
ONTOLOGIE
BASIERTE
MONOSEMIERUNG
joachim kleb
KIT Scientific Publishing

Joachim Kleb

Ontologie-basierte Monosemierung

Ontologie-basierte Monosemierung

von
Joachim Kleb

KIT Scientific Publishing

Dissertation, Karlsruher Institut für Technologie (KIT)
Fakultät für Wirtschaftswissenschaften, 2012
Referenten: Prof. Dr. Rudi Studer, Prof. Dr. Andreas Geyer-Schulz

Impressum

KIT Scientific Publishing

Karlsruher Institut für Technologie (KIT)
KIT Scientific Publishing
Straße am Forum 2
D-76131 Karlsruhe

www.ksp.kit.edu

Print on Demand 2015

ISBN 978-3-86644-958-9
DOI 10.5445/KSP/1000031500

Ontologie-basierte Monosemierung

Bestimmung von Referenzen im Semantic Web

Zur Erlangung des akademischen Grades eines
Doktors der Wirtschaftswissenschaften

(Dr. rer. pol.)

von der Fakultät für
Wirtschaftswissenschaften
am Karlsruher Institut für Technologie

genehmigte

DISSERTATION

von

Dipl.-Inform.(FH) Joachim Kleb

Tag der mündlichen Prüfung: 09.02.2012
Referent: Prof. Dr. Rudi Studer
Korreferent: Prof. Dr. Andreas Geyer-Schulz

Zusammenfassung

Von Geburt an interagiert der Mensch mit der natürlichen Sprache. Er verständigt sich durch diese, nimmt dadurch Wissen auf und teilt dieses anderen mit. Die natürliche Sprache besitzt jedoch eine Eigenschaft, die den Austausch von Wissen erschwert - Ambiguität. Derselbe Name kann eine Vielzahl unterschiedlicher Objekte identifizieren, *z.B.* „Blatt“ als Bezeichner des Bestandteils einer Pflanze oder als Papier. Zur Ermittlung der Bedeutung sind weitere Merkmale nötig, die im Zusammenhang mit dem Begriff beschrieben werden, *z.B.* „Blatt einer Pflanze“. Dieser Kontext wird somit zur Monosemierung benötigt.

Ontologien, die in modernen Anwendungssystemen als Hintergrundwissen zum Einsatz kommen, verlangen zwingend die eindeutige Identifikation der darin beschriebenen Elemente und umgehen somit das Problem der Mehrdeutigkeit. Ontologien finden immer weitere Verbreitung, da sie die Möglichkeit bieten, Wissen durch ein Netzwerk logischer Relationen zu repräsentieren. Dadurch steigt auch die Notwendigkeit, Informationen in natürlicher Sprache in Ontologien zu integrieren bzw. mit den darin bereits enthaltenen Informationen in Zusammenhang zu bringen.

Die vorliegende Arbeit beschäftigt sich mit dieser Thematik, insbesondere mit der Problematik der Ambiguität, die bei der Zusammenführung natürlich-sprachlicher Informationen mit dem durch Ontologien repräsentierten Wissen auftritt. Diese ist darauf zurückzuführen, dass der gleiche natürlich-sprachliche Bezeichner mehrere Elemente der Ontologie bezeichnen kann. Die Arbeit stellt eine grundlegende Methodik vor, wie dennoch das passende Ontologieelement bestimmt werden kann. Der Ansatz orientiert sich am menschlichen Vorgehen und an der Verwendung des Kontextes zur Monosemierung. Dieser stellt den Zusammenhang zwischen den Entitäten dar, die im Text erwähnt werden. Die Arbeit bildet diesen textuellen Kontext anhand der Zusammenhänge innerhalb

des Ontologiegraphen nach und nimmt durch dessen Analyse eine Disambiguierung vor. Hierzu wurde ein auf Spreading-Activation basierendes Verfahren entworfen. Zusätzlich werden verschiedene Modifikationen des Verfahrens zur Steigerung der Ergebnisqualität vorgestellt, *z.B.* hinsichtlich des Graphaufbaus, des Einsatzes von maschinellem Lernen und verschiedener Gewichtungsverfahren. Die Güte des Verfahrens wird mittels eines umfangreichen Evaluationsdatensatz und im Vergleich mit anderen Verfahren nachgewiesen.

Danksagung

Das Schreiben einer Dissertation erstreckt sich über viele Monate. Während dieser Zeit ist man oftmals mit verschiedensten Fragestellungen konfrontiert, die für ein Weiterkommen beantwortet werden müssen. Neben der intensiven Recherche sind persönliche Gespräche eine äußerst wertvolle Quelle, die ein Weiterkommen auch in schwierigen Fragestellungen gewährleisten. Zudem ermöglicht erst die Unterstützung und der Rückhalt vieler Personen das Ziel zu erreichen und somit die Arbeit fertigzustellen.

Zunächst möchte ich mich bei meinem Mentor Prof. Rudi Studer bedanken, der mir die Möglichkeit bot diese Dissertation im Rahmen meiner Tätigkeit bei der Gruppe Wissensmanagement am Forschungszentrum Informatik (FZI) zu erstellen. Insbesondere möchte ich mich herzlich bei Dr. Andreas Abecker bedanken, der mir sehr viel Gelegenheit zur Diskussion von fachlichen Gegebenheiten bot und mir fortwährend mit großem persönlichen Rückhalt unterstützend zur Seite stand. Zudem möchte ich mich bei Dr. Sebastian Rudolph bedanken, der mir in mathematischen Fragen mittels Diskussion zur Seite stand.

Auch möchte ich mich bei meinen direkten Kollegen Dr. Tuvshintur Tserendorj, Jürgen Bock, Veli Bicer sowie Jens Wissmann bedanken, die mir ebenfalls viel Raum zu Gesprächen boten. Ebenfalls gilt mein Dank meinen studentischen Mitarbeitern.

Insbesondere jedoch möchte ich mich jedoch bei meiner Tochter Madita bedanken, die mir immer wieder gezeigt hat, dass es auch ein Leben neben der Dissertation gibt.

Joachim Kleb

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1. Ziele und Aufbau der Arbeit

1.1. Motivation und Hintergrund

Der Zugriff, die Speicherung und der Austausch von Information, die in Form von Gesprächen, Dokumenten und Büchern, Fernsehsendungen, des Internets sowie vergleichbaren Quellen transportiert wird, bildet die Wissensgrundlage in unserer heutigen Welt. Jeder Mensch macht sich einen Anteil dieses Wissens zu eigen, auf das er zugreifen kann. Beschäftigt er sich mit neuen Informationen aus den zuvor genannten Medien, ist es ihm möglich auf sein bereits vorhandenes Wissen zuzugreifen. Dieses kann somit weiterentwickelt, in Frage gestellt und gegebenenfalls korrigiert werden. Hierfür verfügt der Mensch über die Gabe in Texten enthaltene Information oder das in Gesprächen transportierte Wissen kognitiv verarbeiten zu können, *d.h.* zunächst ein Modell der darin enthaltenen Zusammenhänge zu erstellen. Dieses erlaubt es ihm, die enthaltenen Informationen mit dem von ihm bereits zuvor erlernten Wissen in Verbindung zu bringen und somit auswertbar zu machen.

Seit Mitte des letzten Jahrhunderts wurde die Speicherung von Informationen in digitaler Form vorangetrieben. Diese ist heutzutage nicht mehr wegzudenken und digitale Informationsquellen, wie *z.B.* das Internet, sind beinahe jedermann zugänglich und werden als Alternative zu gedruckten, *d.h.* in Papierform vorliegenden, Quellen angesehen. Dies erlaubt zusätzlich zum Zugriff durch den Menschen auch den maschinellen Zugriff durch Computerprogramme. Jedoch ist die zuvor beschriebene menschliche Interpretation des darin enthaltenen Wissens der maschinellen zum heutigen Zeitpunkt in vielen Bereichen überlegen. Die Weiterentwicklung der maschinellen Verarbeitung hat insofern eine Annäherung an die Fähigkeiten der menschlichen Informationsauswertung bzw. gar eine Verbesserung dieser zum Ziel. Letzteres gelingt bereits heute in speziellen Bereichen, *z.B.* die Sichtung

von sehr großen Mengen an Daten (bezüglich vorgegebener Kriterien), die mittels optimierter Algorithmen durch Computerprogramme wesentlich schneller abgeschlossen werden kann. Oftmals sind diese jedoch limitiert auf ein restriktives, auf die spezielle Anwendung hin optimiertes Regelwerk. Um jedoch der menschlichen Vorstellungskraft näher zu kommen, ist zunächst als Grundlage eine strukturierte digitale Speicherung von Daten notwendig. Insbesondere ist eine Abbildung der Art und Weise notwendig, wie diese Daten zueinander in Verbindung stehen. Der Forschungsbereich des Semantic Web beschäftigt sich mit der Erzeugung und Umsetzung einer solchen, wohl-definierten Wissensbasis, die den Informationsaustausch zwischen Computern sowie zwischen Computer und Mensch verbessern soll.

Wie zuvor erwähnt, liegt jedoch der überwiegende Anteil des zur Verfügung gestellten Wissens, *z.B.* Artikel, Bücher *etc.*, in unstrukturierter natürlich-sprachlicher[1] Form vor. Hieraus resultiert, dass die automatische Verarbeitung von Sprache und die Strukturierung des damit vermittelten Wissens bzw. die Verknüpfung dieses Wissens mit bereits existierendem Wissen eine der wichtigsten Aufgaben in der elektronischen Verarbeitung von Information darstellt. In Texten befindet sich eine Vielzahl von verschiedensten Informationen. Bei näherer Analyse dieser Informationen stellt sich heraus, dass insbesondere die Nennung von Aspekten, wie beispielsweise Personen, Orte *etc.*, das durch den Text transportierte Wissen charakterisiert. Diese Aspekte können als Ankerpunkte betrachtet werden, *d.h.* als Elemente, auf die Informationen in diesem und/oder in weiteren Texten Bezug nehmen. Diese werden unter dem Begriff (benannte) Entitäten[2] zusammengefasst.

Im Fokus dieser Arbeit steht der Zusammenhang zwischen semantischen Wissensmodellen (Ontologien) in der Informatik und Entitäten, die innerhalb von natürlich-sprachlichen Quellen erwähnt werden. Der Aufbau eines solchen Zusammenhangs beinhaltet die Interpretation natürlich-sprachlicher Begriffe. Das ist jedoch mit Risiken verbunden. Hierbei ist die Ambiguität (Mehrdeutigkeit) von Begriffen an erster Stelle zu nennen. Beispielsweise sind dem Wort „Läufer“ bis

[1] *Natürliche* Sprache und *menschliche* Sprache werden in dieser Arbeit als synonyme Begriffe erachtet.

[2] „Entität“ bezeichnet einen ontologischen Sammelbegriff für alles Seiende, *z.B.* Personen, Organisationen *etc.* Der Begriff wird in Kapitel 4 erläutert.

zu 24 verschiedene Bedeutungen im Deutschen zugeordnet. Unter anderem bezeichnet dieses Wort eine Art von Teppich, eine Figur beim Schach sowie einen Sportler, der Laufsport betreibt *etc.* Ein Beispiel für einen mehrdeutigen Entitätsbezeichner ist die Existenz zweier Orte mit dem Namen „Mölln“ in Deutschland. Ein Beispiel für die Gefahren, die von Mehrdeutigkeiten ausgehen können, zeigt die Verwechslung beider Orte, die 2009 stattfand. Diese führte zu falschen Einträgen im Einwohnermelderegister [127]. Der Bürgermeister der Gemeinde Mölln (Amt Stavenhagen) stellte fest, dass der Eintrag im Einwohnermelderegister um 100 Einträge vom tatsächlichen Wert abwich. Dies konnte auf eine Verwechslung der Stadt Mölln (Schleswig-Holstein) mit Mölln (Stavenhagen) durch das zuständige Amt in Hamburg zurückgeführt werden.

Ein Grund für die Mehrdeutigkeit von Begriffen ist der beschränkte Namensraum, den Menschen im Alltag verwenden. Polysemie[3] kann sogar als *„Normalfall“* [172] in der Sprache betrachtet werden. Ambiguität findet sich in allen Bereichen der Sprache wieder. Jedoch sind nicht nur Wörter des alltäglichen Sprachgebrauchs betroffen, sondern – wie im Fall „Mölln“ – auch Entitäten. Dies wird unter anderem deutlich bei der Betrachtung der Geburtsstatistik für Deutschland. Die Universität Leipzig stellte fest, dass allein im Jahr 2007 1,98% der 83.886 geborenen Mädchen der Vorname „Marie“ gegeben wurde und somit zirka 1.661 Mädchen dieses Jahrgangs den gleichen Vornamen haben [191]. Insgesamt kamen auf 382.360 Kinder nur 4.164 verschiedene Vornamen. Um eine solche Mehrdeutigkeit aufzulösen, müssen Kontextinformationen zusätzlich zu einem ambiguen Bezeichner verfügbar sein. Beispielsweise kann die Nennung des Nachnamens eines Mädchens mit dem Vornamen „Marie“ oder das zusätzliche Erwähnen eines Freundes eine eindeutige Referenz zur gemeinten Person ermöglichen.

Die folgende Arbeit nimmt Bezug auf die Auswirkung von Mehrdeutigkeit in Wissensmodellen. Sie beschäftigt sich insbesondere mit Wissensmodellen des Semantic Web. Im Vordergrund stehen die Definition von Ambiguität einer Ontologie, sowie die Vorstellung eines Verfahrens zur Disambiguierung mehrdeutiger Entitäten, die in Texten genannt werden. Die Aufgabe ist somit die Bestimmung der

[3] Polysemie bezeichnet eine spezielle Art von Ambiguität. Diese wird in Abschnitt 5.2 eingeführt.

korrekten Referenz zum mehrdeutigen Bezeichner im Rahmen einer textuellen Analyse.

1.2. Forschungsfragen und Ziele

Wissensbasen in Form von Ontologien finden heutzutage zunehmend Verbreitung im World Wide Web. Es existieren zahlreiche Webdienste, die ihre Daten in Form von Ontologien zur Verfügung stellen, *z.B.* DBpedia[4], GeoNames Ontology[5] *etc.* Ontologien erlauben unter anderem die Beschreibung semantischer Zusammenhänge zwischen Entitäten, *z.B.* Personen, Organisationen *etc.* sowie deren konzeptuelle Beziehungen. Inhalte in Ontologien, wie beispielsweise Entitäten, werden anhand von Schlüsseln auf Grundlage des Unified Ressource Identifier beschrieben und sind damit eindeutig zu identifizieren. Der Mensch, der Autor der meisten Ontologien, folgt seiner natürlichen Sprache zur Bezeichnung vieler Ontologieelemente. Mit dem Abgleich natürlich-sprachlicher Bezeichner aus externen Quellen mit den Bezeichnern innerhalb der Ontologie findet die Ambiguität ebenfalls Einzug, da diese ein wesentliches Merkmal menschlicher Sprache ist. Wird beispielsweise der in einem Dokument enthaltene Namen des Ex-Bundeskanzlers „Helmut Schmidt“ in DBpedia gesucht, so kann dieser acht verschiedenen Personen zugeordnet werden.[6] Jegliche Person sowie jegliches Informationssystem steht Folge dessen unstrittig vor dem Problem auf Grundlage weiterer Informationen den im Text referenzierten „Helmut Schmidt“ in der Wissensbasis zu bestimmen.

Die vorliegende Arbeit basiert auf der nachfolgend genannten Hypothese:

Kernhypothese. *Die Disambiguierung mehrdeutiger Entitäten kann mit Hilfe von ontologischen Zusammenhängen erreicht werden.*

Hierbei ist der Zusammenhang zwischen textueller und ontologischer Information von wesentlicher Bedeutung. Die Kernhypothese dieser

4 `http://dbpedia.org` [letzter Zugriff am 12.09.2011]

5 `http://www.geonames.org/ontology/` [letzter Zugriff am 12.09.2011]

6 `http://de.wikipedia.org/wiki/Helmut_Schmidt_(Begriffsklärung)` [letzter Zugriff am 12.09.2011]

Arbeit ist die gegenseitige Ergänzung dieser beiden Informationsquellen, *d.h. a)* der zu analysierende Text und *b)* die Ontologie. In einem gegebenen Kontext wird mittels der Verwendung beider Informationsquellen eine Bestimmung der gesuchten Entitäten aufgrund ihrer Namensnennung möglich. Das Problem ist die Masse von Entitäten innerhalb der Ontologie, die demselben Namen entsprechen und somit im Grad der Mehrdeutigkeit der aufgefundenen Begriffe.

Die Grundannahme dieser Forschungsarbeit basiert auf der Beobachtung des menschlichen Prozesses zur Auflösung ambiguer Personen, Organisationen *etc.* Dieser eigentlich dem Menschen gegenüber unbewusst ablaufende Prozess basiert auf bereits vorhandenen und somit bereits bekannten Informationen, *z.B.* durch eigenes Erleben oder Analyse verschiedener Quellen (Zeitungen, Mitmenschen *etc.*) gewonnenen Wissens. Der Mensch verknüpft die aktuelle Situation mit diesem Wissen und wertet die sich dadurch ergebenden Zusammenhänge aus (vgl. Ansätze aus dem Bereich der Sprachwissenschaft [7, 206] und dem Bereich der Psycholinguistik[7] [4, 216]). In dieser Arbeit wird eine textuell beschriebene Situation *z.B.* ein Zeitungsartikel, mit einer vorliegenden Ontologie verknüpft und auf deren Grundlage wird das Nachverfolgen der im Text dargestellten Zusammenhänge vorgenommen und später ausgewertet.

Die Kerhypothese wirft wissenschaftliche Fragestellungen auf, die teilweise zuvor nicht aufgestellt oder nicht im thematischen Kontext der ontologiebasierten Entitätsdisambiguierung angegangen bzw. gelöst wurden. Im Folgenden werden diese Fragestellungen einzeln benannt und vorgestellt:

Forschungsfrage 1. *Wie wird linguistische[8] Ambiguität im ontologischen Modell reflektiert?*

Diese Frage ist richtungweisend für die vorliegende Arbeit. Sie wirft die Problematik auf, wie sich Ambiguität im ontologischen Modell äußert und erschließt somit den Zusammenhang zwischen der linguistischen und der ontologischen Ambiguität. Die Linguistik ist auf die

[7] Die Psycholinguistik beschreibt ein Teilgebiet der Psychologie, das sich mit der Analyse von Sprachfunktionen, Sprachentwicklung, sprachlicher Kommunikation und deren Regeln und der Interdependenz von Sprache beschäftigt.

[8] „Linguistik“ kommt aus dem Lateinischen („Lingua“ = „Zunge“) und ist ein Synonym für Sprachwissenschaft.

menschliche Sprache fokussiert und nimmt sich hierbei des Problems der Mehrdeutigkeit, der sogenannten Äquivokation[9] (siehe [125]), an. Ausgehend von der Analyse eines natürlich-sprachlichen Textes ist die Frage des Zusammenhangs zwischen dem allgemeinen Ambiguitätsbegriff und der in einer Ontologie vorkommenden Ambiguität grundlegend für weitere Forschungen in diesem Bereich. Das betrifft insbesondere die Übertragung des von Pethöe [174] beschriebenen Modells von Polysemie[10] auf Ontologien.

Im Abschnitt 1.1 wird der Rückgriff des Menschen auf Hintergrundinformationen zur Auflösung der vorhandenen Ambiguität angesprochen. Wie jedoch kann auf solche Informationen zugegriffen werden? Hieraus ergibt sich die nachfolgende Forschungsfrage:

Forschungsfrage 2. *Wie können semantische Zusammenhänge zwischen Entitäten in einer Ontologie lokalisiert und für eine Referenzauflösung bewertet werden?*

Der genannte menschliche Erfahrungsschatz, der als Grundlage zur Auflösung von ambiguen Begriffen verwendet wird, wird in der vorliegenden Arbeit durch eine Ontologie ausgedrückt. Die Zusammenhänge zwischen Entitäten, die Ausgangsbasis einer möglichen Disambiguierung sind, müssen demzufolge in der Ontologie enthalten bzw. lokalisierbar sein. Im Fokus dieser Arbeit liegt jedoch nicht die Integration neuer Hintergrundinformationen in die Wissensbasis, sondern dessen Gegenteil, *d.h.* die Extraktion von Zusammenhängen zwischen den im Kontext genannten Entitäten aus der Wissensbasis. Nur auf Grundlage von Informationen über gegenseitige Beziehungen kann ein solcher Zusammenhang erschlossen werden. Das Hintergrundwissen über die Entitäten ist deshalb von elementarer Bedeutung, um die im gegebenen Kontext referenzierte Entität im Falle mehrerer Möglichkeit zu bestimmen. Die Forschungsfrage beschäftigt sich daher mit der Aufgabe, die Entitäten in der Ontologie aufzufinden und die Zusammenhänge zwischen diesen auszuwerten. Das ist vergleichbar mit dem menschlichen Prozess zur Referenzbestimmung, da auch hier ein beliebiger Zusammenhang zur exakten Referenzauflösung nicht ausreicht. Zum Beispiel reicht

[9] Homonyme und Polyseme

[10] Bei Polysemie handelt es sich um eine spezielle Form von Ambiguität (siehe Abschnitt 5.2.1).

die Beschreibung „Helmut Schmidt wurde in Deutschland geboren“ und somit der Zusammenhang zwischen „Helmut Schmidt“ und „Deutschland“ nicht aus, um zwischen dem ehemaligen Kanzler „Helmut Schmidt“ und dem ehemaligen Fußballspieler „Helmut Schmidt“ unterscheiden zu können.

Somit ist die ausschließliche Lokalisierung dieser Zusammenhänge nicht ausreichend. Die Bewertung eines solchen Zusammenhangs in der Ontologie muss zunächst definiert und auf die Anforderungen, welche die Ambiguität mit sich bringt, angepasst werden. Eine Bewertung muss es zudem ermöglichen, das bestmögliche Resultat zu erzielen bzw. aufzufinden.

