lexical-functional grammar. *Journal of Information Science and Engineering*, 5(4), 305–322.

[Kavalec & Svatek, 2002] Kavalec, M. & Svatek, V. (2002). Information extraction and ontology learning guided by web directory. In *ECAI Workshop on NLP and ML for ontology engineering* Lyon.

[Keller & Bengio, 2005] Keller, M. & Bengio, S. (2005). A neural network for text representation. In *Proceedings of the 15th International Conference on Artificial Neural Networks: Biological Inspirations, ICANN, Lecture Notes in Computer Science*, volume LNCS 3697 (pp. 667–672).: Springer-Verlag.

[Kennedy & Broquist, 2004] Kennedy, S. & Broquist, L. (2004). The wordsmyth educational dictionary-thesaurus. http://www.wordsmyth.net.

[Kilgarriff, 1997a] Kilgarriff, A. (1997a). I don't believe in word senses. *Computers and the Humanities*, 31 (2), 91–113.

[Kilgarriff, 1997b] Kilgarriff, A. (1997b). What is word sense disambiguation good for? In *Proc. Natural Language Processing Pacific Rim Symposium* (pp. 209–214). Phuket, Thailand.

[Kilgarriff & Grefenstette, 2003] Kilgarriff, A. & Grefenstette, G. (2003). Introduction to the special issue on the web as corpus. *Computational Linguistics*, 29(3), 333–347.

[Klapaftis & Manandhar, 2005] Klapaftis, I. P. & Manandhar, S. (2005). Google & wordnet based word sense disambiguation. *Proceedings of the first workshop on learning and extending ontologies by using machine learning methods, International conference on Machine Learning, ICML-05, Bonn, Germany*.

[Kranig, 2005] Kranig, S. (2005). Evaluation of language identification methods. Master's thesis, University of Tübingen.

[Kulkarni & Pedersen, 2005] Kulkarni, A. & Pedersen, T. (2005). Senseclusters: Unsupervised clustering and labeling of similar contexts. In *Proceedings of the Demonstration and Interactive Poster Session of the 43rd Annual Meeting of the Association for Computational Linguistics* Ann Arbor, MI.

[Lafourcade, 2002] Lafourcade, M. (2002). Guessing hierarchies and symbols for word meanings through hyperonyms and conceptual vectors. In *OOIS '02: Proceedings of the Workshops on Advances in Object-Oriented Information Systems* (pp. 84–93). London, UK: Springer-Verlag.

[Landauer et al., 1998] Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to latent semantic analysis. *Discourse Processes*, 25, 259–284.

[Leacock et al., 1998] Leacock, C., Miller, G. A., & Chodorow, M. (1998). Using corpus statistics and wordnet relations for sense identification. *Comput. Linguist.*, 24(1), 147–165.

[Lee et al., 2004] Lee, Y. K., Ng, H. T., & Chia, T. K. (2004). Supervised word sense disambiguation with support vector machines and multiple knowledge sources. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 137–140). Barcelona, Spain: Association for Computational Linguistics.

[Lesk, 1986] Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *SIGDOC '86: Proceedings of the 5th annual international conference on Systems documentation* (pp. 24–26). New York, NY, USA: ACM Press.

[Lewis, 1992] Lewis, D. D. (1992). *Representation and learning in information retrieval*. PhD thesis.

[Lin, 2000] Lin, D. (2000). Word sense disambiguation with a similarity-smoothed case library. *Computers and the Humanities*, 34, 147–152(6).

[Litkowski, 2004] Litkowski, K. (2004). Explorations in disambiguation using xml text representation. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 141–146). Barcelona, Spain: Association for Computational Linguistics.

[Luhn, 1958] Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2, 159–165.

[Magnini & Cavagli, 2000] Magnini, B. & Cavagli, G. (2000). Integrating subject field codes into wordnet. In M. Gavrilidou, G. Crayannis, S. Markantonatu, S. Piperidis, & G. Stainhaouer (Eds.), *Proceedings of LREC-2000, Second International Conference on Language Resources and Evaluation* (pp. 1413–1418). Athens, Greece.