Forschungsfrage 3. *Wie kann die Qualität eines solchen Prozesses verbessert werden?*

Nachdem ein grundlegender Algorithmus zur Problemlösung entworfen wurde, stellt sich die Frage nach möglichen Optimierungen. Hierbei muss abgewogen werden, ob es Anwendungsfälle gibt, die durch eine Variation des Algorithmus besser gelöst werden können. Hierzu müssen die verschiedenen Teile des Verfahrens *a)* die Identifikation der zu berücksichtigenden Entitäten *b)* die Bedeutung der einzelnen Entität im Verfahren und *c)* mögliche Modifikationen im späteren Verfahrensablauf analysiert werden. Die Zielsetzung ist hierbei die Optimierung der Referenzselektion und somit der Disambiguierung der Entitäten.

1.3. Ansatz

Diese Arbeit nimmt sich der Aufgabe an, Entitäten, die innerhalb eines Textes beschrieben wurden, den korrekten Ontologiereferenzen zuzuordnen. Diese müssen anhand ihrer semantischen Beschreibung sich in die durch den Kontext des Textes beschriebene Situation einfügen. Der Auftrag, der zu bewältigen ist, umfasst die Zuordnung einer Entität zum entsprechenden Ontologieelement.

Hierzu sind verschiedene Teilschritte notwendig:

Auffinden von benannten Entitäten Im ersten Schritt findet eine Analyse des Textes statt. Die Aufgabe ist es die Bezeichner von möglichen Entitäten zu bestimmen, *d.h.* Entitäten zu identifizieren. Die Textanalyse berücksichtigt grammatikalische Konstrukte, *z.B.* Nomen, durch die Entitätsbezeichner identifiziert werden können.

Identifikation der Entitäten Im zweiten Schritt werden die erkannten Entitätsbezeichner den Ontologieelementen zugeordnet, die durch diese identifizierbar sind. Hierbei sind verschiedene Kriterien zu beachten, *z.B.* dass Ontologieelemente teilweise über mehrere Bezeichner verfügen *etc.*

Identifikation des Zusammenhangs zwischen Entitäten: Durch die Identifikation möglicher im Text beschriebener Ontologieelemente, *z.B.* der beschriebenen Instanzen, können die Zusammenhänge zwischen diesen in der Ontologie nachverfolgt werden. Durch das Problem der Ambiguität sind verschiedene Abbildungen der textuellen auf die ontologische Repräsentation möglich. Die Aufgabe dieser Arbeit besteht in der Identifikation der zutreffenden Abbildung und somit der Bestimmung der im Kontext des Textes zu referenzierenden Instanzen, *d.h.* ihrer Disambiguierung.

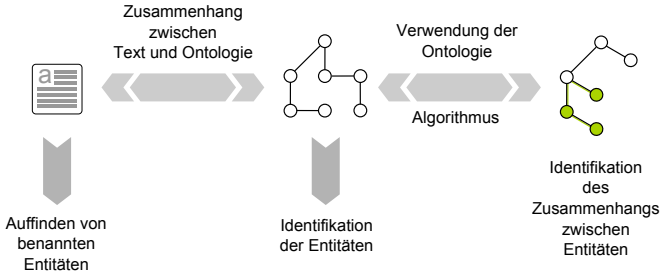


Abbildung 1.1.: Methodischer Ansatz dieser Arbeit

1.4. Eigener Beitrag

Der vorgestellte Ansatz verwendet eine Ontologie, die als Repräsentationsformat des Hintergrundwissens gewählt wurde. Hierbei wird der Anforderung an die Abbildung semantischer Zusammenhänge und der Möglichkeit zu deren Auswertung Rechnung getragen. Ontologien sind eine wichtige Quelle für die Disambiguierung, da sie die gegenseitigen Abhängigkeiten von Entitäten formalisieren. Sie ermöglichen die Beschreibung der direkten Eigenschaften einer Entität – Data-Properties – und der Kategorisierung – Object-Properties.

Zum einen fokussiert sich die vorliegende Arbeit auf das Phänomen der natürlich-sprachlichen Mehrdeutigkeit innerhalb von Ontologien. Diese tritt durch die Verwendung wortgleicher linguistischer Bezeichner für Elemente der Ontologie auf. Ein Beitrag der vorliegenden Arbeit ist die Erörterung des Phänomens der Ambiguität innerhalb von Ontologien. Es besteht großes wissenschaftliches Interesse an der Erforschung sprachlicher Mehrdeutigkeit (siehe [174]). Forschungsthemen waren die Einteilung in mögliche Kategorien, in welche sich Mehrdeutigkeit unterteilen lässt – Polysemie und Homonymie – sowie deren weitere Unterscheidungen, *z.B.* systematische und nicht-systematische Polysemie. Zudem wurden theoretische Vorgehensweisen zur Behandlung und Auflösung von Mehrdeutigkeit vorgeschlagen und erforscht. Bisher ist jedoch weder die Gesamtproblematik linguistischer Mehrdeutigkeit noch eines der genannten Modelle auf die Beschreibungen, die durch Ontologien ermöglicht werden, übertragen worden. Diese Arbeit nimmt sich dieses Umstandes an und definiert erstmals die ontologische Reflektion linguistischer Ambiguität. Zunächst werden die Grundlagen allgemein erörtert. Darauf aufbauend erfolgt im Verlauf der Arbeit eine Fokussierung auf die Ambiguität im Zusammenhang mit Entitäten.

Zum anderen wird in dieser Arbeit ein graphbasiertes Verfahren zur Disambiguierung von Entitäten in Texten mithilfe von Hintergrundwissen in Form einer Ontologie entwickelt. Dieser Beitrag umfasst die Entwicklung eines Graphexplorationsverfahrens basierend auf Spreading Activation [255]. Der überwiegende Anteil von Arbeiten im wissenschaftlichen Umfeld verwendet in diesem Zusammenhang vektorbasierte Verfahren und somit bekannte Machine Learning (ML)

Verfahren.[11] Die Gewinnung von Teilgraphen in Form von Steinerbäumen, die den im Kontext des Textes referenzierten Ausschnitt des semantischen Netzwerks und somit den dort dargestellten semantischen Zusammenhang beschreiben, bietet eine neue Herangehensweise an das Problem der Disambiguierung. Das Verfahren grenzt sich von der Keyword-basierten Suche ab, da es auf keinen übergebenen Schlüsselworten aufbaut. Bei dieser ist ein kontextueller Zusammenhang nicht in allen Fällen gegeben. Die Verwendung von Text, *z.B.* Artikeln oder Büchern, hat zur Folge, dass die beinhalteten Entitäten im Text selbst eine kontextuelle Beziehung miteinander eingehen, die in der Ontologie reflektiert wird. Ausgangsproblem des Verfahrens ist das Vorliegen von Mehrdeutigkeit im analysierten Text. Mehrdeutigkeit bezieht sich hier auf die Bedeutung eines Wortes und *nicht*[12] auf verschiedene Texte, in denen das Wort in der gleichen Form vorkommt. Hierdurch wird die Abgrenzung zur klassischen Suche offensichtlich. Die Forschungsarbeit beschäftigt sich weiterhin mit der Erweiterung des oben genannten Graphverfahrens auf Basis von Spreading Activation. Hierzu werden in der Arbeit drei verschiedene Optimierungsmöglichkeiten entwickelt und evaluiert. Jede dieser Möglichkeiten macht sich spezielle Gegebenheiten im vorgestellten Ansatz zu Nutze. Der Umstand, dass der Mensch dazu neigt in Beziehung stehende Informationen auch innerhalb von Texten kontextuell nahe beieinander zu platzieren, war Grundlage für die in der Arbeit vorgenommene Erweiterung durch eine kontextspezifische Disambiguierung [256].

Es wird untersucht, ob die Verwendung von Abstandsmaßen zur Verbesserung der Ergebnisqualität beiträgt. Hierzu wird analysiert, welche Relationen (Object-Properties) zwischen Entitäten das Verhältnis der Entitäten zueinander im Kontext der zugrundeliegenden Ontologie und des vorliegenden Textes am besten wiederspiegeln.

Auch wird die Optimierung des Verfahrens durch die Integration einer zusätzlichen Lernphase überprüft. Ein solches Lernverfahren ermöglicht eine Speicherung und Verwertung vergangener Disambiguierungsprozesse. Die daraus generierten Maße werden bei der Analyse weiterer bzw. zuvor unbekannter Texte verwendet. Insofern stützt

[11] Siehe Kapitel 14 für die genaue Abgrenzung der verfügbaren Verfahren.

[12] Worte in *kursiv* haben eine hinweisende oder klassifizierende Bestimmung.

sich das Verfahren bei der Generierung neuer Resultate auf vorausgehende Resultate.

Die angesprochene Tendenz des Menschen miteinander in Beziehung stehende Informationen nahe beieinander zu platzieren ist der Ausgangspunkt für zwei weitere Verfahren. Beide Verfahren basieren auf Regeln, die sich Wortmuster innerhalb des Textes zu Nutze machen. Im ersten Verfahren wird eine Anreicherung möglicher Bezeichner für Ontologieelemente anhand deren unterschiedlicher Nennung im Text sowie die Bestimmung konzeptioneller Abhängigkeiten [259] umgesetzt. Regeln können vorgegeben oder automatisch erlernt werden. Zudem ist die Bestimmung von Beziehungen zwischen Ontologieelementen sowie deren direkte Eigenschaften ebenfalls möglich [257]. Diese Informationen, die vom Text direkt den Elementen der Ontologie zugeordnet werden können, ermöglichen eine selektive Auswahl des Ontologieelements für einen textuellen Bezeichner.

Neben den Verfahren zur Lösung des Ambiguitätsproblems erfolgte die Entwicklung einer optimierten Datenbank für die Speicherung von OWL-Ontologien in den Ansätzen [254, 253].

1.5. Gliederung der Arbeit

In diesem Kapitel wird ein Überblick über den strukturellen Aufbau dieser Arbeit dargestellt. Der thematische Aufbau ist in Abbildung 1.2[13] wiedergegeben. Die Arbeit gliedert sich in vier Teilbereiche: *Grundlagen*, *Bestimmung von Entitätsreferenzen in RDF-Graphen*, *Evaluation und thematisch verwandte Arbeiten* sowie eine *Schlussbetrachtung*. Im Folgenden werden die einzelnen Teilbereiche vorgestellt.

Teil I - Grundlagen: Zu Beginn erfolgt eine Einordnung der Arbeit in den Prozess des „*Knowledge Discovery in Databases (KDD)*“ (Kapitel 2). Aufbauend auf diesem Prozess werden die grundlegenden Hintergrundinformationen eingeführt, auf denen diese Arbeit aufbaut. Das betrifft zum einen die Darstellung des Hintergrundwissens in Form

[13] Der Zusammenhang zwischen den Teilbereichen ist von links nach rechts dargestellt.

von einer Ontologie, *d.h.* eines gängigen Formats im Bereich des Semantischen Webs (Kapitel 3). Zum anderen betrifft es die Definition und Repräsentation von Entitäten. Im Rahmen der Kommunikation zwischen Menschen werden Entitäten in der natürlichen Sprache referenziert (Kapitel 4). Um den Zusammenhang zur Wissensbasis zu gewährleisten, muss jede referenzierte Entität zuvor in der Wissensbasis beschrieben worden sein. Die Verwendung semantischer Technologien im Zusammenhang mit Entitäten, unter dem Aspekt der Mehrdeutigkeit, bildet das Thema dieser Arbeit. Die Definition von Ambiguität wird in Kapitel 5 vorgestellt und dabei in die linguistische Definition von Mehrdeutigkeit auf Ontologien übertragen. Die hieraus resultierende Problemstellung ist die Begründung für die Notwendigkeit eines Lösungsansatzes, wie er in Teil II vorgestellt wird.

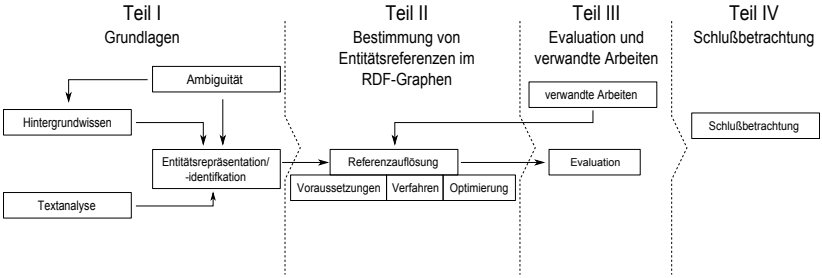


Abbildung 1.2.: Thematischer Aufbau

Teil II - Bestimmung von Entitätsreferenzen in RDF-Graphen: Der zweite Teil gliedert sich in drei Teilbereiche. Zunächst werden in Kapitel 6 die notwendigen Voraussetzungen für den Prozess der Referenzbestimmung bezüglich Text und Wissensbasen vorgestellt. In Kapitel 7 wird in die methodischen und technischen Verfahrensvoraussetzungen eingeführt. Auf diesen beruht der Ansatz der graphbasierten Disambiguierung, dessen Konzeption und Verfahrensablauf ebenfalls in diesem Kapitel vorgestellt werden. Im zweiten Teilbereich wird hierauf aufbauend das entworfene Basisverfahren beschrieben (siehe Kapitel 8). Dieser Teil enthält zudem die auf der Grundlage des Basisverfahrens weiterentwickelten Verfahren. Hierbei handelt es sich um ein Verfahren basierend auf einem bidirektionalen Wissensaustausch (Kapitel 9), ein Verfahren, das auf einer getrennten, kontextabhängigen

Analyse basiert (Kapitel 10) und ein Verfahren, das bestärkendes Lernen zur Resultatoptimierung einsetzt (Kapitel 11). Der dritte Teilbereich umfasst die Vorstellung variierbarer Maße und Parameter, die innerhalb des Algorithmus zum Einsatz kommen (Kapitel 12).

Teil III - Evaluation und thematisch verwandte Arbeiten: Der dritte Teil umfasst zunächst die Evaluation des vorgestellten Basisansatzes und dessen Varianten (Kapitel 13). Diese Evaluation beinhaltet zum einen einen Vergleich gegenüber einem existierenden Ansatz im Bereich der ontologiebasierten Disambiguierung. Zum anderen werden die in dieser Arbeit vorgestellten Algorithmusvarianten anhand eines manuell annotierten Referenzdatensatzes evaluiert. Dieser umfangreiche Datensatz ermöglicht die Gegenüberstellung der verschiedenen Varianten hinsichtlich ihrer Vor- und Nachteile. Außerdem ist eine Übersicht zu verwandten Arbeiten in Kapitel 14 enthalten. Deren jeweilige Unterschiede und Gemeinsamkeiten zu dieser Arbeit werden weiterhin erörtert.

Teil IV - Schlussbetrachtung: Die Arbeit wird mit Kapitel 15 abgeschlossen indem der erzielte Mehrwert dieser Arbeit beschrieben wird. Es erfolgt eine Reflektion des wissenschaftlichen Beitrags, der von dieser Arbeit ausgeht. Mögliche Erweiterungen des Ansatzes werden vorgestellt und dadurch Ideen für mögliche Weiterentwicklungen gegeben.

Teil I.

Grundlagen

2. Ermittlung von Wissen

Eine wesentliche Charakteristik unserer Zeit ist die ständig wachsende Informationsflut, die uns umgibt. Wir sind in nahezu allen Bereichen unseres Lebens mit digitalen Daten konfrontiert, *z.B.* Bankdaten, private und geschäftliche Emails, Einkäufe bei Onlinehändlern *etc.* Viele dieser Daten sind entweder Quelle oder Resultat komplexer Auswertungsprozesse. Beispielsweise bedarf bereits die Anzeige der Kundenrezension eines Artikels beim Onlinehändler Amazon[1] der Auswertung aller getätigten Einzelbewertungen für diesen Artikel. Während diese Auswertung noch recht offensichtlich ist, *d.h.* der Wert der Beurteilung durch den Durchschnitt der Einzelbewertungen berechnet wird, werden auch komplexere Verfahren zur Auswertung von Information durchgeführt. Ein Beispiel ist die automatische Kategorisierung von Nachrichtentexten, *z.B.* der Zuordnung eines Artikels zur Kategorie Politik, Sport *etc.* Hier erfolgt eine Textanalyse, die anhand von enthaltenen Wörtern über die Kategoriezugehörigkeit entscheidet. Um solche Auswertungsprozesse zu ermöglichen, sind Technologien zur Ermittlung versteckten Wissens notwendig. Erfolgt keine Analyse zur Gewinnung (neuer) abgeleiteter Informationen, so wird nicht der gesamte Wert genutzt, den diese Daten beinhalten (vgl. [74, 214, 225]). Der in dieser Arbeit vorgestellte Ansatz dient der Ermittlung von zusätzlichen impliziten Wissens und ermöglicht somit die Gewinnung eines Mehrwerts gegenüber der explizit zugreifbaren Dateninformation.

Zunächst wird in den Vorgang der Wissensgewinnung (Knowledge Discovery) in Abschnitt 2.1 eingeführt. Dieser lässt sich in einen mehrstufigen Prozess übertragen, der im darauf folgenden Abschnitt 2.2 vorgestellt wird. Jeder Prozessschritt enthält jeweils eine Erklärung seiner auf das vorliegende Verfahren bezogenen Umsetzung. Abschnitt 2.3 stellt die verschiedenen Zielsetzungen im Bereich

[1] `http://www.amazon.com` [letzter Zugriff am 10.01.2011]

Data-Mining vor. Hier erfolgt ebenfalls eine Einordnung des in dieser Arbeit vorgestellten Verfahrens.

2.1. Knowledge Discovery

Erstmals erwähnt wurde Knowledge Discovery in Databases (KDD) in der Arbeit von Piatetsky-Shapiro 1991 [176]. Derzeit wird die auf dieser Erklärung aufbauende Definition von Fayyad, Piatetsky-Shapiro und Smyth verwendet: *„Knowledge Discovery in databases*[2] *is the nontrivial process of identifying valid, novel, potentially useful and ultimately understandable patterns in data“* [74].

Durch den Einsatz von KDD-Methoden soll es ermöglicht werden, neue Informationen aus den Daten zu gewinnen. Dies bedeutet, dass neue Strukturen durch den Prozess erkannt werden, welche dem System beziehungsweise dem Benutzer zuvor unbekannt waren und sich nicht auf die Struktur beziehen, welche für die Speicherung des Datensatzes zunächst entwickelt wurde. Man spricht von der Erkennung beziehungsweise dem Benennen der versteckten Muster in den Daten. Muster beschreiben eine Teilmenge der Daten oder ein Modell, das sich auf diese Daten bezieht. Der Prozess der Wissensermittlung selbst birgt einen Informationsgewinn, indem versucht wird Informationen, die implizit und unerwartet in den Daten enthalten sind, zutage zu fördern. Somit hat dieser Prozess dass Ziel weiteres (verstecktes) Wissen auf Grundlage der gegebenen Daten zu gewinnen. Diese impliziten Informationen können zum einen dazu verwendet werden, um weitere Charakteristika des Datensatzes zu beschreiben. Zum anderen können sie als Grundlage für die Analyse neuer Datensätze dienen. Mögliche Aufgaben sind die Zusammenfassung großer Datenmengen oder die Möglichkeit Vorhersagen über neue Datenbestände zu treffen. Beispielsweise können Zusammenfassungen von Bestellungen, oder Vorhersagen über mögliche weitere Einkaufsartikel auf Grundlage der bereits im Warenkorb verfügbaren Artikel, erstellt werden.

[2] Hier erfolgt eine Festlegung auf „Knowledge Discovery in Databases (KDD)“. Der allgemeine Ansatz der Wissensermittlung ist hinsichtlich der Quelle seiner Eingangsdaten nicht festgelegt. Beispielsweise kann auch eine Datei die Menge der zu verarbeitenden Daten beinhalten.

Das Ziel des Emittlungsprozesses liegt in der Ermittlung neuer Informationen durch die Suche nach Regelmäßigkeiten und auftretenden Mustern innerhalb des Datensatzes.

Der KDD-Prozess wird von Fayyad et al. [74] als 9-stufiger Prozess deklariert. Der Prozess ist in nachfolgendem Abschnitt 2.2 detailliert vorgestellt. Aufbauend auf dieser grundlegenden Definition wurden zwei weitere bekannte Prozesse entwickelt [16]. Zum einen SEMMA (Sample, Explore, Modify, Model, Assess), welches vom SAS Institut[3] entwickelt wurde. SEMMA erlaubt die praktische Umsetzung (Organisation, Entwicklung und Wartung) vom Data-Mining Projekten. Dies wird unterstützt durch die hierfür entwickelte Software von SAS. Zum anderen erfolgt eine Unterstützung durch den CRISP-Data-Mining Prozess (CRoss-Industry Standard Prozess for Data Mining) [44]. Dieser Prozess wurde speziell auf die Bedürfnisse der Industrie angepasst. Es beinhaltet welches Projektziel erreicht werden soll, die Projektanforderungen die notwendig sind und im Anschluss daran die Planung der Umsetzung.[4]

2.2. Der KDD-Prozess

Die Definition von Fayyad et al. [74] beschreibt die einzelnen Stufen des KDD-Prozesses. Dieser Prozess wurde zunächst von Brachmann et al. [36] entworfen und später von Fayyad et al. angepasst. Heutzutage dient die Definition von Fayyad als Ausgangspunkt verschiedener Verfahrensvarianten, *z.B.* der nutzerspezifische Ansatz von Engels [67]. Daher wird in dieser Arbeit das Verfahren von Fayyad referenziert, dessen grundlegender Ablauf schematisch in Abbildung 2.1 dargestellt ist. Es beinhaltet die wesentlichen Schritte, die für die Generierung bzw. Extraktion neuen Wissens basierend auf einer gegebenen Datenquelle notwendig sind. Insbesondere soll der

[3] `http://www.sas.com/offices/europe/germany` [letzter Zugriff am 12.09.2011]

[4] Azevedo [16] beschreibt eine hohe Übereinstimmung zwischen SEMMA und dem KDD-Prozess. Er bezeichnet SEMMA als *„a practical implementation of the five stages of the KDD process“*, da es direkt in Zusammenhang mit der SAS Enterprise Miner software (`http://www.sas.com/technologies/analytics/datamining/miner` [letzter Zugriff am 12.09.2011]) steht. Der Fokus von CRISP bezieht sich auf das Verständnis eines Geschäftsmodells und bezieht sich somit auf eine speziellen Anwendungsfall. Daher bildet der beiden zugrundeliegenden KDD-Prozess den Bezugspunkt für die weitere Arbeit.

Prozess den willkürlichen Einsatz von Data-Mining Methoden verhindern und deren sinngemäße Einbettung bezüglich der benötigten Schritte bis hin zur Zielerreichung ermöglichen. Nachfolgend werden die einzelnen Schritte des Prozesses erläutert und auf den in dieser Arbeit vorgestellten Ansatz bezogen. Dies ermöglicht die Einordnung der später vorgestellten Teilverfahren in den Kontext des KDD-Prozesses. Die Orientierung erfolgt hierbei an der von Fayyad et al. vorgeschlagenen Sicht aus der Perspektive des Nutzers und anhand der neun Schritte des KDD-Prozesses, die in Fayyad et al. [74] aufgeführt wurden. Der Ausdruck *„Hier“* führt die vorgenommene Umsetzung des vorgestellten Prozessschrittes im Kontext dieser Arbeit aus.

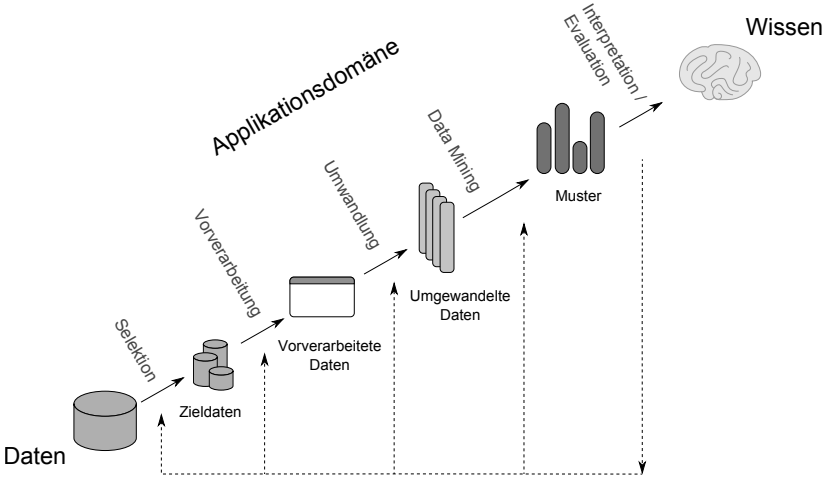


Abbildung 2.1.: Knowledge Discovery in Databases

1. **Anwendungsdomäne:** Dem Gesamtprozess vorausgehen muss der Aufbau eines initialen Verständnisses für das Einsatzgebiet, in dem das Verfahren angewandt werden soll. Der Aufbau eines initialen Verständnisses bedarf der Abschätzung des benötigten Vorwissens und der Definition des Ziels, das erreicht werden soll. Dieses initiale Verständnis ist die Grundlage für die nachfolgenden Prozessschritte. *Hier*: Ziel ist die Zuordnung der innerhalb eines Textes vorkommenden Bezeichner, *z.B.* „Helmut Schmidt", zu den diesen entsprechenden Ontologieelementen, *z.B.* im Rahmen von DBpedia `http://dbpedia.org/page/Helmut_Schmidt` [letzter Zugriff am 12.09.2011] für den deutschen Altkanzler. Jedoch sind die Bezeichner in den meisten Fällen nicht eindeutig und somit nicht direkt zuzuordnen. Beispielsweise sind in DBPedia[5] sieben weitere Personen mit dem Namen „Helmut Schmidt" enthalten, wodurch die Zuordnung anhand des natürlich-sprachlichen Bezeichners nicht eindeutig erfolgen kann. Deshalb muss ein Verständnis für Ontologien (siehe Kapitel 3), die das Hintergrundwissen darstellen, Bezeichnern in Texten (siehe Kapitel 4), die als Eingangsdaten der Analyse dienen, und die damit einhergehende Ambiguität (siehe Kapitel 5) erreicht werden.

2. **Selektion eines Zieldatensets:** Innerhalb dieses Prozessschrittes erfolgt die Selektion des Datensets, das als Grundlage der Analyse verwendet werden soll. *Hier*: Das Verfahren benötigt eine auf die textuelle Datengrundlage angepasste Domänenontologie. Die Ontologie fungiert zusätzlich zum Text als Datengrundlage des Verfahrens. Voraussetzung für eine mögliche Disambiguierung eines textuellen Bezeichners ist mindestens ein Element der Ontologie bzw. des Ontologiegeflechts[6], das diesem zugeordnet werden kann. Die in der Ontologie vorhandenen Informationen sind ausschlaggebend für den Erfolg des Verfahrens, *z.B.* im Zusammenhang mit dem zuvor gegebenen Beispiel die Information über die Personen, mit denen Helmut Schmidt in Kontakt steht, die Städte die er bereist hat *etc.*

[5] `http://dbpedia.org` [letzter Zugriff am 12.09.2011]

[6] Es können auch miteinander in Beziehung stehende Ontologien verwendet werden (siehe Kapitel 3).

3. **Datensäuberung und Vorverarbeitung:** In diesem Prozessschritt wird die vorhandene Datengrundlage aufbereitet, *z.B.* Umgang mit fehlender Information, fehlerhafte Daten *etc.* *Hier*: Im Kontext der Arbeit werden auf dieser Stufe Bezeichner von Ontologieelementen aus den gegebenen Texten extrahiert (siehe Abschnitt 4.2), die als Eingangsmenge des Algorithmus verwendet werden. Für eine spätere Algorithmusvariante sind Informationen, die im direkten Umfeld eines Bezeichners genannt werden, von besonderer Bedeutung (siehe Kapitel 10). Hier erfolgt die Extraktion kontextuell zusammenhängender Textbereiche, die später einen größeren Grad an Informationen zur Identifikation des zugehörigen Ontologieelements liefern sollen. Eine weitere Art der Vorbereitung umfasst die Speicherung früherer algorithmischer Resultate (aus vorausgehenden Disambiguierungen), die zur Verbesserung aktueller Analysen herangezogen werden (siehe Kapitel 11).

4. **Datenreduktion und -projektion:** Das Ziel des auszuführenden Algorithmus ist in diesem Schritt die Suche nach geeignetem Datenmaterial. Beispielsweise können im Falle einer Vektorrepräsentation Verfahren zur Dimensionalitätsreduktion ausgeführt werden. *Hier*: Die in Schritt 3 extrahierten Bezeichner werden ihren möglichen Ebenbildern in der Ontologie zugeordnet (siehe Kapitel 5 und Kapitel 8). Dadurch wird eine Projektion der textuellen Information auf die vorliegende Domänenontologie vorgenommen, *d.h.* auf alle Elemente, die diesen Bezeichner haben.

5. **Verwendung einer spezifischen Data-Mining Methode:** Die beabsichtigten Ziele, die durch den KDD-Prozess erreicht werden sollen, bedürfen einer spezifischen Data-Mining Methode. Diese Methode beschreibt die allgemeine Vorgehensweise und ist dem eigentlichen Verfahren, sowie seinen spezifischen Details übergeordnet. *Hier*: Die Einordnung des in dieser Arbeit präsentierten Verfahrens und dessen Abgrenzung hinsichtlich der verschiedenen Techniken des Data-Mining, werden in Abschnitt 2.3 erörtert.

6. **Explorative Analyse und Festlegung des zugrunde liegenden Modells und der Hypothese:**
   Dieser Schritt beinhaltet die Wahl des Data-Mining Algorithmus und die Methode zur Wissensermittlung. Im Zusammenhang mit dem Algorithmus erfolgt ebenfalls die Wahl des zugrunde liegenden Modells, *z.B.* Vektormodell. *Hier*: Das in dieser Arbeit entworfenen Verfahren basiert auf einem Graphmodel der RDF-Ontologie. Durch ein Data-Mining Verfahren soll der Bereich im Graph lokalisiert werden, *d.h.* der Teilgraph, der für jeden im Text identifizierten Bezeichner einer Entität mindestens ein Ontologieelement enthält und somit die vorhandene Ambiguität auflöst. Ein solcher Teilgraph zeigt die Beziehungen zwischen den im Text vorkommenden Entitäten gemäß der zugrundeliegenden Ontologiedefinition. Das Modell der Graphrepräsentation wird in Kapitel 7 präsentiert. Ein Beispiel hierzu ist in Abbildung 2.2 dargestellt. Im Text werden Bezeichner von Entitäten identifiziert. Danach erfolgt die Identifizierung der Ontologieelemente, die einen dieser Bezeichner besitzen. Innerhalb des Graphmodells wird nach Zusammenhängen zwischen den Elementen gesucht, die in Form von Teilgraphen repräsentiert werden. Die entworfenen Verfahren bauen auf der Technik des „Spreading Activation" auf, um den Teilgraph, der einem Steiner-Graphen entspricht, zu bestimmen.

7. **Data-Mining:** Hier findet die Ausführung des gewählten Data-Mining Verfahrens statt und die Erzeugung durch das Verfahren bestimmten Repräsentationsform der aufzufindenden Zusammenhänge. *Hier*: Die verschiedenen Verfahren werden in den Kapiteln 8, 9, 10 und 11 vorgestellt.

8. **Interpretation der gewonnener Zusammenhänge:** Die Interpretation der Zusammenhänge bedeutet eine Analyse der zuvor bestimmten Ergebnisstrukturen, *d.h.* die Bestimmung des optimalen Resultats. Des Weiterem erlaubt dieser Schritt eine Rückkehr zu früheren Prozessschritten. *Hier*: Die Resultatsbewertung wird anhand der Eigenschaften des Graphen im Bezug zum Data-Mining Verfahren durchgeführt, *z.B.* die Distanz zu den bestimmten Ontologieelementen ausgehend vom Knoten, der den Graph repräsentiert (siehe Kapitel 8). Die Rückkehr zu früheren Prozessschritten kann in zwei Varianten stattfinden. Bei der

kontextbasierten Analyse (siehe Kapitel 10) wird zunächst jeder Kontext separat analysiert. Hier wird versucht die direkt mit einem Bezeichner im Text aufgeführten Informationen zur Zuordnung zu nutzen. Bei der Verwendung von *Bestärkendem Lernen*[7] (siehe Kapitel 11) wird auf zuvor erzielte Resultate zurückgegriffen. Eine Auswertung dieser Resultate soll eine Verbesserung des Disambiguierungsverfahrens bewirken.

9. **Verwenden der erhaltenen Informationen:** Hier handelt es sich um die Einbindung des ermittelten Wissens in weiteren oder auf dem Verfahren aufbauenden Applikationen bzw. auch dessen Überprüfung hinsichtlich von Fehlern, *z.B.* Widersprüchen von zuvor erzielten Informationen. *Hier*: Die Zuweisung zum korrekten Ontologieelement ermöglicht den Zugriff auf die, mit diesem in Zusammenhang stehenden, Informationen. Diese Informationen können Zusammenhänge aufzeigen, die nicht explizit im Text genannt sind, jedoch zu dessen Verständnis einen hohen Beitrag leisten können. Beispielsweise können Ontologien Zusammenhänge zwischen im Text genannten Personen aufzeigen, die außerhalb der im Text dargestellten Beziehungen liegen und somit weitere Einblicke ermöglichen. Beispielsweise kann dies in anderen Applikation dem Benutzer als zusätzliche Information angezeigt werden. Auch weitergehende Prozesse, *z.B.* Clustering, können anhand der bestimmten ontologischen Merkmale durchgeführt werden.

[7] engl. Reinforcement Learning

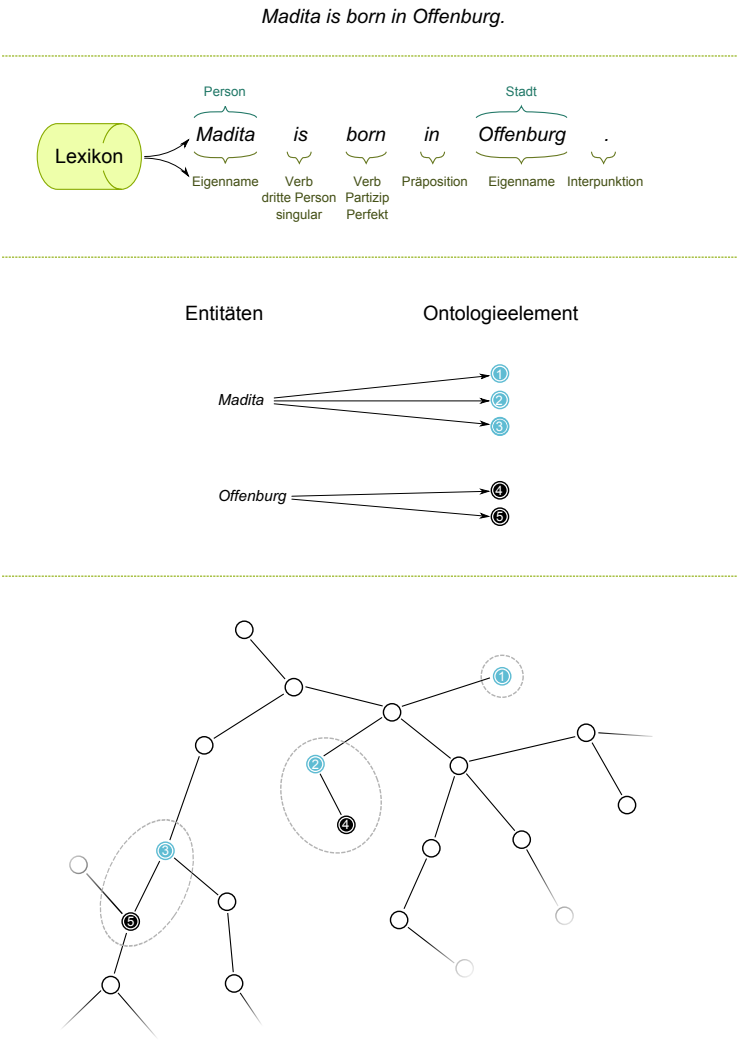


Abbildung 2.2.: Verfahrensüberblick

2.3. Data-Mining

Der Unterschied zwischen Data-Mining und KDD liegt darin, dass Data-Mining ein spezifisches Verfahren beschreibt, das Informationen basierend auf einer Datengrundlage gewinnt, während KDD den gesamten Prozess der Wissensermittlung beschreibt. Data-Mining ist somit ein Teil des KDD-Prozesses (vgl. [74]). In der Literatur werden die Aufgaben der Selektion, der Vorverarbeitung von Daten und der Transformation oftmals auch unter Data-Mining zusammengefasst [103]. Dieser Umstand wird auch aus der Prozessdarstellung in Abschnitt 2.2 ersichtlich. Basierend auf der Definition von Fayyad et al. beschreibt Data-Mining den Prozess zur Erkennung von Mustern und Relationen im Datensatz.

Folgende Data-Mining-Aufgaben werden von David Hand, Heikki Mannila und Padhraic Smyth [103] definiert:

Exploratory Data Analysis (EDA): Analyse der Daten ohne vorhergehende Zieldefinition. (Zufallsfunde).

Descriptive Modeling: Erstellung eines beschreibenden Datenmodells. Segmentierung der Daten und Clusteranalyse. Ebenfalls Erstellung einer kompakten Beschreibung (Zusammenfassung) der Daten.

Predictive Modeling: Klassifikation/Regression. Erstellung eines Modells, das Vorhersagen für den Wert eines/mehrerer Merkmale ermöglicht.

Discovering Pattern and Rules: Identifikation von Mustern und Strukturen. Erstellung von beschreibenden Regeln, welche die Beziehung zwischen Daten spezifizieren.

Retrieval by Content: Aufgrund einer gegebenen Menge an Merkmalen werden alle Daten aufgefunden, die eine Übereinstimmung hinsichtlich dieser Merkmalsmenge aufweisen.

Fayyad et al. unterscheiden zwischen den Zielen *Überprüfung* und zwischen *Ermittlung*, die durch Data-Mining gelöst werden können. Das in dieser Arbeit entwickelte Verfahren lässt sich dem zweiten Ziel zuordnen. Es ermöglicht nur bedingt Vorhersagen durch die Wiederverwendung von vorherigen Resultaten zur Tendenzbestimmung des

im aktuellen Kontext gemeinten Ontologieelements für einen gegebenen Bezeichner. Das Verfahren ermöglicht eine Beschreibung des relationalen Zusammenhangs zwischen Ontologieelementen basierend auf der Grundlage eines zuvor gegebenen Textes. Es wird für den Text ein Modell erzeugt, welcher aus der Kombination von Ontologiesubgraphen erstellt wird, jedoch auch künstliche, *d.h.* nicht durch das Ontologiemodell vorgegebene, Relationen enthalten kann (siehe Kapitel 10). Gemäß der von Hand et al. aufgestellten Kategorien hinsichtlich der Aufgaben des Data-Mining erfolgt eine Einordnung des in dieser Arbeit entwickelten Verfahrens in die Kategorie *Retrieval by Content*. Im Unterschied zur klassischen Suche liefert das vorgestellte Verfahren ein *beschreibendes Modell* der Zusammenhänge, der in dem gegebenen Text beschriebenen Ontologieelemente, *d.h.* der im Text beschriebene Zusammenhang zwischen den genannten Entitäten findet sich im Graphmodell wieder bzw. wird dort reflektiert. Dieser Zusammenhang bildet die Grundlage zur Disambiguierung mehrdeutiger Entitätsbezeichner (vgl. Abschnitt 6.4).

3. Semantic Web

Das World Wide Web stellt eine riesige Menge an Informationen zur Verfügung. Aktuell umfasst es eine Ansammlung von zirka 13.46 Billionen Internetseiten.[1] Das in diesen Seiten enthaltene Wissen birgt enormes Potential hinsichtlich der darin verborgenen Informationen. Leider existiert zumeist kein Modell, das die in einer Webseite beschriebenen Daten miteinander in Beziehung bringt. Falls Webseiten über ein solches verfügen, so sind die verwendeten Modelle zumeist unabhängig voneinander und strukturieren gleiche Inhalte in völlig unterschiedlicher Form. Die Verwendung eines gemeinsamen Formats würde ermöglichen, das im Internet vorhandene Wissen miteinander in Beziehung zu setzen. Dieses Kapitel erklärt den Zugriff auf diese Informationen unter Verwendung einer speziellen Datengrundlage – Ontologien. Eine Ontologie bezeichnet ein formales Modell zur Repräsentation von Wissen, das die oben genannten Vorteile bietet.

Zunächst wird in Abschnitt 3.1 der Begriff „Semantic Web“ erläutert und die Vision vorgestellt, die sich dahinter verbirgt. Die Wissensstruktur Ontologie, sowie die zwei am weitesten verbreiteten Ontologiesprachen, (RDF und OWL) werden im Abschnitt 3.2 beschrieben.

3.1. Vision

Beginnend von den ersten Entwürfen des World Wide Web von Tim Berners-Lee [134], bis hin zu einem weltumspannenden Netzwerk von 13.46 Billionen Internetseiten, hat das Web eine enorme Entwicklung erfahren. Heute ist es für viele Menschen eine bevorzugte Informationsquelle, *z.B.* für das Ermitteln von Öffnungszeiten, Kinoprogrammen, Adressen *etc.* Hierbei ist die einfache Erstellung von Webseiten, *d.h.* die Bereitstellung von Information, ein enormer Vorteil gegenüber

[1] `http://www.worldwidewebsize.com/` [letzter Zugriff am 12.09.2011]

anderen Medien. Der Einfluss dieses Kommunikationsmediums zeigt sich dadurch, dass allein in der EU 2009 bereits 65% der Bevölkerung Zugriff auf das Internet hatten [70].

Ein Vorteil und gleichzeitig ein Nachteil ist die Ausrichtung der auf den Internetseiten vorgestellten Information auf die Menschen als Konsumenten. Dies bedeutet, dass Internetseiten darauf ausgerichtet sind, Menschen einen einfachen Zugriff auf die dort enthaltenen Informationen zu ermöglichen. Der angesprochene Nachteil liegt darin, dass eine computergestützte Analyse über kaum direkt verwertbare Informationen verfügt (*z.B.* Links), sondern die menschliche Sprache analysieren muss, um den auf der Webseite dargestellten Sachverhalt zu erfassen. Es wurden in der Wissenschaft zwar Ansätze, *z.B.* Methoden zur Interpretation der menschlichen Sprache entwickelt *etc.* (siehe [149]), jedoch ermöglichen diese keine exakte Erfassung der Informationen, sondern lediglich eine partielle Deutung der durch die Webseite kommunizierten Botschaften. Der Mensch ordnet diese in einen konzeptuellen Zusammenhang ein, den die Webseite selbst nicht als Datenstruktur enthält. Auf diesen Missstand wurde bereits durch Tim Berners Lee et al. 2001 hingewiesen [22]. In der Konsequenz stellte er die Lösung durch das „Semantic Web“ in Aussicht. Die Autoren definierten das Semantische Web als:

The Semantic Web is an extension of the current web in which information is given a well-defined meaning, better enabling computers and people to work in cooperation.

Hierdurch wird deutlich, dass das Semantic Web eine Weiterentwicklung des Internets ist, jedoch eine wohldefinierte Semantik besitzt, die vornehmlich auch Computern den Zugriff auf darin zur Verfügung gestellte Informationen ermöglicht. Feigenbaum bezeichnete das Semantic Web als *„[...] an enhancement that gives the Web far greater utility“* [75]. Der gesteigerte Nutzwert geht darauf zurück, dass immer mehr Menschen konzeptuelle Modelle[2] über Gebiete erzeugen, in denen sie Wissen gesammelt haben. Diese Gebiete sind vollkommen unterschiedlich, *z.B.* Musik, Orte, Bücher *etc.* Dieses formalisierte Wissen kann maschinell ausgelesen werden und durch

[2] Neben der rein konzeptuellen Modellierung wird auch davon ausgegangen, dass die Existenz von Individuen, die diesen Konzepten zugeordnet sind, mit eingeschlossen ist.

seine klare Struktur Computern sowie Menschen helfen, Nutzen aus den Daten zu ziehen. Das WWW Consortium (W3C), das sich der Entwicklung des Semantic Web angenommen hat, bestätigt dies durch die Definition des Semantic Web als „*The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries*“ [112]. Zugleich wird die oben angesprochene Fehlerquelle, die bisherige Wissensermittlung anhand der vagen Interpretation natürlicher Sprache, entschärft.

Die angesprochenen konzeptuellen Modelle bedürfen jedoch einer standardisierten Form, um Verbreitung zu finden und die Interpretation sowie Kombination dieser Modelle zu ermöglichen. Im Semantic Web werden solche konzeptuellen Modelle durch Ontologien beschrieben.