[Mahesh, 1996] Mahesh, K. (1996). *Ontology Development for Machine Translation: Ideology and Methodology*. Technical Report MCCS???96???292, Computing Research Laboratory New Mexico State University.

[Mahesh & Nirenburg, 1995a] Mahesh, K. & Nirenburg, S. (1995a). Semantic classification for practical natural language processing. In *Sixth ASIS SIG/CR Classification Research Workshop: An Interdisciplinary Meeting* Chicago IL.

[Mahesh & Nirenburg, 1995b] Mahesh, K. & Nirenburg, S. (1995b). A situated ontology for practical nlp. In *Workshop on Basic Ontological Issues in Knowledge Sharing, International Joint Conference on Artificial Intelligence (IJCAI-95)* Montreal, Canada.

[Mavroeidis et al., 2005] Mavroeidis, D., Tsatsaronis, G., Vazirgiannis, M., Theobald, M., & Weikum, G. (2005). Word sense disambiguation for exploiting hierarchical thesauri in text classification. In *9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD)* (pp. 181–192).: Springer-Verlag LNAI.

[McCallum et al., 1998] McCallum, A., Rosenfeld, R., Mitchell, T. M., & Ng, A. Y. (1998). Improving text classification by shrinkage in a hierarchy of classes. In *ICML* (pp. 359–367).

[McCarthy, 1997] McCarthy, D. (1997). Word sense disambiguation for acquisition of selectional preferences. In P. Vossen, G. Adriaens, N. Calzolari, A. Sanfilippo, & Y. Wilks (Eds.), *Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications* (pp. 52–60). New Brunswick, New Jersey: Association for Computational Linguistics.

[McCarthy et al., 2004a] McCarthy, D., Koeling, R., Weeds, J., & Carroll, J. (2004a). Using automatically acquired predominant senses for word sense disambiguation. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 151–154). Barcelona, Spain: Association for Computational Linguistics.

[McCarthy et al., 2004b] McCarthy, D., Koeling, R., Weeds, J., & Carroll, J. A. (2004b). Finding predominant word senses in untagged text. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics* (pp. 279–286). Barcelona, Spain.

[Meinhardt, 1984] Meinhardt, H.-J. (1984). Invariante, variante und prototypische merkmale der wortbedeutung. *Zeitschrift für Germanistik*, 5(1), 60–69.

[Mihalcea, 2002] Mihalcea, R. (2002). Word sense disambiguation with pattern learning and automatic feature selection. *Natural Language Engineering*, 8, 343–358. Cambridge University Press.

[Mihalcea, 2005] Mihalcea, R. (2005). Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing* (pp. 411–418). Vancouver, British Columbia, Canada: Association for Computational Linguistics.

[Mihalcea, 2007] Mihalcea, R. (2007). Using wikipedia for automatic word sense disambiguation. In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL 2007)* Rochester.

[Mihalcea et al., 2004] Mihalcea, R., Chklovski, T., & Kilgarriff, A. (2004). The senseval-3 english lexical sample task. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 25–28). Barcelona, Spain: Association for Computational Linguistics.

[Mihalcea & Faruque, 2004] Mihalcea, R. & Faruque, E. (2004). Senselearner: Minimally supervised word sense disambiguation for all words in open text. In *Proceedings of ACL/SIGLEX Senseval-3* Barcelona, Spain.

[Mihalcea & Moldovan, 1998] Mihalcea, R. & Moldovan, D. (1998). Word sense disambiguation based on semantic density. In *Proceedings of COLING-ACL '98 Workshop on Usage of WordNet in Natural Language Processing Systems* Montreal, Canada.

[Mihalcea & Moldovan, 1999] Mihalcea, R. & Moldovan, D. I. (1999). An automatic method for generating sense tagged corpora. In *AAAI '99/IAAI '99: Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference innovative applications of artificial intelligence* (pp. 461–466). Menlo Park, CA, USA: American Association for Artificial Intelligence.