3.2. Ontologien

Der Begriff „Ontologie“ wird in zwei wissenschaftlichen Bereichen verwendet - in der Philosophie und im Bereich der künstlichen Intelligenz (KI). Der Begriff Ontologie in der KI ist abgeleitet von dessen Bedeutung in der Philosophie. Der Begriff wurde ursprünglich von den Studenten von Aristoteles eingeführt. Diese verwendeten ihn als Synonym zur Beschreibung des Aristoteles „first philosophy“ in Metaphysics, IV, 1 [193] - der Studie der Dinge, die möglicherweise existieren (vgl. Smith in [83]). Die erste Aufzeichnung des Wortes im Englischen erfolgte 1721 im Oxford English Dictionary, welches die Bedeutung des Worts mit *„an Account of being in the Abstract“* beschrieb. Dies lässt sich als die „Natur des Seins“ interpretieren. Wichtig sind die Fragestellungen, die hiermit verbunden sind. So beschäftigt sich dieser Zweig der Philosophie mit Fragen der Art: „Welche Klassen von Entitäten sind für eine vollständige Beschreibung und Erklärung der Vorgänge in dieser Welt nötig?“ [83] (siehe auch [98]). Somit steht die Beschreibung der Zusammenhänge im Vordergrund. Diese Art der Fragen gibt zudem einen Hinweis auf die Analogie zu der Verwendung des Begriffes in der Informatik.

Im Kontext der Informatik bezeichnet der Begriff Ontologie eine Struktur zur Wissensrepräsentation. Die philosophische Fragestellung nach der „Natur des Seins“ wird gemäß Gruber [96, 97] interpretiert

als *„[...] what 'exists' is that which can be represented"*. Der Fokus der Fragestellung, die in der Philosophie auf eine grundsätzliche und allgemein gültige Antwort ausgerichtet ist, verschiebt sich hin zu einer spezifischen Sicht auf ein KI System. Es gibt verschiedene Definitionen für den Begriff der Ontologie (vgl. [99]), die bekannteste jedoch wurde von Thomas Gruber [97] formuliert: *„An ontology*[3] *is an explicit specification of a conceptualization."*

Unter Konzeptualisierung[4] versteht Gruber eine vereinfachte und abstrakte Sicht auf den Teil der Welt, der durch die Ontologie beschrieben werden soll, insbesondere die vorkommenden Konzepte und Objekte sowie die Relationen zwischen diesen. Die Konzeptualisierung ist spezifisch für das jeweilig gegebene Interessensgebiet auszusuchen, *d.h.* den thematischen Bereich, der durch das in der Ontologie repräsentierte Wissen beschrieben werden soll. Die abstrakte Sicht drückt aus, dass Ontologien nicht auf die bloße Existenz von bestimmten Individuen[5] fixiert sind, sondern auf eine Beschreibung des allgemeingültigen Zusammenhangs [98] sowie des Zusammenhangs zwischen den Individuen. Die Konzepte und deren Zusammenhänge beschreiben hierbei das Vokabular zur Wissensdarstellung. Gruber bezeichnet eine Ontologie weiterhin als eine „explizite Spezifikation". Darunter versteht er, dass die vorkommenden Konzepte und deren Beschränkungen, *z.B.* Gleichheit von Konzepten, eindeutig und somit unmissverständlich definiert sind. Aufbauend auf der Definition von Gruber definierte Borst eine Ontologie als eine *„formal specification of a shared conceptualisation"* [34]. „Formal" bezieht sich auf die Tatsache, dass die in der Ontologie enthaltenen Beschreibungen maschinell lesbar sein müssen. Die hierfür verwendete Sprache, *z.B.* Frame-Logic (F-Logik) [122], muss eine eindeutige Interpretation der Ontologie ermöglichen, *d.h.* mehrdeutige Möglichkeiten der Interpretation sind ausgeschlossen. Der Unterschied zwischen der philosophischen Betrachtung der in der KI liegt und in der notwendigen Formalisierung

---

3 Im Rahmen dieser Dissertation unter mit dem Begriff *Ontologie* nicht nur das konzeptuelle Modell sondern auch die zugehörigen Instanzen sowie deren Relationen verstanden.

4 Zu Beginn wurde eine Ontologie primär als konzeptuelle Domänendefinition betrachtet. Zwischenzeitlich jedoch werden auch die den Konzepten zugewiesen Individuen mit einbezogen. Das in dieser Arbeit vorgestellte Verfahren ist hauptsächlich auf Individuen ausgerichtet.

5 Die Begriffe „Individuen" und „Instanzen" werden synonym in dieser Arbeit verwendet.

des Wissens. Woods beschreibt den Unterschied zwischen beiden Gebieten durch die Aussage *„Philosophers have generally stopped short of trying to actually specify the truth conditions of the basic atomic propositions, dealing mainly with the specification of the meaning of complex expressions in terms of the meaning of elementary ones. Researchers in artificial intelligence are faced with the need to specify the semantics of elementary propositions as well as complex ones.“* [241]. „Shared“ bezieht sich auf die Definition der Ontologie. Die Definition soll die gemeinsame Sicht einer Gruppe von Menschen darstellen und nicht auf der Interpretation der Verhältnisse in einem System durch eine einzelne Person beruhen. Die „geteilte“ Sicht ermöglicht es erst ein Gebiet adäquat zu beschreiben. Dies ist darin begründet, dass der Einzelne zumeist nur wenige, subjektiv festgelegte Aspekte einer Domäne beschreibt. Der Begriff „shared conceptualisation“ geht jedoch weiter, indem er ein für alle zu verwendendes Vokabular, *d.h.* die Begrifflichkeiten der Ontologie, für diese Domäne beschreibt. Erst das Fundament einer gemeinsamen Begriffswelt ermöglicht den Austausch von Wissen (vgl. [59]). In dieser Arbeit wird auf der erweiterten Definition einer Ontologie durch Studer et al. [222] bzw. deren Erweiterung um „of a domain of interest“ (vgl. [223]) aufgebaut:

Ontologiedefinition. *„An ontology is a formal, explicit specification of shared conceptuatlization of a domain of interest“*

Diese Definition basiert auf einer Kombination der zuvor genannten Ontologiedefinitionen. Sie beschreibt, dass die von Gruber erwähnte „explizite“ Spezifikation in einer formalen Sprache erfolgt und somit deren Interpretation in einer wohl-definierten Art und Weise, *d.h.* deren maschinelle Lesbarkeit, garantiert. Die festgelegte Beschränkung auf das Interessensgebiet birgt zwei wesentliche Vorteile. Erstens ermöglicht sie die Fokussierung auf die Gegebenheiten dieser Domäne, *d.h.* es wird eine detailreiche Darstellung der in der Domäne vorherrschenden Verhältnisse gefördert. Dies bedeutet jedoch nicht, dass eine vollständige Darstellung der Domäne erreicht werden muss (vgl. [218]). Zweitens ermöglicht sie die modulare Darstellung von Teilbereichen. Diese Module können miteinander in Beziehung gesetzt, *d.h.* kombiniert werden, und damit die Sicht auf eine Domäne oder den Zusammenhang zwischen Domänen darstellen.

Zusammenfassend betrachtet ermöglicht eine Ontologie eine klare strukturelle Darstellung von Wissen.[6]

Dieses Wissen bezieht sich auf die dargestellte Domäne und bildet den Kern bzw. das „Herzstück" der Wissenrepräsentation dieser Domäne [43]. Gleichzeitig ermöglicht die formale Definition bzw. das Vokabular der Ontologie die maschinelle Lesbarkeit dieser Wissensbasis und somit den Zugriff durch KI-Systeme. Durch Ontologien wird es möglich neue intelligente Informationssysteme zu bauen. Diese finden Anwendung in den Bereichen des Wissensmanagement (*z.B.* [258]), Information Retrieval, Sprachverarbeitung und Informationsintegration [77]. Je nach Anwendung werden unterschiedliche Anforderungen an Ontologien gestellt. Guarino [100] teilt Ontologien in vier verschiedene Klassen ein:

Top-Level Ontologien: In dieser Art von Ontologie werden Konzepte modelliert, die unabhängig von einer Domäne oder einer konkreten Problemstellung sind. Bekannte Top-Level Ontologien sind die Suggested Upper Merged Ontology (SUMO)[7] und die Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE)[8]. SUMO unterscheidet auf der ersten Ebene „Physical und Abstract Entity" während DOLCE hauptsächlich zwischen „Endurants" (Substanzen und Objekten) sowie „Perdurants" (Ereignissen und Prozessen) differenziert. Oberle et al. [166] beschreibt DOLCE als erste Wahl bezüglich der Aufgabe von Referenzierungen, während die in SUMO gegebene Taxonomie sich gut für Klassifizierungsaufgaben eignet. Weitere Top-Level Ontologien sind die Dublin Core Ontology[9], eine Ontologie zum Dublin Core Standard, und OpenCyc[10], eine

6 Zu Beginn war die Beschreibung der Struktur, *d.h.* der konzeptuellen Zusammenhänge, das primäre Ziel einer Ontologiedefinition. Dies hat sich jedoch im Laufe der Zeit gewandelt, so dass heutzutage insbesondere die Beschreibung und das Vorhandensein von Instanzen wesentlich für viele Einsatzzwecke von Ontologien ist. Zum Beispiel besteht DBPedia (`http://wiki.dbpedia.org/` [letzter Zugriff am 12.09.2011]), eine Ontologie, die auf einem Gemeinschaftsprojekt zur Extraktion von strukturierten Daten aus Wikipedia (`http://www.wikipedia.org/` [letzter Zugriff am 12.09.2011]) basiert, primär aus Instanzwissen und verfügt im Verhältnis dazu nur über einen geringen Anteil konzeptueller Information.

7 `http://ontology.teknowledge.com` [letzter Zugriff am 12.11.2011]

8 `http://www.loa-cnr.it/DOLCE.html` [letzter Zugriff am 12.09.2011]

9 `http://www.cs.umd.edu/projects/plus/SHOE/onts/dublin.html` [letzter Zugriff am 12.09.2011]

10 `http://www.opencyc.org` [letzter Zugriff am 12.09.2011]

Ontologie, die eine allgemeine, *d.h.* nicht domänenbeschränkte, Wissensbasis beschreibt.

Domänen Ontologien: Diese beschreiben eine generische Domäne. Diese sind einem abgegrenzten Interessengebiet zugeordnet, *z.B.* „Bücher“ oder „Informatik“. Ein Beispiel ist die Geonames-Ontologie[11], welche die Geographie-Domäne darstellt oder die SwetoDblp-Ontologie[12], die wissenschaftliche Veröffentlichungen beinhaltet.

Aufgaben-Ontologien: Diese widmen sich der Beschreibung einer bestimmten Tätigkeit, *d.h.* einer generischen Aufgabe, *z.B.* „Autofahren“ oder „Verreisen“. Auch Arbeitsabläufe können mit diesem Typ von Ontologie detailliert beschrieben werden.

Anwendungs-Ontologien: Diese Art von Ontologien bezeichnet eine Mischung zwischen Domänen- und Aufgaben-Ontologie. Die Konzepte in diesen Ontologien werden oft an die Tätigkeiten innerhalb der Domäne angepasst, *z.B.* „technische Fette“ oder „Schmierstoffe“ im Bereich einer Autowerkstatt. Ein weiteres Beispiel ist die Experimental Factor Ontology (EFO)[13], diese wird als *„an application focused ontology modelling the experimental factors in ArrayExpress*[14]*“*. Die EFO gibt die Faktoren der Experimente (Aufgabe) an, die in ArrayExpress als genomische Datenbank gespeichert ist[15] werden.

11 `http://www.geonames.org/ontology/` [letzter Zugriff am 12.09.2011]

12 `http://knoesis.wright.edu/library/ontologies/swetodblp` [letzter Zugriff am 12.09.2011]

13 `http://www.ebi.ac.uk/efo` [letzter Zugriff am 12.09.2011]

14 ArrayExpress beschrieben, eine Datenbank für Genomik Experimente einschließlich Genexpression (siehe `http://www.ebi.ac.uk/arrayexpress` [letzter Zugriff am 12.09.2011]).

15 Für die Speicherung von Ontologien gibt es verschiedene Systeme, *z.B.* [29] oder das vom Autor, Jörg Henss und Stephan Grimm entwickelte Datenbanksystem Mnemosyne, das speziell für OWL-Ontologien entwickelt wurde [254, 253].

3.2.1. Ontologiesprachen

Wie zuvor dargestellt, ermöglichen Ontologien eine strukturelle Darstellung von Wissen, die für die Ausführung vieler KI-Systeme eine Grundvoraussetzung ist. Ihre Aufgabe ist es, die in einem System vorkommenden Objekte sowie die Zusammenhänge zwischen diesen zu charakterisieren. Zur Beschreibung von Ontologien wurden verschiedene Sprachen entwickelt. Im Folgenden werden die zwei verbreitetsten, das Resource Description Framework (RDF) (mit der Erweiterung um RDF-Schema) und die Web Ontology Language (OWL), vorgestellt.

RDF(S)

RDF als Beschreibungssprache für Ontologien wird von Tim Berners-Lee in seinem Artikel über das Semantic Web als Sprache zur Beschreibung von Ontologien eingeführt - *„Meaning is expressed by RDF“* [22]. Das W3C beschreibt den Zusammenhang als *„The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries [...]. It is based on the Resource Description Framework (RDF)“* [112]. Dies bezeichnet RDF als die Grundsprache[16] zur Beschreibung von semantischen Zusammenhängen. Der in dieser Arbeit vorgestellte Ansatz baut ebenfalls auf Ontologien dieses Formats auf.

Die Entwicklung von RDF begann 1999 und RDF wurde 2004 als W3C Standard [128] spezifiziert. Die Sprache basiert auf der Extensible Markup Language (XML) [37] und wurde entworfen, um den Austausch von Daten im Web unter Beibehaltung der ursprünglichen Bedeutung durchführen zu können. RDF ermöglicht die Beschreibung von Daten durch Metadaten, *d.h.* die Bereitstellung von Informationen über Daten. Dies wird auch häufig als die Annotation[17] von Daten bezeichnet.

[16] Der Duden beschreibt eine Grundsprache als tatsächlich bezeugte oder auch nur erschlossene Sprache, aus der mehrere verwandte Sprachen hervorgegangen sind, zu denen sie die gemeinsame Vorstufe darstellt (vgl. `http://www.duden.de/rechtschreibung/Grundsprache` [letzter Zugriff am 12.09.2011]).

[17] Eine Annotation ist eine zusätzliche Beschreibung der Daten, die jedoch keine direkten Auswirkungen auf die Daten selbst hat, *d.h.* die Daten selbst bleiben unverändert.

Beispiele für solche Annotationen sind das Hinzufügen von Informationen, wie Autor, Regisseur oder Erstellungsdatum. Das ist insbesondere vorteilhaft bei Ressourcen, wie *z.B.* Audio- oder Videodaten, die eine solche Möglichkeit nicht explizit zur Verfügung stellen.[18]

Jedes Objekt ist in RDF durch einen Uniform Resource Identifier (URI) eindeutig identifizierbar. Die Verwendung dieses Formats lässt somit keine Mehrdeutigkeit zu. Der beschriebene Zusammenhang kann als Graph repräsentiert werden.

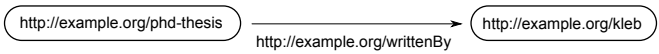


Abbildung 3.1.: RDF-Tripel

In der Definition von RDF wird der Graph in Tripel (siehe Abbildung 3.1) zerlegt, welche einen konkreten Zusammenhang zwischen zwei Objekten definiert.[19]

Um Aussagen über eine Ressource zu treffen, werden drei grundlegende Elemente benötigt:

1. *Typen von Individuen*, *z.B.* Regisseur, Schauspieler, Autor *etc.*
2. *Beziehungen* zwischen diesen, *z.B.* Schauspieler → arbeitet für → Regisseur
3. *Datentypen* von Individuen oder auch anderen Ressourcen, wie *z.B.* Zahlen oder Strings[20]

Damit wird es möglich, die semantischen Beziehungen zwischen Individuen einer Domäne zu definieren. RDF selbst bietet jedoch nur die Beschreibung von Ressourcen und ein sehr eingeschränktes Vokabular, *z.B.* ist die Beschreibung von Konzepthierarchien und Datentypen nicht möglich. Hierfür wurde RDFS [38] entwickelt. RDFS stellt ein konkretes Vokabular für RDF dar, welches ermöglicht ein konzeptuelles Modell zu definieren, *z.B.* Person → arbeitet für → Person.

RDFS erfüllt somit alle Anforderungen, die von der im vorhergehenden Abschnitt vorgestellten Ontologiedefinition verlangt werden. Es

[18] abhängig vom verwendeten Datenformat

[19] Die Graphrepräsentation von RDF wird in Abschnitt 7.4 beschrieben.

[20] Bei einem String handelt es sich um eine Folge von Zeichen, Buchstaben oder Wörtern. Die sind in einer Zeichenkette mit variabler Länge zusammengefasst.

ermöglicht die Beschreibung der semantischen Zusammenhänge in einem Gebiet. RDFS ist maschinell lesbar und verfügt über eine formale Semantik. Die Definition von Klassen, die Zuordnung von Klassen und Relationen und die Bestimmung der Quelle und des Ziels einer Relation werden durch RDFS ermöglicht.

OWL

In RDF(S) ist es beispielsweise nicht möglich den Wert von Eigenschaften zu beschränken, Disjunktionen auszudrücken, sowie Mengen oder Kardinalitäten exakt anzugeben. Es gibt es Fälle, in denen die Ausdrucksfähigkeit von RDF(S) nicht hinreichend ist, um den Detailgrad auszudrücken, der vom Autor einer Ontologie zur Beschreibung gewünscht wird.

OWL ist im Gegensatz zu RDF eine sehr ausdrucksmächtige Sprache, die auf Beschreibungslogik basiert.[21] Die Sprache wurde zunächst von der Web Ontology Working Group des W3C 2004 [243] vorgeschlagen und 2009 mit OWL2 überarbeitet [244]. OWL selbst baut auf früheren Beschreibungssprachen, wie DAML+OIL[22] auf. Formale Logik ermöglicht das automatische Schlussfolgern (Reasoning), welches erlaubt, implizite Informationen explizit zu machen. Zudem kann mittels automatischen Schlussfolgerns überprüft werden, ob die beschriebenen Definitionen gegenseitig miteinander vereinbar, *d.h.* konsistent, sind. Beim Entwurf von OWL stand die Abwägung zwischen der Ausdrucksmächtigkeit der Sprache und deren Einfluss auf die Skalierbarkeit des Reasoning im Vordergrund. Bei OWL2 wurde die Ausdrucksmächtigkeit gegenüber OWL1 erhöht (siehe [244]). OWL-Ontologien können ebenfalls als Graphen repräsentiert werden. Aufgrund der komplexen Logik sind diese Graphen aber nicht zu vergleichen mit den zuvor genannten RDF(S)-Graphen, da die Semantik beider Sprachen unterschiedlich ist.

Für OWL existieren drei Sprachen unterschiedlicher Ausdrucksmächtigkeit, um den Ansprüchen verschiedener Anwendergruppen zu genügen.[23]

---

21 Beispielsweise ist es möglich Symmetrie, Transivität, Funktionalität und Inverse zu beschreiben.

22 `http://www.daml.org` [letzter Zugriff am 12.09.2011]

23 vgl. `http://www.w3.org/TR/owl-features/` [letzter Zugriff am 12.09.2011].

- OWL-Lite - stellt nur eine geringe Ausdrucksmächtigkeit zur Verfügung. Es können einfache Beschränkungen zu Klassen hinzugefügt werden. Es eignet sich gut zur Erstellung von Taxonomien und Thesauri.
- OWL-DL - beschreibt den Kompromiss zwischen Ausdrucksmächtigkeit und Berechenbarkeit, *d.h.* alle enthaltenen logischen Konstrukte sind berechenbar. Bei der Ausdrucksmächtigkeit gibt es daher Einschränkungen gegenüber OWL-Full hinsichtlich der Beschreibung von Klassen. OWL DL genügt den Anforderungen der Description Logic und baut auf einer entscheidbaren terminologischen Logik auf.Es stützt sich auf eine Teilmenge der Prädikatenlogik erster Stufe. Für das Reasoning in OWL-DL Ontologien gibt es unterschiedliche algorithmische Ansätze (siehe auch [33]).
- OWL-Full - stellt die größte Ausdrucksmächtigkeit zur Verfügung. Problematisch an OWL-Full ist, dass nicht alle Aussagen überprüft werden können. So ist die Feststellung der Konsistenz der Ontologie nicht in allen Fällen möglich (siehe auch [204]).

Ob RDF(S) oder OWL zur Beschreibung einer Ontologie verwendet wird, hängt vom benötigten Detailgrad der zu beschreibenden semantischen Zusammenhänge ab. OWL ermöglicht eine große Ausdrucksmächtigkeit sowie das automatische Schlussfolgern von impliziten Zusammenhängen.[24] Dies ist jedoch verbunden mit einem größeren Aufwand für die maschinelle Verarbeitung. RDF(S) gestattet die Definition weniger komplexer Ontologien und ermöglicht einen direkten und schnellen Zugriff auf die beschriebene Information. Das in dieser Arbeit vorgestellte Verfahren bezieht sich auf Ontologien im RDF(S)-Format.[25] Dies begründet sich darin, dass die meisten Ontologien heutzutage immer noch im RDF(S)-Format vorliegen.

[24] Existieren beispielsweise mehrere RDF-Dokumente zu einer Person, die für eine Data-Property unterschiedliche Werte verwenden, *z.B.* unterschiedliche Angaben für den Wohnort, so kann mittels OWL die Inferenz gezogen werden, dass es sich um den gleichen Ort handeln muss.

[25] Alle auf die in dieser Arbeit vorgestellte Verfahren benötigten Informationen können ebenfalls aus einer OWL-Ontologie extrahiert werden.

4. Entitäten

Im bisherigen Verlauf der Arbeit wurde von der Zuordnung eines Ontologieelements bzw. im Fall von Ambiguität mehrerer Ontologieelemente zu einem im Text aufgefundenen Bezeichner gesprochen. Der Begriff „Bezeichner“ ist für alle möglichen textuellen Beschreibungen gültig. Das in dieser Arbeit vorgestellte Verfahren fokussiert sich jedoch auf die textuelle Beschreibung von Entitäten, *d.h.* Entitätsbezeichner. Folglich steht die Zuweisung des richtigen Ontologieelements zu einem Entitätsbezeichner, *d.h.* dessen Referenz in der Ontologie, im Fokus dieser Arbeit.[1]

In Abschnitt 4.1 wird zunächst auf die allgemeine Definition von Entität sowie die Zuordnung eines textuellen Bezeichners zu einer Entität (Benannte Entität) eingegangen. In diesem Zusammenhang von besonderem Interesse sind die Gebiete der Sprachanalyse sowie des Wissensmanagements durch Ontologien. Auf beiden Gebieten baut diese Arbeit auf. Die Einbindung von Entitäten in beiden Bereichen wird in den Abschnitten 4.2 und 4.3 vorgestellt. Eine Gegenüberstellung beider Bereiche erfolgt in Abschnitt 4.4. Abschnitt 4.5 behandelt den Bezug zu weiteren Gebieten der Informatik.

4.1. Entität und Benannte Entität

Das Wort „Entität“ ist vom lateinischen Begriff „ens“ abgeleitet, der als „seiend“ bzw. „Ding“ interpretiert werden kann. Seinen Ursprung hat der Begriff in der Philosophie. Die nähere Bedeutung des Begriffs

[1] Der vorgestellte Ansatz kann prinzipiell auch für Bezeichner verwendet werden, die nicht in Zusammenhang mit einer Entität stehen. Daher erfolgte zunächst keine Einschränkung des Terminus „Bezeichner“.

beschreibt Roberto Poli [178] mit dem Rückgriff auf die Definition der „Stoiker“[2], die Entität auf drei verschiedene Arten definierten:

1. als Individuum (*soma*), *d.h.* als existierender Körper
2. etwas Körperloses (*on*)
3. als etwas Unbestimmtes (*ti*).

Poli führt aus, dass *soma* über ein identifizierendes Merkmal verfügt und ebenfalls *on* in den meisten Fällen ein solches besitzt im Gegensatz zu *ti*.

Die obige Definition trifft im Allgemeinen Sprachgebrauch ebenfalls zu. So definiert das Oxford Dictionary[3] eine Entität als *„a thing with distinct and independent existence“*. Man kann eine Entität als etwas bezeichnen, das unterscheidbar und unabhängig von anderen Dingen existiert. Eine Einbildung oder Abstraktion kann daher als Entität bezeichnet werden. Ein körperloses Individuum wäre beispielsweise die Sprache, die wir Menschen verwenden, während man Sie als Leser dieser Arbeit als ein lebendes Individuum bezeichnen kann und somit ebenfalls als eine Entität. Bei körperlosen Entitäten handelt es sich beispielsweise auch um unbestimmte Bezeichnungen. Entitäten sind somit materiell oder immateriell. Oftmals wird der Begriff (siehe Definition zuvor) auch für Unbestimmtes verwendet, so spricht beispielsweise der Präsident von Aserbaidschan auf seinem Webauftritt von *„The ethnos within the nation-State cannot be the political-legal entity“*[4]. Entität wird hierbei interpretiert als Existenz selbst und in diesem Fall somit als Existenz eines solchen Umstandes.Entity ist in der Politikwissenschaft und bei Politikern eine Ausweich-Notabel. Mit *„Politischer Entity“* werden Länder bezeichnet, die man nicht als Staat bezeichnen möchte, *z.B.* Palästina, Taiwan, Mazedonien *etc.*).

[2] Die „Stoiker“ waren eine philosophische Bewegung während der Hellenistischen Zeit. Die „Stoiker“ betrachten Gefühle wie Angst oder Eifersucht als Fehlurteil bzw. dessen Folge (siehe auch Stanford Encyclopedia of Philosophy (`http://plato.stanford.edu/entries/stoicism/` [letzter Zugriff am 12.09.2011])).

[3] `http://oxforddictionaries.com/` [letzter Zugriff am 12.09.2011]

[4] `http://www.president.az/pages/9/print?locale=en` [letzter Zugriff am 12.09.2011]

Der Begriff „Benannte Entität“ (Named Entity[5]) unterscheidet sich durch das Hinzufügen eines Namens. Silvio Brendler formulierte folgende Definition: *„Unter ‚Name‘ (wie auch ‚Eigenname‘) wird ein Substantiv verstanden, das ein als Individuum betrachtetes Objekt bezeichnet“* [20]. Diese grundlegende Definition beschreibt eine Benannte Entität als Entität, die durch eine Assoziation mit einem identifizierenden Merkmal erweitert wird. Das reduziert die oben dargestellten Arten von Entitäten zu *sumo* und *on*, da *ti* kein identifizierbares Merkmal besitzt.[6] Eine Benannte Entität kann über mehrere Namen identifiziert werden. Zudem ist die Deklaration als Substantiv eine nicht zu unterschätzende Charakteristik, *z.B.* werden im Deutschen Substantive ausnahmslos mit einem Großbuchstaben begonnen.

Satoshi Sekine [212] definierte eine Benannte Entität durch die Aussage: *„In the expression ‚Named Entity‘, the word ‚Named‘[...]“* refers *„to those entities for which one or many rigid designators [...] stand for the referent“*. Er folgt der Definition von Saul Kripke hinsichtlich des Begriffs „rigid designator“. Kripke [130] beschreibt einen rigiden Bezeichner als einen Term, der erstens das gleiche Objekt in allen Welten aussucht, in denen dieses Objekt existiert. Des Weiteren darf er kein anderes Objekt in allen Welten auswählen, in denen das Objekt nicht existiert. Drittens sind rigide Bezeichner natürlich-sprachliche Eigennamen (vgl. [46]). Ein Beispiel für einen rigiden Bezeichner ist „Rudi Studer“ während der Ausdruck „Leiter der Forschungsgruppe Wissensmanagement am AIFB[7]“ eine *nicht* rigide Bezeichnung darstellt. Letztere ist nicht rigide, da in einer anderen Welt eine andere Person Leiter dieser Gruppe sein könnte. Daniel Jurafsky und James H. Martin [117] generalisierten die von Sekine aufgestellte Definition durch die Aussage: *„By named entity, we simply mean anything that can be referred to with a proper name“*. Sie teilen somit die Auffassung von Mikheev[8] [152] und Evans[9] [72], welche die Definition des

5 Es gibt keine offiziell gültige Übersetzung des Begriffs. Im Folgenden wird der deutsche Begriff „Benannte Entität“ verwendet, welche die Zuweisung eines Bezeichners ausgedrückt und damit auf das wesentliche Unterscheidungsmerkmal hingewiesen wird.

6 Beispielsweise ist die Entität, die sich hinter dem Begriff „Existenz“ verbirgt, vom Typ *ti*. Diese Entität kann nicht als Individuum betrachtet werden.

7 Institut für Angewandte Informatik und Formale Beschreibungsverfahren (AIFB) am Karlsruher Institut für Technologie

8 *„What counts as a Named Entity depends on the application that makes use of the annotations“*

9 *„The set of required name types varies from case to case “*

„proper name“ als applikationsabhängig betrachten. Der Zusammenhang mit Kripke ist dadurch gegeben, dass er proper names ebenfalls als rigide Bezeichner beschreibt [130]. Im Zusammenhang mit Ontologien beschreibt Guarino [98] den Ausdruck als eine Eigenschaft, die in allen Welten gilt. Beispielsweise ist eine Zuordnung zum Konzept „Person“ rigide, da diese sich für eine Instanz nicht ändert. Eine Zuordnung zum Konzept „Student“ ist jedoch nicht rigide, da diese nur eine zeitlich begrenzte Gültigkeit aufweist.

Ein Eigenname allein ist allerdings nur eine hinreichende Bedingung zur Identifikation eines Objektes. Er ermöglicht zwar die angesprochene Identifikation einer Entität, garantiert jedoch keine eineindeutige Bezeichnung bzw. eindeutige Identifikation. Das in Kapitel 1 erwähnte Beispiel des Entitätsbezeichners „Mölln“ ist hier ebenfalls passend. Dieser identifiziert sowohl die Stadt Mölln in Stavenhagen als auch die Stadt Mölln in Schleswig-Holstein. Kripke beschreibt keine klaren Verfahrensweisen zur Berücksichtigung von mehrdeutigen[10] Eigennamen. Lycan betonte die Bindung des rigiden Bezeichners an dem Blickwinkel des Betrachters, *„Kripke argues that when one uses the name 'Nixon' to refer a person in this world and then starts describing hypothetical scenarios or alternative possible worlds, continuing to use the name, one is talking about the same person.“* [142]. Zur Vertiefung dieser Diskussion aus Sicht der Philosophie wird auf weiterführende Literatur verwiesen, *z.B.* [240, 141, 142].

Im Rahmen dieser Arbeit wird auf die Definition von Sekine zurückgegriffen, da die Definition von Jurafski sehr allgemein gehalten ist. Nadeau und Sekine [157] beschreiben im Rahmen der Sprachanalyse (siehe Abschnitt 4.2) explizit die Betrachtung von Entitäten deren rigide(r) Bezeichner zum Zeitpunkt der Analyse nicht nachgewiesen werden können bzw, kann. Eine exakte Darstellung und Erläuterung der genannten Phänomene ist in Abschnitt 6 gegeben. Nachfolgend wird näher auf den Begriff Eigenname und die Einbettung von Benannten Entitäten im Kontext der natürlichen Sprachanalyse sowie im Kontext von Ontologien eingegangen.

[10] Kripkes Aussage im Rahmen seiner Arbeit zu Ambiguität ist: „It is very much the lazy man's approach to philosophy to posit ambiguities when in trouble“ [129].

4.1.1. Eigennamen

In den meisten Arbeiten bzw. wissenschaftlichen Artikeln wird die Bedeutung bzw. Definition des Eigennamens als bekannt vorausgesetzt und nicht explizit erwähnt. Im Rahmen der Definition von Benannten Entitäten stellt der Eigennamen jedoch ein wesentliches Kriterium dar, das einer Erläuterung bedarf. Der Duden definiert einen Eigennamen als *„etwas Bestimmtes, Einmaliges [...]; er ist in der Regel einzelnen Lebewesen oder Dingen zugeordnet und gestattet diese zu identifizieren"* [64]. Diese Definition stimmt überein mit dem vorgestellten Bezug zu Entitäten (*hier:* Lebewesen und Dinge). Auch Wittgenstein beschreibt diesen Zusammenhang *„A name means an object. The object is its meaning"* [84] . Diese Definition verfasste er im Rahmen seiner Arbeit zu Ontologien (aus philosophischer Sicht), in der Objekte[11] und den Entitäten gleichgesetzt werden.

Hinsichtlich der Definition von Eigennamen weist der Duden auf den Unterschied zu Gattungsnamen hin, die eine *„Gruppe von Lebewesen oder Dingen"* bezeichnen. Hierbei definiert eine Gattung die Übereinstimmung der ihr zugeordneten Entitäten hinsichtlich vorgegebener Merkmale. Beide Definitionen sind nicht immer klar zu trennen. Dies zeigt sich bei bestimmten Nachnamen, *z.B.* „Bäcker" oder „Müller", die gleichzeitig für Berufsgruppen stehen.

Quirk et al. [186] unterscheiden „proper noun" von „proper name", indem ersteres nur aus einem Wort bestehen darf[12]. Beide bezeichnen eine eindeutige Referenz zu einem Objekt oder Lebewesen. Die Autoren unterscheiden zwischen „proper nouns without articles" und „proper nouns with articles".[13] In die erste Kategorie gehören unter anderem Ortsnamen, *z.B.* „New York". Die zweite verbindet Zählbegriffe mit Eigennamen, *z.B.* „The New York Times" (Zeitschrift). Dies

[11] Von ihm stammt ebenfalls die Aussage: *„Objects can only be names. Signs are threir representatives. I can only speak about them: I cannot put them into words. Propositions can only say how things are, not what they are".* Er drückt hiermit das Problem bei der Bezeichnung von Objekten aus.

[12] Im gängigen Sprachgebrauch wird diese Unterscheidung jedoch nicht berücksichtigt und beide Begriffe werden alternativ verwendet.

[13] Quirk gibt ebenfalls eine Unterkategorisierung dieser beiden Gattungen an. Zur ersten Kategorie zählt er Personen-, Zeit- und Geographische Namen. Die zweite Kategorie unterteilt er in unmodifizierte Begriffe (*z.B.* „Der Rhein") und in Begriffe mit einer vorausgehend bzw. nachfolgenden Modifizierung (*z.B.* „Das *Weiße* Haus" bzw. „Das Denkmal *der Gefallenen").*

weist auf die Problematik in der Erkennung von Eigennamen in Texten hin.

Der Autor dieser Arbeit schließt sich der Sichtweise von Jurafski et al. [117] an, dass Eigennamen von Entitäten anwendungsspezifisch definiert werden. Die hier angegebenen Merkmale gelten insbesondere für Verwendung von Eigennamen auf der Basis der natürlichen Sprache.

4.2. Entitäten im Kontext natürlicher Sprachanalyse

Im wissenschaftlichen Umfeld der Computerlinguistik[14] wurde der Begriff „named entity“ erst 1996 von Ralph Grishman et al. im Rahmen der sechsten Message Understanding Konferenz[15] offiziell eingeführt [95]. Die Aufgabendefinition ging zurück auf das Vorhaben, für die zu entwickelnden Analyseprozesse auf bisherige Komponenten der Informationsextraktion zurückzugreifen, *z.B.* Komponenten, die bereits Funktionen auf Ebene der Terme zur Verfügung stellten. Im Vordergrund stand insbesondere die Domänenunabhängigkeit. Hieraus entstand die „named entity“ Aufgabe, *d.h.* alle Namen, die auf Personen, Organisationen, geographische Orte, Zeitangaben, Währungen und Prozentausdrücke hinweisen, müssen im Text erkannt werden. Ein Beispiel in SGML[16] wird im Artikel von Grishman gegeben (siehe Abbildung 4.1).

```
Mr. <ENAMEX TYPE="PERSON">Dooner</ENAMEX> met with <ENAMEX TYPE="PERSON">Martin
Puris</ENAMEX>, president and chief executive officer of <ENAMEX
TYPE="ORGANISATION">Amirati & Puris</ENAMEX>, about <ENAMEX
TYPE="ORGANIZATION">McCann</ENAMEX>´s acquiring the agency with billings of <NUMEX
TYPE="MONEY">$400 million</NUMEX>, but not nothing has materialized.
```

Abbildung 4.1.: Beispiel einer Annotation basierend auf SGML

14 *engl.* Natural Language Processing (NLP)

15 MUC wurde 1990 von der DARPA (Defense Advance Research Projects Agency) ins Leben gerufen. Die Konferenz hatte zum Ziel eine Lösung zu finden, um Informationen aus *unstrukturiertem* Text zu extrahieren.

16 Standard Generalized Markup Langaguage (`http://www.w3.org/MarkUp/SGML/` [letzter Zugriff am 12.09.2011])

Es wird unterschieden zwischen ENAMEX (Entity Name Expressions) und NUMEX (Numeric Expressions). Von besonderer Relevanz ist, dass die Definition von ENAMEX als auch die Definitionen von Sekine, Kripke und Grishman auf natürlich-sprachlichen Ausdrücken für die Angabe von Namen aufbauen. NUMEX bezeichnet hingegen quantitative Ausdrücke, die formalen Ausdrücken entsprechen.[17]

Natürlich-sprachliche Ausdrücke lassen sich klar von formalen Ausdrücken unterscheiden. Die natürliche Sprache kann definiert werden als Sprache(n), die von Menschen gesprochen bzw. geschrieben wird (siehe auch Kapitel 5). Sie wird „natürlich" genannt, da sie von uns Menschen nicht wissentlich eingeführt bzw. erfunden wurde, *d.h.* sie wird natürlich erlernt und ist Teil der Evolution des Menschen [19]. Ein wesentliches Merkmal der natürlichen Sprache ist, dass sie in ihrer Ausdrucksfähigkeit *nicht* begrenzt ist. Das heißt, dass über alles damit gesprochen werden kann. Bestehende Einschränkungen basieren auf dem kulturellen und lokalen Kontext, *d.h.* Menschen in einem solchen Kontext verwenden nur eine gewisse Menge an natürlich-sprachlichen Begriffen (*z.B.* unterschiedliche Landessprachen). Jedoch können diese Begriffe eine Änderung der Bedeutung über die Zeit erfahren sowie neue Begriffe hinzukommen. Diese Eigenschaft unterscheidet die natürliche von der formalen Sprache. Eine formale Sprache findet Verwendung im Rahmen der Beschreibung eines künstlich erzeugten Modells [117]. Eine solche Sprache verfügt selbst über eine Menge an Ausdrücken, die sich aus einem begrenzten Alphabet von vordefinierten Symbolen zusammensetzen. Eine formale Sprache T wird über ein Alphabet $\sum^*$, *d.h.* $T \subseteq \sum^*$ definiert. Eine solche Sprache verfügt über eine endliche, vordefinierte Ausdrucksfähigkeit[18]. Das Modell ermöglicht demzufolge die Generierung dieser Ausdrücke und deren Erkennung.

Die von Grishman formulierte Aufgabe bezieht sich auf natürlich-sprachlichen Text, in dem die Entitäten aufgefunden werden sollen. Er begründet den Forschungszweig der

[17] Die Zuordnung von Zahlbegriffen, *z.B.* „$400 million" zu Eigennamen (wie oben definiert) wird kritisch gesehen, da sie nicht auf konkrete, sondern auf abstrakte Objekte hinweisen. Die Definition eines Eigennamens abhängig von der Anwendung ermöglicht jedoch auch solche Begriffe.

[18] Formale Sprachen gehen oftmals einher mit formalen Grammatiken. Letztere ermöglichen die eindeutige Beschreibung der Sprache und erlauben den Nachweis, ob ein gegebenes Wort Teil der Sprache ist.

„Named Entity Recognition (NER)“. Wie die obige Nennung von Entitätstypen bereits erahnen lässt, wurde von Grishman ebenfalls die „Named Entity Classification (NEC)“ als Aufgabe definiert, *d.h.* die Zuordnung des korrekten Typs bzw. Kategorie für eine gegebene Entität. Im Laufe der Zeit vergrößerte sich die Menge der zunächst definierten Entitätsklassen. Das IREX-Projekt[19] [210] erweitert diese Menge durch die Klasse „artefact“. Das „Automatic Content Extraction (ACE)“ Programm erweitert es um die Klassen „graphical/political Entities“ und „facility“. Da die Typen der verwendeten Entitäten von der aktuellen Aufgabe und der Domäne abhängen, werden die beschriebenen Entitätstypen kontinuierlich erweitert. Forscher wie Fleischmann [82], Bick [25], Etzioni [69], Zhu [251] *etc.* fügten neue Kategorien von Typen hinzu. Sekine [211] verfolgt den Ansatz die verschiedenen und teilweise widersprüchlichen Erweiterungen durch das Aufstellen einer Liste von 200 verschiedenen Kategorien einheitlich festzulegen.[20] Aus dieser Entwicklung der Kategorisierung kann geschlossen werden, dass die Domäne die Anzahl der Entitäten sowie deren Klassen vorgibt bzw. definiert. Beispielsweise fokussieren sich die MUC-Konferenzen hauptsächlich auf die Analyse von Nachrichtentexten. Für die Analyse von Texten in einer anderen Domäne, *z.B.* der Biologie, sind andere Kategorien von Entitäten, *z.B.* Gene, Proteine *etc.* notwendig, um die Domäne korrekt repräsentieren zu können.

4.3. Entitäten in Ontologien

In Kapitel 3.2 wird der Begriff der Ontologie in der Informatik, als auch in der Philosophie an der philosophische Fragestellung „Welche Klassen von Entitäten sind für eine vollständige Beschreibung und Erklärung der Vorgänge in dieser Welt nötig?“ [83] vorgestellt. Die einer Ontologie zugeordnete Aufgabe ist die Beschreibung von Entitäten, deren Zusammenhängen und der Klassen, in welche sie eingeordnet werden können. Innerhalb der Ontologiedefinition in der Informatik kommt diese Sichtweise ebenfalls zum Einsatz.

[19] **I**nformation **R**etrieval and **Ex**traction Exercise

[20] Die zeitliche Entwicklung der Entitätsklassen kann detailliert im Übersichtsdokument zu diesem Thema von Sekine [157] nachgelesen werden.

Je nach Ausdrucksmächtigkeit der Sprache stehen unterschiedliche Konstrukte zur Definition von Entitäten bereit. OWL definiert Individuals, Properties und Classes (vgl. [244]) als Entitäten gemäß der oben vorgestellten Beschreibung. Der Autor dieser Arbeit stimmt mit der Definition von Manaf et al. [147] bezüglich der Beschreibung von „Benannten Entitäten" in OWL überein: *„A named entity refers to a named class, a named individual or a named property"*. Diese Definition baut auf der Zuordnung einer URI auf, die über den *rdf:id*-Tag einer Klasse, Relation oder einem Individuum hinzugefügt wurde. Gestützt auf die Aussage von Jurafski [117], dass ein „proper name" abhängig von der jeweiligen Applikation bzw. vom Einsatzzweck ist, kann hier der URI als Eigenname interpretiert werden und somit den benötigten eindeutigen Bezeichner stellen. In die gleiche Argumentationsrichtung geht die Aussage von Steinberger und Pouliquen [220], welche im Rahmen sprachübergreifender[21] NER darauf verweisen, dass *„the language independ level is organized arroud the pivot (a conceptual proper name) which is represented by a unique identification number (ID)"*. In beiden Fällen ist eine ID, wie sie auch eine URI repräsentiert, identifizierendes Merkmal. Im Rahmen von RDF-Ontologien stellen Ressourcen Entitäten dar. Falls diese eine korrekte URI referenzieren (*d.h.* keine „blank nodes" repräsentieren), entsprechen sie ebenfalls den Anforderungen für eine „Benannte Entität".