[Miller et al., 2005] Miller, G. A., Fellbaum, C., Tengi, R., Wolff, S., Wakefield, P., Langone, H., & Haskell, B. (2005). Wordnet - a lexical database for the english language. http://www.cogsci.princeton.edu/ wn/index.shtml.

[Mladenic, 1998] Mladenic, D. (1998). Turning yahoo to automatic web-page classifier. In *European Conference on Artificial Intelligence* (pp. 473–474).

[Mohammad & Pedersen, 2004] Mohammad, S. & Pedersen, T. (2004). Complementarity of lexical and simple syntactic features: The syntalex approach to senseval-3. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 159–162). Barcelona, Spain: Association for Computational Linguistics.

[Morris & Hirst, 1991] Morris, J. & Hirst, G. (1991). Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Comput. Linguist.*, 17(1), 21–48.

[Mueller, 2006] Mueller, T. (2006). H2 database engine. http://www.h2database.com.

[Navigli & Lapata, 2007] Navigli, R. & Lapata, M. (2007). Graph connectivity measures for unsupervised word sense disambiguation. In *IJCAI* (pp. 1683–1688).

[Navigli & Velardi, 2004] Navigli, R. & Velardi, P. (2004). Structural semantic interconnection: a knowledge-based approach to word sense disambiguation. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 179–182). Barcelona, Spain: Association for Computational Linguistics.

[Netscape Communications Corporation, 2004] Netscape Communications Corporation (2004). Open directory project. http://dmoz.org.

[Neumann et al., 2000] Neumann, G., Braun, C., & Piskorski, J. (2000). A divide-and-conquer strategy for shallow parsing of german free texts. In *Proceedings of the sixth conference on Applied natural language processing* (pp. 239–246). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

[Nica et al., 2004] Nica, I., Montoyo, A., Vazquez, S., & Marti, M. A. (2004). An unsupervised wsd algorithm for a nlp system. In *International conference on applications of natural language to information system*, volume 3136 (pp. 288–298). Salford: Springer, Berlin.

[Niu et al., 2004] Niu, C., Li, W., Srihari, R. K., Li, H., & Crist, L. (2004). Context clustering for word sense disambiguation based on modeling pairwise context similarities. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 187–190). Barcelona, Spain: Association for Computational Linguistics.

[Niu et al., 2003] Niu, C., Zheng, Z., Srihari, R., Li, H., & Li, W. (2003). Unsupervised learning for verb sense disambiguation using both trigger words and parsing relations. In *Proceedings of Pacific Association for Computational Linguistics (PACLING03)* Halifax, Nova Scotia, Canada.

[Niu et al., 2005] Niu, Z.-Y., Ji, D.-H., & Tan, C.-L. (2005). Word sense disambiguation using label propagation based semi-supervised learning. In *Proceedings of the ACL*.

[Oh & Choi, 2002] Oh, J.-H. & Choi, K.-S. (2002). Word sense disambiguation using static and dynamic sense vectors. In *Proceedings of the 19th international conference on Computational linguistics* (pp. 1–7). Morristown, NJ, USA: Association for Computational Linguistics.

[Ohler, 1986] Ohler, M. (1986). Sprachspiel, sprechakt, regel - über wittenstein und searle. In W. Leinfellner & F. M. Wuketits (Eds.), *The Tasks of Contemporary Philosophy*. Hölder-Pichler-Tempsky.

[Overberg, 1999] Overberg, P. (1999). Merkmalssemantik vs. prototypensemantik - anspruch und leistung zweier grundkonzepte der lexikalischen semantik. Master's thesis, Universität Münster.

[Patwardhan et al., 2003] Patwardhan, S., Banerjee, S., & Pedersen, T. (2003). Using measures of semantic relatedness for word sense disambiguation. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics* (pp. 241–257). Mexico City, Mexico.

[Pedersen, 2004] Pedersen, T. (2004). The duluth lexical sample systems in senseval-3. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 203–208). Barcelona, Spain: Association for Computational Linguistics.