Von besonderer Relevanz für diese Arbeit ist die Unterscheidung zwischen natürlich-sprachlichen Namen und formalen Namen, wie im vorherigen Abschnitt vermerkt. Die formale Syntax für die Erzeugung einer URI ist im RFC2396 festgehalten [21]. Für die Zuordnung zu natürlich-sprachlichen Quellen und damit auch zur besseren Verständlichkeit für den Menschen sind natürlich-sprachliche Namen für die oben genannten Ontologieelemente notwendig. Das Vokabular, welches durch RDF(S) Standard zur Verfügung gestellt wird, erlaubt die Angabe von natürlich-sprachlichen Bezeichnern durch das *rdfs:label*-Tag. Hier kann ein beliebiger optional durch Leerzeichen getrennter String verwendet werden. Des Weiteren ist es möglich einem Element mehrere solcher *rdfs:label*-Tags[22] zuzuordnen.

[21] Die Methode ist nicht auf eine Sprache, *z.B.* Deutsch, beschränkt, sondern kann auf mehrere Sprachen angewendet werden.

[22] Im Rahmen dieser Arbeit wird in diesem Zusammenhang der englischen Begriff „Tag" verwendet, um auf Sprachelemente eines Metadatenstandards Bezug zu nehmen.

Der Autor dieser Arbeit stimmt mit Manaf et al. darin überein, dass ein Elementname von Bedeutung für den Menschen ist, falls er eine Beziehung zum zu identifizierenden Objekt aufweist. Beispielsweise kann eine Klasse, der Personen zugeordnet sind, durch den Namen „Person“ identifiziert werden. Ein solcher Name wird als „semantischer Bezeichner“ spezifiziert. Dies steht im Gegensatz zu einer Beschreibung durch eine URI, die diese Bedingung nicht erfüllt.[23]

Die Angabe von natürlich-sprachlichen Bezeichnern kann über *rdfs:label* oder über eigene Datatype-Properties erfolgen. Gegenüber der eigenen Definition zu bevorzugen sind jedoch definierte und gängige Vokabulare, die bereits Tags für die Bezeichnung von Elementen zur Verfügung stellen. Der Rückgriff auf öffentliche Vokabulare wird in der Semantic Web FAQ[24] ausdrücklich empfohlen: *„the ‚ethos‘of the Semantic Web is to share and reuse as much as possible“*. Eines der standardisierten und bekannten Vokabulare ist das *Simple Knowledge Organization System (SKOS)*. Es wurde 1999 ins Leben gerufen, um unter anderem die Spezifikation von Thesauri, Subject-Headings und Taxonomien zu ermöglichen. Insbesondere aus der Möglichkeit Thesauri darzustellen resultiert die Möglichkeit, Beschreibungen für Bezeichner von Benannten Entitäten abzuleiten. SKOS definiert diesbezüglich Beschreibungen, wie beispielsweise *skos:prefLabel*, um einer Entität einen primären, *d.h.* bevorzugten, Bezeichner zuzuweisen. Mit *skos:alterLabel* können alternative Bezeichner hinzugefügt werden. Diese Möglichkeiten der Bezeichnung setzt die Definition einer Entität mit einem bevorzugten rigiden Bezeichner sehr adäquat um. Beispielsweise kann der vollständige Name „Christian Schäuble“ als bevorzugter Bezeichner verwendet werden während „Christian“, „Schäuble“, „Chris“ *etc.*, *d.h.* Vorname, Nachname, Spitzname *etc.* als Alternativbezeichnungen angegeben werden können.

[23] Manaf et al. beschreiben im oben genannten Artikel Techniken zur Auflösung einer URI in einen natürlich-sprachlichen Ausdruck. Dies erfordert jedoch eine Voranalyse basierend auf einer Heuristik.

[24] `http://www.w3.org/2001/sw/SW-FAQ#` [letzter Zugriff am 12.09.2011]

4.4. Gegenüberstellung der beiden Gebiete

Tim Berners-Lee beschreibt den Unterschied durch die Aussage *„The Semantic Web will enable machines to COMPREHEND semantic documents and data, not human speech and writings“* [22]. Aus seiner Sicht sind auf Grundlage dieses Zitats beide Welten miteinander unvereinbar, da es die primäre Aufgabe der Sprachanalyse ist, zum Verstehen der menschlichen Sprache beizutragen und hierfür maschinelle Methoden einzusetzen.

Dieses Zitat beschreibt jedoch den Zusammenhang nicht vollständig. Viele Verfahren der Sprachanalyse bauen auf Wissensbasen auf bzw. verwenden diese innerhalb ihrer Methoden. Der Grund hierfür ist offensichtlich: Ein Kernmerkmal natürlich-sprachlicher Texte ist deren Unstrukturiertheit.[25] Jegliche Art der Struktur muss bestimmt werden, *z.B.* das Erkennen von Entitäten (NER), deren Klassifikation (NEC), das Erkennen von Zusammenhängen zwischen ihnen *etc.* Wissensbasen im Gegensatz dazu sind strukturiert. Sie enthalten bereits Zusammenhänge zwischen Entitäten und je nach Art der Wissensbasis auch weitere Informationen, *z.B.* deren Typzugehörigkeit. Wissensbasen in Form von Ontologien sind hierfür verwendbar. Somit tragen Ontologien indirekt zum Verständnis von Informationen bei, die in auf natürlicher Sprache basierenden Quellen enthalten sind.

Ein wesentlicher Unterschied liegt in der Repräsentation von Entitäten. Entitäten in natürlich-sprachlichen Texten können erwartungsgemäß aufgrund den im Text erwähnten natürlich-sprachlichen Bezeichnern identifiziert werden. Entitäten in Ontologien müssen nicht zwangsläufig über einen natürlich-sprachlichen Bezeichner verfügen. Es gibt jedoch auch Ontologien, die speziell zur Abbildung von natürlich-sprachlichen Bezeichnern und deren Zusammenhänge entworfen wurden. Solche Ontologien definieren somit eine lexikalische Datenbasis. Ein derartiges Beispiel ist die zuvor bereits erwähnte „Wordnet“-Ontologie, die lexikalische Zusammenhänge zwischen natürlich-sprachlichen Wörtern abbildet, *z.B.* Synonyme, Worttypen (Hauptwörter *etc.*), Kategorien *etc.*

[25] Abgesehen von der Strukturierung nach Kapiteln, Abschnitten *etc.*

4.5. Entitäten in anderen Gebieten der Informatik

Die in Abschnitt 4.1 vorgestellte allgemeine Entitätsdefinition erlaubt eine sehr generische Verwendung des Begriffs „Entität“. Oftmals erfolgt eine Gleichstellung der Begriffe „Entität“ und „Objekt“. Hierdurch wird der Einsatz des Begriffes häufig unbestimmt und nur schwer begründbar. Es gibt jedoch Teilgebiete in der Informatik, die den Begriff in einer ähnlichen bzw. nahezu übereinstimmenden Art und Weise mit den zuvor vorgestellten Bereichen der Sprachanalyse oder der Wissensinterpretation verwenden. Hervorzuheben ist der Bereich der Datenbanken. Insbesondere das relationale Datenbankmodell. Dieses ist in den Grundzügen auf das Entity-Relationship Model von Peter Chen zurückzuführen, welches *„incorporates some of the important semantic information about the real world“ [45].*

Der Begriff der Entität findet auch im Bereich der Unified Modelling Language (UML) Anwendung. Beispielsweise werden Akteure in Use-Case Diagrammen als Entitäten bezeichnet und es existiert ebenfalls der Begriff des Entity-Concepts[26] (vgl. [6]). UML-Klassen-Diagramme ermöglichen die Repräsentation von Klassen eines zugrundeliegenden Systems, die Beziehungen zwischen diesen Klassen sowie deren klassenspezifische Attribute und Operationen. Die Diagramme ermöglichen die Klassen innerhalb der modellierten Domäne auszuwerten und die Anforderungen als konzeptuelles Modell zu erfassen.

Es werden Entitäten verwendet, um ein applikationsspezifisches Domänenmodell zu erstellen. Es besteht die Möglichkeit eine konzeptuelle Zuordnung vorzunehmen sowie Entitäten Eigenschaften zuzuweisen. Interaktionen zwischen Entitäten können zudem durch Relationen ausgedrückt werden. Diese Merkmale zeichnen ER-Modelle und UML-Diagramme als Wissensmodelle aus. Beide Modelle können in das Format der in Kapitel 3.2 eingeführten Ontologien überführt und somit dem Bereich der Wissensmodelle (vgl. 4.3) zugeordnet werden.

[26] *„An entity concept is an abstraction representing passive data and information“* [6]

5. Ambiguität

Von William G. Lycan stammt bezüglich Ambiguität die Aussage *„a name is unambgiuous only by historical accident“* [142]. Er bezog sich darauf, dass selbst Personennamen, die normalerweise unverwechselbar sind, *d.h.* eindeutig zugewiesen werden können, im Laufe der Zeit mehrfach vergeben werden. Im Zusammenhang hiermit beschrieb Neill Ambiguität bei der Analyse von Information als allgegenwärtiges Problem. Von ihm stammt ebenfalls die Aussage *„information processing yields ambiguity“* [160] . Besonderen Stellenwert besitzt diese Problematik im Bereich der Textanalyse, *d.h.* der Linguistik. Sarah Schrauwen sieht in Ambiguität: *„One of the biggest problems in the field of Computational Linguistics“* [207]. Das Problem liegt in der Missinterpretation von Daten, falls die Auflösung der Ambiguität nicht durchgeführt werden kann und dies somit zu Verlusten bzw. Fehlern in der übermittelten Information führt. Diese Gefahr macht Ambiguität (Mehrdeutigkeit) von lexikalischen Vokabeln[1], die als Bezeichner von Entitäten verwendet werden, zu einem Schwerpunkt der vorliegenden Arbeit und ist für diese von zentraler Bedeutung.

Zunächst werden die verwendeten Begriffe in Abschnitt 5.1 eingeführt. Der Bezug zur linguistischen Definition von Mehrdeutigkeit sowie deren Unterarten wird in Abschnitt 5.2 vorgestellt. Abschnitt 5.3 beinhaltet die lexikalische Darstellung von Mehrdeutigkeit. Ein allgemeines Modell für Mehrdeutigkeit wird in Abschnitt 5.4 präsentiert. Abschnitt 5.5 überträgt und spezifiziert das darauf basierende Modell für Mehrdeutigkeit in Wissensbasen. Auf mehrdeutige Entitäten wird in Abschnitt 5.6 eingegangen.

[1] In diesem Kontext wird von lexikalischen Vokabeln gesprochen, da diese Benennungen/Wörter auf einem Wörterbuch basieren. Als Wörterbuch wird im Allgemeinen ein Lexikon einer Sprache verstanden, *z.B.* 'Wahrig - Wörterbuch der *deutschen* Sprache'. Die Repräsentation lexikalischer Zusammenhänge in Ontologien wird im Laufe des Kapitels erläutert.

5.1. Begriff, Benennung, Bedeutung und Referent

Im Folgenden wird der kognitive Prozess vorgestellt, der bei der Bestimmung der Bedeutung eines Bezeichners vollzogen wird. Ausgangspunkt für diese Bestimmung ist zunächst die Konfrontation mit einem Bezeichner, *z.B.* „Baum“, dessen Bedeutung festgestellt werden soll. Der Zusammenhang zwischen einem Bezeichner (Benennung), dem konkreten Objekt und der zugehörigen Bedeutung (Begriff) ist im semiotischen Dreieck[2] [167][3] der Abbildung 5.1 dargestellt.

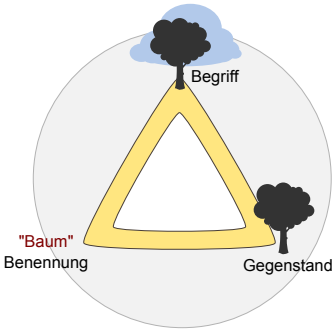


Abbildung 5.1.: Semiotisches Dreieck

Eine Benennung ist gegeben durch einen sprachlichen Ausdruck, *d.h.* eine Buchstabenfolge. Jede Benennung spezifiziert einen Gegenstand, *d.h.* ein Objekt, und somit einen beliebigen Ausschnitt aus der konkret wahrnehmbaren oder der vorstellbaren Welt. Taucht beispielsweise der Bezeichner „Baum“ im Kontext des Satzes „In unserem Garten hängen bereits Äpfel am *Baum*“ auf, so kann geschlussfolgert werden, dass es sich hierbei um einen konkreten Baum im Garten des Sprechers handelt. In diesem Beispiel bezieht sich die Benennung somit auf ein Objekt in der realen Welt. Es können ebenfalls immaterielle oder

[2] Mit *Semiotik* wird die Theorie vom Wesen, der Entstehung und dem Gebrauch von Zeichen bezeichnet (siehe auch [114]).

[3] Die Ersterscheinung des Buches von Odgen und Richards war bereits 1923. Diese Autoren beschreiben in ihrem Werk das dreiseitige Zeichenmodell, auf das in dieser Arbeit Bezug genommen wird (vgl. Abbildung 5.1).

abstrakte Gegenstände bezeichnet werden (siehe Kapitel 4). Eine Benennung ist zugleich immer mit einem Begriff verbunden. Ein Begriff kennzeichnet einen Denkprozess, der eine Menge von Gegenständen mit gemeinsamen Eigenschaften beinhaltet, basierend auf einem Abstraktionsprozess. Dieser enthält zudem Beziehungen zwischen Gegenständen und deren konzeptuelle Abhängigkeit. Der Sprachwissenschaftler Ferdinand de Saussure, der als Begründer der modernen Linguistik betrachtet wird, bezeichnete den Begriff als ein „mentales Bild", das durch die Benennung erzeugt wird.

Müller beschreibt *„Bedeutung ist das, was mit einem sprachlichen Ausdruck assoziiert wird"* [156]. Griebel definiert Bedeutung als einen Bewusstseinsinhalt, der zum einen ein Ergebnis des gesellschaftlichen Erkenntnisprozesses und zum anderen ein Resultat kommunikativer Tätigkeit ist [94]. Beide Ausführungen verdeutlichen, dass diese Definitionen mit der des oben genannten „Begriffs" gleichgesetzt werden können.

Im Rahmen dieser Arbeit ist zudem die Bezeichnung der „Referenz" von Bedeutung. Müller entwickelte eine Definition, die besagt, dass eine Referenz sich auf ein konkretes Objekt bezieht, das in der aktuellen Situation eindeutig identifizierbar ist. Beispielsweise ist die Referenz in der realen Welt, auf die sich der Ausdruck „Hund" im Satz „Ich gehe mit dem Hund spazieren" bezieht, der Hund, der gerade neben mir läuft. Ein weiteres Beispiel ist die Beziehung zwischen „Baum" als Benennung/Ausdruck und dem Objekt Baum. Das Objekt ist die Referenz des Ausdrucks. Man bezeichnet daher Referenz als ein Phänomen der Sprachverwendung, während Bedeutung Teil des Sprachsystems ist. Somit kann nicht in jedem Zusammenhang, *d.h.* in jeder Situation, eine gültige Referenz erzeugt werden.

5.2. Arten von Mehrdeutigkeit

Die Zuordnung eines Wortes zu einem Gegenstand ist oftmals *nicht* eindeutig, *d.h.* mehrere Gegenstände lassen sich durch denselben Ausdruck identifizieren. Dieser Umstand wird als *Mehrdeutigkeit, Ambiguität bzw. Vagheit* bezeichnet. Im alltäglichen Sprachgebrauch werden diese Begriffe häufig alternativ verwendet, jedoch ist deren Bedeutung aus linguistischer Sicht klar zu trennen. Norbert Fries [85]

beschreibt „Mehrdeutigkeit“ als Oberbegriff von „Ambiguität“ und „Vagheit“. Unter Mehrdeutigkeit wird die Möglichkeit verstanden eine Einheit in mehrfacher Weise zu interpretieren. Die Aussage von Edgar Schneider [203] bezüglich des Unterschieds zwischen Ambiguität und Vagheit lautet *„**Ambiguitäten** sind also zu diesem Thema [203] disambiguierbar durch eine potentielle Selektion aus einer begrenzten und diskreten Menge eindeutiger Interpretationen. Bei **Vagheiten** steht eine derartige Selektionsmenge nicht zur Verfügung; [...][4]“*.[5] Ein Begriff wie „Schloß“ wird beispielsweise als ambig bezeichnet, da diesem verschiedene Bedeutungen klar zugewiesen sind, *z.B.* ein Schließmechanismus (technische Interpretation), ein repräsentatives Gebäude (architektonische Interpretation) *etc.* Jedoch ist die vage Aussage „Ich komme *gleich*“ unbestimmt und kann daher keiner konkreten Bedeutung zugewiesen werden.[6] Klavans [126] führt zudem den Begriff der „Generativity“ ein, welcher beschreibt, dass Wörter in neuen, kreativen Konstellationen gebraucht werden. Hier fehlt jedoch die Grundlage der Beschränktheit auf eine gegebene Selektionsmenge, da die Verwendung des Wortes zuvor in einer solchen kreativen Konstellation nicht definiert wurde.

Wie bereits von Neil beschrieben, ist Ambiguität allgegenwärtig. Seiner Aussage folgte *„[...] awareness is the resolution of that ambiguity“* [160]. Dieses Bewusstsein zu erlangen, erfordert eine detaillierte Übersicht über die verschiedenen Arten von Ambiguität[7] (vgl. Abbildung 5.2)[8]. Im Folgenden werden die einzelnen Unterkategorien von Ambiguität dargestellt. Darauf baut Abschnitt 5.5 auf, indem dort die Auswirkungen jeder jeweiligen Art von Ambiguität in Ontologien beschrieben werden und somit aufgezeigt wird in welcher Art sich Ambiguität dort äußert. Eine Übersicht der verschiedenen Arten der Mehrdeutigkeit ist ebenfalls in Anhang A dargestellt.

---

4 *„sie sind gegebenenfalls durch weitere Informationen präzisierbar, jedoch ist der Erhalt dieser weiteren Informationen von der kommunikativen Situation abhängig und hat mit der jeweiligen sprachlichen Form im Grunde nichts zu tun“*

5 Diese Unterscheidung findet sich ebenfalls bei den diesbezüglichen Definitionen weiterer Wissenschaftler (*z.B.* Pinkal [177] und Rieger [189]).

6 „gleich“ ist hierbei ein variabler Zeitausdruck. Je nach Person, die ihn verwendet, kann der Begriff sich von Sekunden bis hin zu wenigen Minuten steigern.

7 Im Anhang A befindet sich eine detaillierte Auflistung mit Kurzbeschreibung und Beispielen.

8 Die Hauptquellen für die erstellte Unterkategorisierung von Ambiguität sind [174, 173, 14, 12, 121, 41, 138, 42, 124, 107]. Diese sind auch die Quellen für Erstellung der Abbildung 5.2.

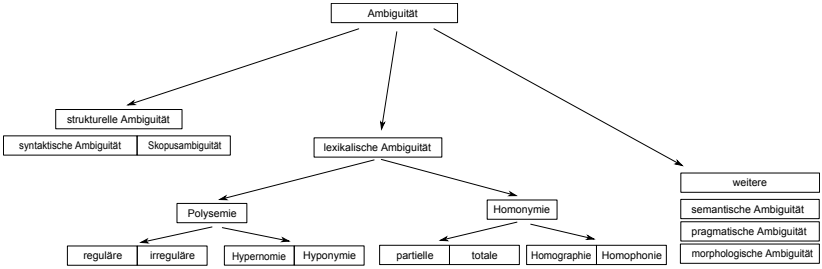


Abbildung 5.2.: Arten von Ambiguität

5.2.1. Polysemie

„Polysemie ist seit mehr als einem Jahrzehnt eines der am intensivsten untersuchten Gebiete der lexikalischen Semantik – vielleicht sogar das am intensivsten untersuchte“ [173]. Diese Aussage aus dem Jahr 2001 verdeutlicht die Problematik und die Bedeutung, die mit Polysemie[9] verbunden ist. Polysemie ist definiert als:

Definition 5.1. *Polysemie beschreibt das Phänomen, dass ein Wort zwei oder mehr Bedeutungen[10] hat. Diese Bedeutungen stehen in Beziehung zueinander.*

Das Wort geht hierbei auf ein Lexem[11] zurück. Das bedeutet, dass die damit verknüpften Bedeutungen auch untereinander Gemeinsamkeiten oder Beziehungen zueinander aufweisen.[12] Dies ist in Abbildung 5.3 dargestellt. Polysemie entsteht zum einen durch das Teilen der Kernbedeutung, zum Beispiel wird „Pferd“ auf ein Lexem bezogen, das (1) „Pferd“ als Tier und (2) „Pferd“ als Schachfigur bezeichnet. Beide Bedeutungen sind durch die äußerlichen Merkmale, die sie teilen, miteinander verbunden. Ein weiteres Beispiel sind die Bedeutungen „Gebäude“ und „Institution“ für den Begriff „Schule“. Hierbei existiert die Property „residiert“ zwischen den Bedeutungen. Diese spiegelt die Beziehung zwischen den Bedeutungen wieder. Zum

9 Der Begriff stammt aus dem Griechischen *poly* „viel“ und *sema* „Zeichen“.

10 Pethö [174] weist darauf hin, dass eine „Bedeutung“ abhängig vom vorliegenden Ansatz ist (siehe auch Abschnitt 5.5.2).

11 Lexem bezeichnet eine Gruppe von Wörtern, die wesentliche Eigenschaften teilen, *z.B.* Grundbedeutung und Wortform.

12 Alternativ spricht man von der Verwandtschaft auf der Inhaltsebene.

anderen geht die Polysemie von Wörtern auf ihre etymologischen[13] Eigenschaften zurück, die demselben Lexem zugeordnet sind. Zum Beispiel besitzt das Wort „Flügel“ die beiden Bedeutungen (1) Flugorgan von Vögeln/Insekten und (2) seitlicher Teil eines Hauses.

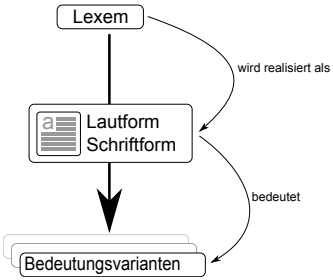


Abbildung 5.3.: Polysemie aus [140], S. 62

Zwischen den Kernbedeutungen polysemer Ausdrücke bestehen somit semantische Relationen. Diese werden im Allgemeinen unterschieden in:

- Enthaltenseinbeziehungen (semantisch übergeordnet (Hypernomie), semantisch untergeordnet (Hyponymie))[14]
- Übertragungsbeziehung (Ähnlichkeitsbeziehung[15] (Metapher), Abbildung[16] (Metonymie))

Die Begründung für die Existenz bzw. die Bedeutung von Polysemie hat Hakulinen treffend bezeichnet: *„Ohne Polysemie“* [bzw. Ambiguität im Allgemeinen] *„ , wenn jeder Begriff, jede Vorstellung, jede Bedeutungsnuance, jede gedankliche Regung einen auch lautlich differenzierten Ausdruck haben müsste, wäre das Erlernen einer Sprache in der Praxis eine nicht zu bewältigende Aufgabe, wodurch es wiederum unmöglich wäre, dass sie auf eine derartige sinnlose Grenzenlosigkeit anwächst“* [102]. Dies zeigt

[13] Etymologie beschreibt die Geschichte der Sprache, *d.h.* deren zeitliche Entwicklung.

[14] Beispielsweise schließt der Begriff „Fahrrad“ die Bedeutungen Rennrad und Klapprad mit ein.

[15] Dieser Ausdruck bezeichnet eine bildhafte Umschreibung, *z.B.* „Tischbein“.

[16] Dieser Ausdruck steht für eine Namensübertragung aufgrund Sinnähnlichkeit, z.B. diese Möbel sind aus „Kiefer“ (Objekt für Stoff).

erstens die grenzenlose Verbreitung des Phänomens Polysemie und zweitens dessen vollständige Integration in unseren Sprachalltag.

Abgesehen von der Grundbedeutung und der etymologischen Unterscheidbarkeit führte Apresjan [12] zwei unterschiedliche Kategorien ein: reguläre und irreguläre Polysemie.

Definition 5.2 (Systematische (Reguläre) Polysemie)**.**

Apresjan [12] formulierte diese Art der Polysemie durch folgende Definition[17], die durch zwei weitere Punkte ergänzt werden kann:

- *Polysemie wird als **regulär** bezeichnet, falls das Wort A Sememe[18] A_i und A_j aufweist und sich zumindest noch ein Wort B findet, dessen Sememe B_i und B_j in den gleichen semantischen Relationen zueinander stehen wie A_i und A_j. Dabei dürfen A_i und B_i sowie A_j und B_j untereinander nicht synonym[19] sein [12].*
- *Es kann eine Regel abgeleitet werden, welche die konkrete Instanz der regulären Polysemie beschreibt.*
- *Diese Regel ist interlingual und bleibt bei die Übersetzung des Begriffs in andere Sprachen erhalten.*

Beispiele sind hierfür „Glas“ als Trinkgefäß und „Glas“ als Angabe einer Getränkemenge. Dies gilt ebenfalls für „Flasche“,„Eimer“,„Tasse“ *etc.* Auch andere Zusammenhänge sind davon betroffen, *z.B.* „Schule“ als Gebäude, als Institution, als Ansammlung der ihr zugehörigen Personen *etc.* Auch ersichtlich in „Die Schule ist renovierungsbedürftig“, „Die Schule benötigt mehr Mittel“ und „Die Schule macht einen Ausflug“. Die Regel, die hier abgeleitet werden kann, lautet: Vom Namen einer Institution ausgehend kann man sich in der Regel auf die

[17] Der erste Aufzählungspunkt geht auf seine Definition zurück. Die zwei weiteren Punkte finden sich beispielsweise bei Pethö [174].

[18] Ein Semem bezeichnet die Hauptbedeutung eines Wortes sowie die semantischen Merkmale (*z.B.* Stadt: Eigenschaft: über 50000 Einwohner *etc.*). Letztere werden als Seme bezeichnet. Ein Sem ist die kleinste Einheit der Bedeutung sprachlicher Zeichen (vgl. [94]).

[19] Synonym bedeutet „gleichnamig“ bzw. „gleichbedeutend“. Meines Erachtens trifft in diesem Kontext die Definition nur für *strikte* Synonymie zu, *d.h.* die Synonyme, *z.B.* Streichholz und Zündholz besitzen eine *exakt* gleiche Bedeutung, und sind nicht auf eine partielle Überlappung beschränkt. Beispielsweise überlappen sich die Begriffe „Er ist etwas *wirr*“ und „Er ist etwas *durcheinander*“.

Gesamtheit der Menschen beziehen, die dieser Institution angehören sowie auf das Gebäude, in dem diese Institution ihren Sitz hat bzw. das diese Menschen beherbergt. Das gilt ebenfalls für „Universität", „Firma" *etc.*

Eine zentrale Eigenschaft von systematischer Polysemie ist, dass diese nur kontextabhängig aufgelöst werden kann. Grund hierfür ist ihre unterspezifizierte lexikalische Bedeutung.[20] Dies ist ebenfalls den obigen Beispielen zu entnehmen. Diese Eigenschaft wird auch als *konzeptuelle Fokussierung* bezeichnet (vgl. Kiefer [121]). Nur der Kontext „renovierungsbedürftig" und die Ableitung, dass von den gelisteten Bedeutungen nur Gebäude (lokal) renovierungsfähig sind, ermöglicht eine Disambiguierung. Zusammenfassend resultiert der systematische Charakter dieser Art von Polysemie aus der Existenz bestimmter *genereller* Beziehungen zwischen Gegenständen unterschiedlicher *Sorte*[21] (siehe hierzu Abschnitt 6.1.2). Die Systematik ist ferner durch die Übertragbarkeit dieser Beziehung auf mehrere (verschiedene) polyseme Begriffe gegeben (vgl. Flasche, Glas, Becher *etc.* im obigen Beispiel).

Die Regularität dieser Art der Polysemie ist ebenfalls typisch für das Phänomen der Metonymie[22] (vgl. [12]).

Definition 5.3 (Nicht-systematische (Irreguläre) Polysemie)**.**

Diese ist definiert durch:

- *„Polysemy is called irregular if the semantic distinction between a_i and a_j is not exemplified in any other word of the given language." [12]*
- *Sie ist nicht interlingual.*
- *Es kann keine übergreifende Regel angegeben werden.*
- *Die verschiedenen Bedeutungen stehen in mindestens einem Merkmal in einer Beziehung zueinander.*[23]

[20] Dies äußert sich darin, dass der Eintrag im Lexikon selbst für eine Auflösung der Mehrdeutigkeit nicht ausreichend ist (siehe Abschnitt 5.3).

[21] Die Bedeutungen können somit in Gruppen klassifiziert werden, *z.B.* „Tisch" zu „Möbel".

[22] Metonymie bezeichnet die Verwendung eines Wortes im übertragenen Sinn.

[23] vgl. allgemeine Kriterien für Polysemie

Während reguläre Polysemie im Wesentlichen durch Metonymie entsteht, ist irreguläre Polysemie eher auf metaphorische Ausdrücke zurückzuführen. Die Abbildung der semantischen Beziehungen zwischen zwei der hierbei beteiligten Konzepte lässt sich durch ein konzeptuelles Mapping der gemeinsamen Eigenschaften darstellen (vgl. [132]).

Beispielsweise kann der Ausdruck „laufen“ dieser Art von Polysemie zugeordnet werden. Mögliche Bedeutungen sind unter anderem:

- man ist zu Fuß in Bewegung
- ein Gerät ist in Bewegung, *z.B.* Auto
- ein Vertrag, der eine bestimmte Laufzeit besitzt

Gemeinsam ist diesen Bedeutungen, dass sie eine zeitlich befristete Bewegung bzw. Gültigkeit darstellen.

Auch für diese Art der Polysemie ist die Situation ausschlaggebend, in der die Äußerung stattfindet. Die Zuordnung bzw. Bestimmung initialer Bezeichner (Namen) für Objekte ist nicht zwangsläufig nachvollziehbar. Pethö [174] führt hierfür das Beispiel des englischen Wortes „glass“ an. So bedeutet dieser (1) Material, (2) Behälter und (3) Brille. Gemeinsames Merkmal ist das Material, das (1) bezeichnet und aus dem (2) und (3) hergestellt werden. Warum der Behälter oder die Brille jedoch mit „glass“ bezeichnet wurden ist logisch nicht nachvollziehbar, da genauso gut andere Begriffe hätten gewählt werden können. Dies wird deutlich durch das deutsche Wort „Brille“, das diese spezifische Mehrdeutigkeit[24] nicht enthält. Somit ist keine systematische, sprachübergreifende Bedeutung gegeben.

[24] Dafür gibt es andere Mehrdeutigkeiten, *z.B.* „Brille“ als Sehhilfe oder als Toilettensitz (Homonym).

5.2.2. Homonymie

Eine weitere Unterkategorie lexikalischer Ambiguität ist Homonymie[25]. Bußmann definiert Homonymie als:

Definition 5.4 (Homonymie)**.** *„Homonyme Ausdrücke verfügen über die gleiche Ausdrucksform hinsichtlich Orthographie[26] (=Homographie) und Aussprache (=Homophonie) bei unterschiedlicher Bedeutung und oft verschiedener etymologischer Herkunft" [41]*

Das Wort geht auf *zwei* syntaktisch gleiche Lexeme[27] zurück. Dies bedeutet, dass die damit verknüpften Bedeutungen untereinander im Allgemeinen keine Gemeinsamkeiten oder Beziehungen zueinander aufweisen. Dies ist auch in Abbildung 5.4 dargestellt. Ein Beispiel hierfür ist der Ausdruck „Kiefer", der zum einen Nadelbaum und zum anderen einen Knochen beschreibt.

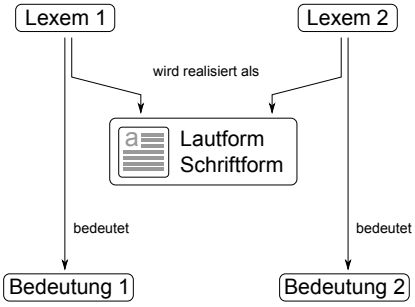


Abbildung 5.4.: Homonymie aus Löbner [140], Seite 62

Unterschieden werden die **partielle** Homonymie und die **totale** Homonymie. Ausschlaggebend ist, ob die Übereinstimmung für alle Formen der beteiligten Wörter gegeben ist. „Bank" (Sitzgelegenheit/ Geldinstitut) ist ein Beispiel für partielle Polysemie, *d.h.* für die Form „Bänke" ist keine Übereinstimmung mit „Banken" vorhanden. Im Fall des Bezeichners „Weiche" (Schienenübergang/Körperflanke) ist totale Homonymie gegeben. Bezüglich der Lautform (Aussprache)

[25] Begriff stammt aus dem Griechischen *homonnymia*, die „Gleichnamigkeit".
[26] Schreibweise
[27] Ein Lexem beschreibt eine unabhängige Einheit im Wörterbuch.

unterscheidet man **Homographie**, *z.B.* „T̲enor“ und „Ten̲or“, sowie **Homophonie**, *z.B.* „Seite“ und „Saite“.

Wichtigstes Merkmal der Homonymie ist das *Fehlen semantischer Beziehungen zwischen den Bedeutungen syntaktischer bzw. in der Aussprache übereinstimmender Wörter*.

5.2.3. Strukturelle Ambiguität

Diese spezielle Art der Ambiguität tritt bei der Interpretation von Sätzen auf. Einzelne Phrasen eines Satzes können einander unterschiedlich zugeordnet und somit unterschiedliche Aussagen erzeugt werden. Strukturelle Ambiguität wird unterschieden in syntaktische Ambiguität und Skopusambiguität.

Syntaktische Ambiguität Diese Art von Mehrdeutigkeit geht auf *syntaktische* Merkmale zurück. Die Art und Weise, verschiedene Phrasen (Satzteile) miteinander in Beziehung zu setzen, entscheidet über die Satzaussage, *d.h.* die Information, die mit dem Satz vermittelt wird. Der gegebene Satz „Heike lief Klaus mit dem Stift hinterher“ kann je nach Phrasenbeziehung interpretiert werden als „Heike lief [*Klaus*[*mit dem Stift*]] hinterher“ oder als „Heike [*lief* [Klaus] *mit dem Stift*] hinterher“. *„Syntaktische Mehrdeutigkeiten sind eine Folge des Regelsystems. Sie brauchen keine semantischen Mehrdeutigkeiten nach sich zu ziehen“* [246]. Insofern steht bei dieser Art der Mehrdeutigkeit die grammatikalische, *d.h.* regelbasierte, Analyse im Vordergrund.

Skopusambiguität Diese liegt vor, falls die Beziehungen zwischen den Skopi[28] von logischen Teilausdrücken eines Ausdrucks mehrdeutig sind. Ein typisches Beispiel hierfür ist „Jeder Student hat ein Buch gelesen“. Es sind zwei Interpretationen möglich: (1) Alle Studenten haben das gleiche Buch gelesen oder (2) Jeder Student hat sein eigenes Buch gelesen. Die logische Ordnung zwischen verschiedenen Satzteilen ist bei dieser Form von Ambiguität unklar.

[28] Skopi = Reichweite eines Quantors/Ausdruck

5.2.4. Pragmatische und Morphologische Ambiguität

Pragmatische Ambiguität Bei dieser Art von Ambiguität interpretiert der Hörer/Leser die Intention des Sprechers/Autors falsch. Beim Beispielsatz „Wir treffen uns nächsten Donnerstag“ könnte *z.B.* der Donnerstag in dieser oder in der nächsten Woche gemeint sein.

Auch kann dieser Fall auftreten, falls der Verweis mithilfe eines Pronomens auf eine Entität unklar ist (anaphorischer Gebrauch des Pronomens). Die Aussage „Er ließ den Stift auf den Tisch fallen und zerkratzte ihn“ ist mehrdeutig hinsichtlich des Objektes, das zerkratzt wurde - entweder der Stift oder der Tisch.

Morphologische Ambiguität Diese Art von Ambiguität wird auch als kategoriale Mehrdeutigkeit bezeichnet und tritt auf, wenn ein Wort in verschiedenen Wortklassen gebraucht wird. Beispielsweise bedeutet der englische Begriff „bark“ in der Verbform „bellen“ und als Substantiv „Borke“. Im Deutschen kann durch die Groß-/Kleinschreibung dieses spezifische Problem teilweise umgangen werden.[29] Jedoch gilt sie immer noch für Flexions-[30], Derivations[31]- und Kompositionsausdrücke[32]. Beispielsweise wird die Aussage des Wortes „Sinn“ mit dem Wort „Bedeutung“ gleichsetzt. „Sinnlich“ hingegen bezeichnet man eine Situation, welche die Sinne anspricht und an der die Beteiligten Gefallen finden.

5.2.5. Semantische bzw. Referenzielle Ambiguität

Ein weiterer Fall von Ambiguität umfasst mehrdeutige Begriffe innerhalb eines Satzes, deren verschiedene Interpretationsmöglichkeiten Einfluss auf die Gesamtaussage des Satzes haben. Diese Mehrdeutigkeit tritt dann auf, wenn Leser und Autor einem Wort oder einem

[29] Es bleibt die Problematik am Satzanfang, da das dort verwendete Wort einen groß geschriebenen Buchstaben besitzen muss.

[30] Dies bezeichnet die Gestaltung eines Wortes hinsichtlich der grammatikalischen Situation innerhalb des Satzes.

[31] Dies bezeichnet die Bildung neuer Wortformen.

[32] Dies betrifft die Zusammenführung unterschiedlicher Wörter zu einem neuen Wort, *z.B.* Schiffsrumpf.

Satzteil verschieden interpretieren, *d.h.* verschiedene Bedeutungen zuordnen. In der Konsequenz ergeben sich unterschiedliche Interpretationen des Satzes einerseits für den Leser und andererseits für den Autor. Dieser Fall tritt vor allem dann ein, falls der Autor die Aussage des Satzes nicht exakt formuliert. Der Satz „Iraqi head seeks arms" beispielsweise auf zwei verschiedene Arten interpretiert werden, *z.B.* (1) head = chief oder (2) head = atomical part of a body. Der Autor dieser Arbeit stimmt mit Jerold Katz überein, der folgende Definition verfasste:

Definition 5.5. *„Semantic ambiguity [...] occurs when an underlying structure*[33] *contains an ambiguous word or words that contribute its (their) multiple senses to the meaning of the whole sentence" [120]*

Insofern ist es eine hinreichende Bedingung, dass mindestens ein Wort über mehrere Bedeutungen verfügt und in der Konsequenz ambig ist. Die obige Definition wird durch den Hinweis erweitert, dass diese Wörter in einem Lexikon hinterlegt sind und somit auf der lexikalischen Ambiguität, *d.h.* Polysemie oder Homonymie, beruhen. Je nach Interpretation kann der zugrundeliegende Satz eine völlig neue Aussage erhalten. Semantische Ambiguität ist von struktureller Ambiguität klar zu trennen. Der Begriff Lexikon ist variabel definierbar und kann folglich auch anwendungsspezifische Informationen enthalten. Bezogen auf das zuvor gegebene Beispiel der Geburtenstatistik könnten alle Kinder mit dem Namen „Marie" dementsprechenden Lexikoneinträgen zugeordnet werden (siehe dazu auch Abschnitt 5.5).

Die zuvor genannte Bedingung ist *nur* hinreichend, *d.h.* nicht in allen Fällen kann ein mehrdeutiges Wort mit allen zugewiesenen Bedeutungen innerhalb eines semantisch ambiguen Satzes verwendet werden. Zum Beispiel im Satz „Die Schule brennt" ist die Interpretation von Schule als Institution nicht verwendbar. Das Verb „brennt" gibt hier die Auswahlrestriktion vor. In Folge dessen ist die Mehrdeutigkeit in diesem Fall eingeschränkt. Es erfordert jedoch ein Disambiguierungsverfahren, um die möglichen Fehlinterpretationen auszuschließen und die richtige Bedeutung auszuwählen (siehe Teil II).

[33] Dies betrifft den aktuell vorliegenden Satz.

5.2.6. Zusammenfassung

In diesem Teilkapitel werden die unterschiedlichen Arten von Mehrdeutigkeit vorgestellt. Die individuellen Eigenschaften der jeweiligen Ambiguität werden beschrieben. Strukturelle Ambiguität, *d.h.* syntaktische und Skopus-Ambiguität, ist fokussiert auf die mehrdeutige Struktur des Satzes, der je nach vorgenommener Strukturierung unterschiedliche Bedeutungen erhält. Pragmatische Ambiguität beschreibt zum einen auf unbestimmte und daher nicht abgrenzbare Informationen (*z.B.* next Friday) und zum anderen den anaphorischen[34] Gebrauch von Pronomen. Morphologische Ambiguität basiert auf den verschiedenen Bedeutungen eines Wortes je nach zugehöriger Wortklasse. Im Gegensatz zur strukturellen Ambiguität bedarf jede dieser Bedeutungen eines zugehörigen lexikalischen Eintrags, um eine korrekte Bedeutungszuordnung zur Wortform zu ermöglichen. Lexikalische Ambiguität nimmt sich den speziellen Formen der Mehrdeutigkeit in Lexika an. Hier werden reguläre Polysemie (generelle Beziehungen zwischen den verschiedenen Bedeutungen), irreguläre Polysemie (Teilen mindestens eines spezifischen Merkmals) und Homonymie (gleiche Wortform) vorgestellt und unterschieden.

In der vorliegenden Arbeit erfolgt eine Fokussierung auf semantische bzw. referentielle Ambiguität. Diese bezieht sich auf verschiedene Möglichkeiten der Satzinterpretation ausgehend von mehrdeutiger *lexikalischer Information* und ist somit verantwortlich für referentielle Mehrdeutigkeit in Ontologien (vgl. Kapitel 5.5).

5.3. Polysemie in Lexika

Der Autor dieser Arbeit folgt der zusammenfassenden Beschreibung von Dölling [63], der hauptsächlich zwischen vier verschiedenen Methoden von Polysemie in Lexika unterscheidet:

1. Die dem Wort zugeordneten Bedeutungen werden in *separaten* lexikalischen Einträgen aufgeführt.
2. Alle zugeordneten Bedeutungen werden im *gleichen*, *d.h.* in *einem*, lexikalischen Eintrag aufgeführt.

[34] Der Verweis eines Satzteils auf einen anderen.

3. Dem Wort wird nur *eine* Bedeutung aus der Liste der möglichen Bedeutungen hinzugefügt. Die verbleibenden Bedeutungen können aus dieser Bedeutung abgeleitet werden.[35]
4. Dem Wort wird nur eine *unterspezifizierte* (abstrakte) Grundbedeutung hinzugefügt. Aus dieser Bedeutung können *alle* Bedeutungen abgeleitet werden.

Diese Einteilung zeigt auf, dass im Falle der beiden zuerst genannten Methoden *alle Bedeutungsvarianten* im Lexikon *explizit aufgeführt* sind. Es kann je nach kontextueller Situation aus den gegebenen Varianten die korrekte Variante ausgewählt werden. Im Fall der ersten Methode ist die Vorgehensweise *Äquivalent zum Eintrag von Homonymen*, *d.h.* auch bei Homonymie erhält jede Bedeutung einen separaten Eintrag pro Bedeutung (vgl. Abschnitt 5.2.2). Dölling nimmt an, dass bei der zweiten Vorgehensweise eine Ordnung der Einträge, *d.h.* eine Rangfolge angenommen wird. Insofern kann das zuerst genannte Wort als die wahrscheinlichste Bedeutung angenommen werden.

Im Gegensatz zu den ersten beiden Methoden sind bei den letzten beiden *die verschiedenen Bedeutungsvarianten* nur teilweise explizit (Methode 3) und somit vorwiegend *implizit genannt*. Hierbei wird jeweils angenommen, dass die verschiedenen Bedeutungen abgeleitet werden können. In der dritten Methode ist eine Basisbedeutung gegeben, *d.h.* aus der Menge der möglichen Bedeutungen wird diejenige ausgewählt auf die alle anderen aufbauen. Alle weiteren Bedeutungen müssen aufgrund kontextueller Information abgeleitet werden. Die vierte Methode setzt sogar eine vollständige Ableitung aller Methoden voraus. Bis auf eine Bedeutung in Methode 3 und hinsichtlich aller Bedeutungen in Methode 4 kann keine Überprüfung der möglichen Bedeutungen vorgenommen werden. Beide Methoden bieten somit nicht die Möglichkeit einer Auswahl, sondern nur der Ableitung von Bedeutungen. In Methode 3 erfolgt diese Ableitung von der Kernbedeutung anhand des im Text genannten Kontexts. Bei Methode 4 muss die abstrakte Bedeutung durch den Kontext zu einer spezifischen Bedeutung gewandelt werden.