[Pedersen et al., 2006] Pedersen, T., Kulkarni, A., Angheluta, R., Kozareva, Z., & Solorio, T. (2006). An unsupervised language independent method of name discrimination using second order co-occurrence features. In *Computational Linguistics and Intelligent Text Processing*, volume 3878/2006 of *Lecture Notes in Computer Science* (pp. 208–222).: Springer Berlin / Heidelberg.

[Pedersen & Mihalcea, 2005] Pedersen, T. & Mihalcea, R. (2005). Advances in word sense disambiguation. In *43rd Annual Meeting of the Association for Computational Linguistics* University of Michigan, Ann Arbor, USA.

[Porter, 1980] Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3), 130–137.

[Preiss, 2004] Preiss, J. (2004). Probabilistic wsd in senseval-3. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 213–216). Barcelona, Spain: Association for Computational Linguistics.

[Purandare & Pedersen, 2004] Purandare, A. & Pedersen, T. (2004). Word sense discrimination by clustering contexts in vector and similarity spaces. In *Proceedings of the Conference on Computational Natural Language Learning (CoNLL)* Boston, MA.

[Quillian, 1969] Quillian, M. R. (1969). The teachable language comprehender: a simulation program and theory of language. *Commun. ACM*, 12(8), 459–476.

[Ramakrishnan & Bhattacharyya, 2003] Ramakrishnan, G. & Bhattacharyya, P. (2003). Text representation with wordnet synsets using soft sense disambiguation. In *8th International Conference on Applications of Natural Language to Information Systems* Burg (Spreewald), Germany.

[Ramakrishnan et al., 2004a] Ramakrishnan, G., Prithviraj, B., & Bhattacharya, P. (2004a). A gloss-centered algorithm for disambiguation. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 217–221). Barcelona, Spain: Association for Computational Linguistics.

[Ramakrishnan et al., 2004b] Ramakrishnan, G., Prithviraj, B., Deepa, A., Bhattacharyya, P., & Chakrabarti, S. (2004b). Soft word sense disambiguation. In *International Conference on Global Wordnet (GWC 04)* Brno, Czeck Republic.

[Ramsay, 1985] Ramsay, A. (1985). Effective parsing with generalized phrase structure grammar. In *ACL Proceedings, Second European Conference* (pp. 57–61).

[Resnik & Yarowsky, 1997] Resnik, P. & Yarowsky, D. (1997). *Tagging Text with Lexical Semantics: Why, What and How?*, chapter A perspective on word sense disambiguation methods and their evaluation, (pp. 79–86). SIGLEX (Lexicon Special Interest Group) of the ACL.

[Resnik & Yarowsky, 1999] Resnik, P. & Yarowsky, D. (1999). Distinguishing systems and distinguishing senses: new evaluation methods for word sense disambiguation. *Nat. Lang. Eng.*, 5(2), 113–133.

[Rigau et al., 1997] Rigau, G., Atserias, J., & Agirre, E. (1997). Combining unsupervised lexical knowledge methods for word sense disambiguation. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics* (pp. 48–55). Morristown, NJ, USA: Association for Computational Linguistics.

[Rodget, 1852] Rodget, P. (1852). *Thesaurus of English Words and Phrases*. Longman.

[Sahlgren, 2005] Sahlgren, M. (2005). An introduction to random indexing. In *Proceedings of Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering* Copenhagen, Denmark.

[Salton & McGill, 1983] Salton, G. & McGill, M. (1983). *Introduction to modern Information Retrieval*. Number ISBN:0070544840. New York, NY: McGraw-Hill.

[Salton & Yu, 1975] Salton, G. & Yu, C. T. (1975). *Effective Information Retrieval using Term Accuracy*. Technical Report 75-249, Department of Computer Science, Cornell University, Ithaca, New York.

[Santamaria et al., 2003] Santamaria, C., Gonzalo, J., & Verdejo, M. F. (2003). Automatic association of web directories to word senses. *Computational Linguistics*, 29(3).

[Schmid, 1996] Schmid, U. (1996). *Kognitive Modellierung. Eine Einf?hrung in logische und algorithmische Grundlagen*. Heidelberg, Berlin, Oxford: Spektrum Akademischer Verlag GmbH.

[Schütze, 1995] Schütze, H. (1995). *Ambiguity and Language Learning: Computational and Cognitive Models*. PhD thesis, Stanford University.

[Schütze, 1998] Schütze, H. (1998). Automatic word sense discrimination. *Comput. Linguist.*, 24(1), 97–123.

[Searle, 1971] Searle, J. (1969 / dt. 1971). *Speech Acts: An Essay in the Philosophy of Language Dt. Übersetzung: Sprechakte: Ein sprachphilosophischer Essay*. Frankfurt am Main: Cambridge University / dt. Suhrkamp.

[Searle, 1997] Searle, J. R. (1997). Sprechakte: Ein sprachphilosophischer essay. Suhrkamp Taschenbuch Wissenschaft, Frankfurt a.M.

[Sebastiani, 2002] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1), 1–47.

[Seo et al., 2004] Seo, H.-C., Rim, H.-C., & Kim, S.-H. (2004). Kunlp system in senseval-3. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 222–225). Barcelona, Spain: Association for Computational Linguistics.

[Sleator & Temperley, 1991] Sleator, D. & Temperley, D. (1991). *Parsing English with a Link Grammar*. Computer Science technical report CMU-CS-91-196, Carnegie Mellon University.

[Soderland, 1999] Soderland, S. (1999). Learning information extraction rules for semi-structured and free text. *Machine Learning*, 34, 233–272.

[Soderland et al., 1995] Soderland, S., Fisher, D., Aseltine, J., & Lehnert, W. (1995). Crystal: Inducing a conceptual dictionary. In C. Mellish (Ed.), *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (pp. 1314–1319). San Francisco.

[Stevenson & Wilks, 2001] Stevenson, M. & Wilks, Y. (2001). The interaction of knowledge sources in word sense disambiguation. *Computational Linguistics*, 27(3), 321–349.

[Strapparava et al., 2004] Strapparava, C., Gliozzo, A., & Giuliano, C. (2004). Pattern abstraction and term similarity for word sense disambiguation: Irst at senseval-3. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 229–234). Barcelona, Spain: Association for Computational Linguistics.

[Towell & Voorhees, 1998] Towell, G. & Voorhees, E. M. (1998). Disambiguating highly ambiguous words. *Comput. Linguist.*, 24(1), 125–145.

[van Rijsbergen, 1979] van Rijsbergen, C. J. (1979). *Information Retrieval*. London, Boston: Butterworth-Heinemann.

[Vázquez et al., 2004] Vázquez, S., Romero, R., Suárez, A., Montoyo, A., García, M., Martín, M. T., García, M. A., & Urena, L. A. (2004). The r2d2 team at senseval-3. In R. Mihalcea & P. Edmonds (Eds.), *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text* (pp. 248–252). Barcelona, Spain: Association for Computational Linguistics.

[Velardi et al., 1991] Velardi, P., Fasolo, M., & Pazienza, M. T. (1991). How to encode semantic knowledge: a method for meaning representation and computer-aided acquisition. *Computational Linguistics*, 17(2), 153–170.

[Volkmann, 1999] Volkmann, L. (1999). *Fundamente der Graphentheorie*. Springer Verlag Wien/New York.

[Wallner, 1983] Wallner, F. (1983). Läßt sich die sprechakttheorie als eine präzisierende fortführung und systematische ausarbeitung von wittgensteins später philosophie verstehen? In *Der Mensch und die Wissenschaften vom Menschen* (pp. 1033 – 1040). Innsbruck: G. Frey and J. Zelger.

[Widdows, 2003] Widdows, D. (2003). Unsupervised methods for developing taxonomies by combining syntactic and statistical information. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology* (pp. 197–204). Morristown, NJ, USA: Association for Computational Linguistics.

[Widdows, 2004] Widdows, D. (2004). *Geometry and Meaning*. Center for the Study of Language and Inf.

[Widdows & Peters, 2003] Widdows, D. & Peters, S. (2003). Word vectors and quantum logic - experiments with negation and disjunction. In *Proceedings of Mathematics of Language*, number 8: Stanford University.

[Wilks, 1975] Wilks, Y. (1975). An intelligent analyzer and understander of english. *Commun. ACM*, 18(5), 264–274.

[Winnemöller, 2001] Winnemöller, R. (2001). Organisation von e-mails. Master's thesis, Universität Hamburg, Hamburg.

[Winnemöller, 2004] Winnemöller, R. (2004). Constructing text sense representations. In G. Hirst & S. Nirenburg (Eds.), *ACL 2004: Second Workshop on Text Meaning and Interpretation* (pp. 17–24). Barcelona, Spain: Association for Computational Linguistics.

[Winnemöller, 2005] Winnemöller, R. (2005). Knowledge based feature engineering using text sense representation trees. In *International Conference RANLP - 2005* Borovets, Bulgaria.

[Winnemöller, 2008] Winnemöller, R. (2008). Using meaning aspects for word sense disambiguation. In *9th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)* Haifa, Israel.

[Witten & Frank, 2000] Witten, I. H. & Frank, E. (2000). *Data Mining: Practical machine learning tools with Java implementations*. Morgan Kaufmann, San Francisco.

[Witten & Frank, 2005] Witten, I. H. & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. San Francisco: Morgan Kaufmann, 2 edition.

[Wittgenstein, 1984a] Wittgenstein, L. (1984a). *Werkausgabe Bd. I*, chapter Tractatus Logico Philosophicus. Suhrkamp Verlag, Frankfurt am Main.

[Wittgenstein, 1984b] Wittgenstein, L. (1984b). *Werkausgabe Bd. I*, chapter Philosophische Untersuchungen. Suhrkamp Verlag, Frankfurt am Main.

[Yahoo Inc., 2004] Yahoo Inc. (2004). Yahoo. http://www.yahoo.com.

[Yang et al., 2005] Yang, R., Kalnis, P., & Tung, A. K. H. (2005). Similarity evaluation on tree-structured data. In *SIGMOD '05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data* (pp. 754–765). New York, NY, USA: ACM Press.

[Yang et al., 2003] Yang, Y., Zhang, J., & Kisiel, B. (2003). A scalability analysis of classifiers in text categorization. In *ACM SIGIR* (pp. 96–103).

[Yarowsky, 1992a] Yarowsky, D. (1992a). Word-sense disambiguation using statistical models of roget's categories trained on large corpora. In *Proceedings of the 14th conference on Computational linguistics* (pp. 454–460). Morristown, NJ, USA: Association for Computational Linguistics.

[Yarowsky, 1992b] Yarowsky, D. (1992b). Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. In *Proceedings of COLING-92* (pp. 454–460). Nantes, France.

[Yarowsky, 1993] Yarowsky, D. (1993). One sense per collocation. In *Proceedings of the ARPA Workshop on Human Language Technology* San Francisco, CA: Morgan Kaufman.

[Yarowsky, 1994] Yarowsky, D. (1994). Decision lists for lexical ambiguity resolution: Application to accent restoration in spanish and french. In *Meeting of the Association for Computational Linguistics* (pp. 88–95).

[Yarowsky, 1995] Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. In *Meeting of the Association for Computational Linguistics* (pp. 189–196).

Index

Eidesstattliche Erklärung

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Dissertation

Zur bedeutungsorientierten
Auflösung von
Wortmehrdeutigkeiten

Vorschlag einer Methodik

selbst verfasst und keine anderen als die angegebenen Hilfsmittel verwendet habe. Ferner versichere ich, an keiner anderen Hochschule einen Antrag auf Eröffnung eines Promotionsprüfungsverfahrens gestellt zu haben.

Hamburg, den 08. September 2008