Dölling ordnet nicht-systematische Polysemie der Methode 2 und systematische Polysemie der Methode 4 zu.

[35] Ein Beispiel hierfür ist „Schule“ mit der Zuordnung von „Institution“. Die *weiteren* Bedeutungen, *z.B.* „Gebäude“ *etc.*, können abgeleitet werden.

5.4. Allgemeines Modell

Das allgemeine Modell zeigt eine Generalisierung der zuvor dargestellten lexikalischen Polysemie und Homonymie anhand eines konzeptuellen Kontextmodells. Eine detaillierte Darstellung des konzeptuellen Systems erfolgt in Teil II der Dissertation.

5.4.1. Polysemie

Pethö beschrieb in seinen Aufsatz über Polysemie [174] eine zusammenfassenden Beurteilung von Polysemie und deren Visualisierung, die den Ausgangspunkt für die Grafik 5.5 bildete. Initial gegebenes Merkmal ist ein *Wort*, das im *Lexikon*[36] nachgeschlagen wird. In diesem befindet sich *ein* Eintrag, der *mit einer Menge* Adressen und somit allen dem Wort zugeordneten Bedeutungen verknüpft ist. Diese Adressen sind *„indexed to appropriate parts of the conceptual system“* [174]. Diese Adressen werden als Intensionen[37] bezeichnet. Die Extensionen[38] (Umfang) werden durch das konzeptuelle System bestimmt. Pethö selbst beschreibt Intension und Extension durch *„addresses may all be considered different intensions of the word in question. The extension of the word is not determined directly, but via the conceptual system“* [174].

Wird beispielsweise das Wort „Flügel“ im Sinne des dargestellten Modells im Lexikon nachgeschlagen, dann werden die dementsprechenden Adressen (Intensionen) aufgefunden (*z.B.* „Seitenteil eines Haus“, „Musikinstrument“ *etc.*). Sie sind jeweils in dem gegebenen konzeptuellen System eingebettet, welches deren weitere Beschreibung umfasst, *d.h.* unter anderem wie der Flügel aussieht, welche Objekte mit diesem Flügel in Verbindung stehen (*z.B.* der Pianist, der auf dem Flügel spielt) *etc.* Diese Einbettung des Objekts in die Wissensbasis wird als Extension bezeichnet.

[36] Wie bereits angemerkt ist dieses Lexikon abhängig von der Anwendung.

[37] Pethö gebraucht den den Begriff *Intension nicht* im Sinne der gängigen Definition im Bereich der Logik, sondern bezeichnet damit das „conceptual level“ eines Begriffs. Hier realisiert durch die zugewiesenen Adressen, auch genannt Intensionen.

[38] Auch *Extension* wird *nicht* vollständig im Sinne der klassischen Logik verwendet. Extension bezeichnet das „referential level“ und somit diejenigen Elemente des konzeptuellen Systems, welche mit der Intension in Verbindung gebracht werden können.

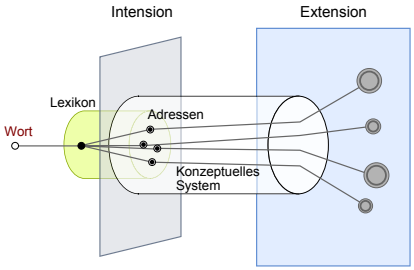


Abbildung 5.5.: Modell für Polysemie (vgl. [174])

Hinsichtlich der lexikalischen Repräsentationsform eines *systematisch* polysemen Wortes verwendet Pethö einen sog. *Adressgenerator*, der die gesuchten Adressen generiert. Dies stimmt mit der oben genannten Sichtweise Döllings für Methode 4 überein, wobei dem Adressgenerator die Aufgabe der Ableitung zukommt. *Nicht-systematische* Polysemie verbindet Pethö mit *verschiedenen lexikalischen Einträgen* und der Methode 1 aus der Sicht von Dölling.[39] Würde man hingegen zu der im Abschnitt zuvor eingeführten dritten Methode tendieren, so würde die primär zugeordnete Adresse als Eingabe für den Adressgenerator dienen.

5.4.2. Homonymie

Der Autor dieser Arbeit verwendet das von Pethö entworfene Modell der Polysemie und überträgt die Begrifflichkeiten von Intension und Extension sowie die damit verbundenen Zusammenhänge auf das Phänomen der Homonymie (siehe Abbildung 5.6). Diese Anpassung erfolgt hinsichtlich der in Kapitel 5.2.2 genannten Eigenschaften von Homonymie. Ein gegebenes Wort wird *unterschiedlichen* Einträgen im Lexikon zugeordnet. Dies entspricht der lexikalischen Darstellung, wie sie in Methode 2 beschrieben wird.[40] Diese voneinander getrennten Einträge werden jeweils auf *eine* eigene Adresse abgebildet. Somit unterscheidet sich das Intensionsmodell von dem zuvor vorgestellten

[39] Da die Methoden 1 und 2 sich überlappen, ist diese Sichtweise nicht als widersprüchlich zu erachten.

[40] Wenngleich die Methoden auf polyseme Einträge ausgerichtet sind, trifft diese Form der lexikalischen Darstellung für die Homonymie zu.

Modelausschnitt der Polysemie. Die Definition einer Extension bleibt jedoch gleich, *d.h.* diese umfasst konzeptuelle Zugehörigkeit sowie die Beziehungen zu anderen Ontologieelementen.

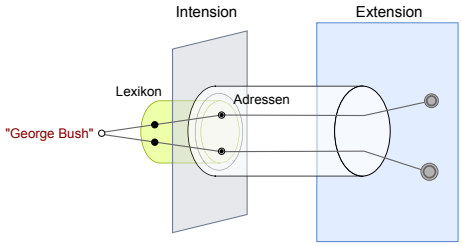


Abbildung 5.6.: Modell für Homonymie (am Beispiel des Ausdrucks „Georg Bush“)

5.4.3. Zusammenhang mit Semiotischen Dreieck

Vergleicht man das zuvor vorgestellte Modell mit dem semiotischen Dreieck, so ist ein direkter Zusammenhang erkennbar (siehe Abbildung 5.7). Mehrdeutigkeit führt zu verschiedenen semiotischen Dreiecken, da bei gleicher Benennung Unterschiede bzgl. des konkreten Gegenstandes sowie des zugehörigen Begriffs vorhanden sind. Es kann je Objekt/Gegenstand ein eigenes semiotisches Dreieck konstruiert werden. Die Benennungen bleiben hierbei identisch. Intension, *d.h.* die entsprechende Adresse, beschreibt das konkrete Objekt bzw. den im semiotischen Dreieck erwähnten Gegenstand. Die Extension wird im semiotischen Dreieck dem Begriff zugeordnet. Dieser enthält die Eigenschaften des Gegenstandes und im Falle von Polysemie Beziehungen zwischen den Extensionen. Daraus ergibt sich der Zusammenhang zwischen Begriff und Extension.

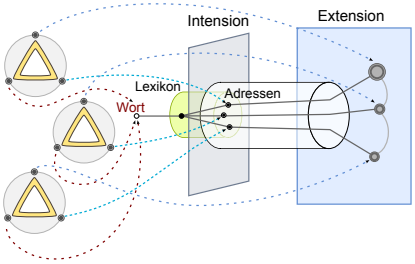


Abbildung 5.7.: Zusammenhang zum semiotischen Dreieck

5.5. Ambiguität in Wissensbasen

In Kapitel 3.2 wird im Rahmen der Definition von Ontologien die Beschreibungssprachen, *z.B.* RDF(S), OWL *etc.*, erwähnt, die als Grundlage für die Repräsentation von Ontologien verwendet werden. Apresjan [14] stellt drei wesentliche Anforderungen an eine Metasprache[41] („semantische Sprache“):

- Die Bezeichnungen müssen monosem[42] sein, *d.h.* Synonymie, Polysemie und Homonymie müssen ausgeschlossen sein.
- Die Zahl der Bezeichnungen muss im Sinne der Ökonomieanforderungen so gering wie möglich sein.
- Die Zahl der Bezeichner muss ausreichen, um alle Bedeutungen zu beschreiben.

Der Grund für das Aufstellen dieser Anforderungen ist die Mehrdeutigkeit der menschlichen Sprache, wie sie in den vorhergehenden Kapiteln behandelt werden. Tim Berners-Lee spricht vom Zwang zur Eindeutigkeit *„Human language thrives when using the same term to mean somewhat different things, but automation does not“* [22]. Dies wird bei der Definition von RDF berücksichtigt, indem Unified Resource Identifiers verwendet werden. Das zeigt auch seine Aussage: *„The triples of RDF form webs of information about related things. Because RDF uses URIs to encode this information in a document, the* ***URIs ensure that concepts***

41 *„Eine Sprache über eine Sprache“* [233]
42 eindeutig

are not just words *in a document but are* ***tied to a unique definition*** *that everyone can find on the Web"* [22].

Der Ursprung dieser Modelle wird jedoch vernachlässigt. Der Mensch als Autor dieser Modelle greift auf die natürliche Sprache bei der Definition von Ontologien zurück. Dies führt dazu, dass natürlich-sprachliche Begriffe ihren Weg in diese Definitionen finden und dort gespeichert werden. Leenheer und Moor beschreiben, dass im Hinblick auf *„the conceptual modelling task [...] the distinction between lexical level (term for a concept) and conceptual level (the concept itself) is often weak or even ignored"* [135]. Dies führt dazu, dass die oben genannte Forderung nach monosemen Bezeichnern zwar auf der Ebene der hierfür ausgewählten Grammatik (URIs) gegeben ist, diese aber hinsichtlich der natürlich-sprachlichen Bezeichner durchaus als problematisch zu erachten ist. Der Autor dieser Arbeit stimmt mit Leenheer et al. [135] überein, die dieses Problem auf zwei wesentliche Merkmale zurückführen:

- *„no matter how expressive ontologies might be, they are all in fact lexical representations of concepts, relationships, and semantic constraints"* [135]
- *„linguistically, there is no bijective mapping between a concept and its lexical representation"* [135]

Insbesondere die zweite Aussage beschreibt das Problem, dass verwendete natürlich-sprachliche Bezeichner *nicht nur einem* Konzept bzw. einem Ontologieelement zugeordnet sein müssen. Das ist in der Mehrdeutigkeit sprachlicher Bezeichner begründet. Es kann ein Term t als Bezeichner von x verschiedenen Ontologieelementen fungieren ($x \geq 1$). Es besteht die Gefahr, dass dieser Term *polysem* oder *homonym* ist. Die Tendenz zur Mehrfachvergabe von gleichen Namen wurde auch durch Furnas et al. [86] nachgewiesen. Diese führten 1987 ein Experiment durch, das nachwies, dass ohne weitere Hilfestellung immer noch in $\tilde{1}5\%$ der Fälle Menschen dem gleichen Objekt auch denselben Namen zuweisen. Falls sie unterstützende Hinweise erhalten oder es sich um Begriffe innerhalb eines Fachgebietes handelt, steigt die Tendenz zur gleichen Benennung.

5.5.1. Sprachbezeichnung in Ontologien

Die Voraussetzungen zur Verwendung (zusätzlicher) natürlich-sprachlicher Bezeichner für Ontologieelemente wurden bereits im RDF(S)-Standard geschaffen. Hier ist der Tag *rdfs:label* definiert, um die Angabe einer *„human-readable version of a resource's name"* [38] zu ermöglichen (vgl. Methode 1 in Abschnitt 5.3).

Außerhalb des Standards RDF/RDFS wurden zusätzlich Standards bzw. Vokabulare entwickelt, die auch im Rahmen von Ontologien Anwendung finden. Ein prominenter und weit verbreiteter Standard ist SKOS - Simple Knowledge Organization System [242]. Hier sind zwei Tags für die Angabe natürlich-sprachlicher Bezeichner vorgesehen, *skos:prefLabel* und *skos:alterLabel*. Diese ermöglichen die Angabe eines primären und verschiedener alternativer Bezeichner, wodurch auch eine implizite Gewichtung der Bezeichner gegeben wird.[43] Es gibt weitere Vokabulare, *z.B.* Foaf [92] *etc.*, welche die Angabe von natürlich-sprachlichen Bezeichnern ermöglichen. Abgesehen davon können eigene Vokabulare entwickelt werden.

Ebenfalls existieren Ontologien, die speziell für die Abbildung lexikalischer Zusammenhänge entwickelt wurden. Beispielsweise ist mit Wordnet [153] ein Wörterbuch der englischen Sprache als Ontologie vorhanden. Existierende Ontologien können mit einem solchen Wörterbuch bzw. Ontologie verknüpft werden. Hierdurch entsteht ein enormer Umfang an sprachlichen Bezeichnern für Ontologieelemente, da Wörterbuch-spezifische Eigenschaften, *z.B.* Synonyme[44], Holonyme[45], Meronyme[46], Hyponyme[47], Hypernyme[48] *etc.*, für diese Bezeichner zur Verfügung stehen. Dies kann in manchen Fällen auch zu Widersprüchen zwischen linguistischer und ontologischer Definition führen, *z.B.* Widersprüche zwischen lexikalischen Holonymen im Lexikon und der Konzepthierarchie in der Domänenontologie. Weiterhin existieren Verfahren (*z.B.* [39]), die auf auf Wordnet aufbauen,

[43] Durch Umkehrprozessierung kann diese Rangfolge erstellt werden (vgl. Methode 2 in Abschnitt 5.3).
[44] lexikalische Begriffe mit ähnlicher bzw. übereinstimmender Bedeutung
[45] Ausdruck, der zum gegebenen Ausdruck im Verhältnis „ist Ganzes von" steht
[46] Ausdruck, der zum gegebenen Ausdruck im Verhältnis „ist Teil von" steht
[47] Ausdruck ist ein Unterbegriff von
[48] Ausdruck ist ein Oberbegriff von

um eigene Lexika davon abzuleiten (siehe Kapitel 14). Wie zuvor angesprochen kann es bei der Verwendung von „externen“ Lexika zu einem Konflikt zwischen domänenbezogenem Modell (Domänenontologie) und allgemeinem Modell (Lexika) kommen, da Grenzen des Domänenmodells durch das Lexikon überschritten werden. Dies ist in der allgemeinen bzw. domänenunabhängigen Sicht von lexikalischen Modellen begründet.

5.5.2. Zusammenhang Ontologie und Modell der Mehrdeutigkeit

Ambiguität tritt innerhalb einer Ontologie durch die Verwendung von mehrdeutigen Begriffen auf. Mehrdeutigkeit ist hierbei nicht im klassischen Sinne, *d.h.* auf der Grundlage des vollständigen Wortschatzes einer gegebenen Sprache, zu verstehen. Diese Mehrdeutigkeit ist beschränkt auf die innerhalb der Ontologie vergebenen Begriffe. Die Vergabe ist auf das verwendete Vokabular zurückzuführen, das den Gebrauch von natürlich-sprachlichen Bezeichnern innerhalb der Ontologie definiert (siehe vorheriger Abschnitt). Eine Mehrdeutigkeit liegt somit vor, falls ein Wort über das Vokabular mehreren Ontologieelementen zugewiesen wird. Somit kann dieses Wort nicht mehr eindeutig ein Element referenzieren und die von Aprejan verlangte Monosemie für Metasprachen ist nicht mehr gegeben. Im Rahmen dieser Arbeit wird für die Darstellung dieser Mehrdeutigkeit ebenfalls das Modell von Pethö verwendet.

Ein gegebenes Wort wird im Lexikon der Ontologie nachgeschlagen. Wird kein externes Lexikon, sondern ein Vokabular (*z.B.* RDF(S)) verwendet, dann können über einen Vergleich der *rdfs:labels* und somit der lexikalischen Bezeichner diejenigen Elemente herausgefunden werden, die diesem Bezeichner zugeordnet sind (vgl. Methode 1 Abschnitt 5.3).[49] Elemente sind hierbei gleich Adressen, die mit URIs gekennzeichnet sind. Diese Adressen beschreiben zugleich die Intensionen für den lexikalischen Begriff. Das konzeptuelle System selbst wird durch die Ontologie repräsentiert. Jedes Element, das aus der Adressermittlung folgt, besitzt eine zugehörige Extension in der Ontologie.

[49] Jedes Element verfügt über einen eigenen Bezeichner und das entspricht der Methode 1 von Dölling bzgl. lexikalischer Struktur. Diese besagt, dass je Element ein separater Eintrag im Lexikon existiert.

Diese Extension beschreibt ein relationales Geflecht der Ontologie, in der das Element das zentrale Merkmal darstellt. Die Ermittlung von möglichen Extensionen ist in Teil II der vorliegenden Arbeit beschrieben.

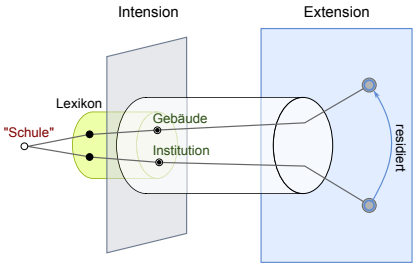


Abbildung 5.8.: Beispiel Polysemie in Ontologien

Homonymie/Polysemie Bei einer Beschreibung mithilfe von *rdfs:label* kann Homonymie und Polysemie nicht auf lexikalischer Ebene getrennt werden. Polysemie kann hier auf Ebene des Lexikons nicht ausgedrückt werden. Im Rahmen dieser Arbeit wird Homonymie und Polysemie in Ontologien wie folgt definiert durch:

Definition 5.6.

- *Von **Polysemie** wird ausgegangen, falls zwischen den Extensionen von Ontologieelementen mit den wortgleichen, zugrunde liegenden Bezeichnern mindestens eine miteinander geteilte Property vorliegt, z.B. das Aussehen für Pferd als Schachfigur und als Tier.*[50]
- *Bei **Homonymie** ist diese Property nicht gegeben (siehe Abbildung 5.9).*

Ein Beispiel für die Definition von *Polysemie* aufgrund des Zusammenhangs verschiedener Extensionen basierend auf Elementen ist durch den Bezeichner „Schule“ in seiner Bedeutung als „Gebäude“, „Gesamtheit der Schüler“ und „Institution“ gegeben, siehe Abbildung 5.8.

[50] Eine Ontologie beschreibt ein individuelles Bild einer Domäne. Daher muss nicht in allen Fällen eine vorliegende natürlich-sprachliche Polysemie (*d.h.* gemäß allgemeiner Lexika) ebenfalls innerhalb der Grenzen der durch die Ontologie repräsentierten Domäne auftreten.

Diese Beziehung kann durch Data- und/oder Object-Properties ausgedrückt werden.

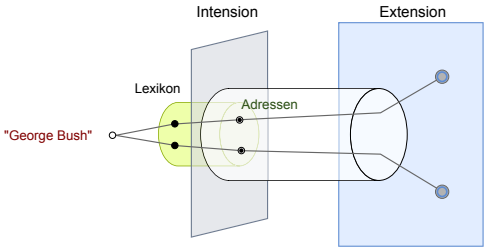


Abbildung 5.9.: Beispiel Homonymie in Ontologien

Im Falle eines homonymen Bezeichners, *z.B.* „Kiefer“ (Baum und Gesichtsknochen) ist anzunehmen, dass diese direkte Relation nicht vorliegt.[51]

Die semantischen Beziehungen, die in einer Ontologie abgebildet werden, sind abhängig von der darzustellenden Domäne und beinhalten nicht die Abbildung aller theoretisch möglichen Sachverhalte. Es kann ohne Beschränkung der Allgemeinheit angenommen werden, dass semantische Beziehungen zwischen möglichen polysemen Entitäten auch in einer Ontologie vorhanden sind. Diese Beziehungen sind möglichst direkt und ohne Zwischenelemente.

Bei Homonymen ist dieser Sachverhalt im Allgemeinen nicht zutreffend. Es kann jedoch nicht ausgeschlossen werden, dass basierend auf der Domäne semantische Beziehungen zwischen den Bedeutungen definiert sind. Solche Beziehungen beinhalten oftmals Zwischenelemente. Beispielsweise im Falle des homonymen Begriffs „Bank“ ist eine mögliche Situation, dass eine Bank (Sitzgelegenheit) auf einem Platz vor einer Bank (Geldinstitut) steht.

In den weiteren Abbildungen hinsichtlich der Intensionen und Extensionen innerhalb von Ontologien ausgehend von gegebenen Begriffen, wird auf die Darstellung des Lexikons verzichtet. Weil hier – wie zuvor erläutert – kein Unterschied zwischen Homonymie und

[51] Dies ist auch in Abbildung 5.9 dargestellt. Die unterschiedlichen Elemente, die „George Bush“ jeweils darstellen, sind nicht miteinander verbunden. Dass eine Verbindung existiert und es sich in diesem Fall um einen polysemen Bezeichner handelt, kann jedoch nicht apriori ausgeschlossen werden.

Polysemie vorhanden ist, da jedes Ontologieelement einen separaten Lexikoneintrag darstellt.[52] Abbildung 5.10 visualisiert dies für die obigen beiden Beispiele.

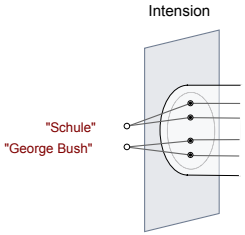


Abbildung 5.10.: Zusammenhang zwischen Begriff und Intension

Zusammenhang Sprachlexikon und Ontologielexikon Abbildung 5.11 zeigt den Zusammenhang zwischen einem Lexikon und einer Ontologie.[53]

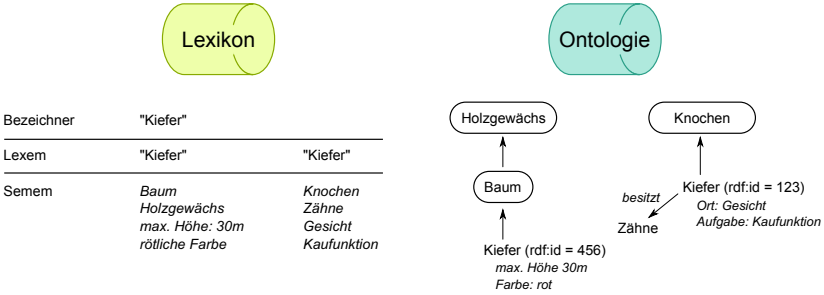


Abbildung 5.11.: Getrennte Extensions für homonyme Bezeichner

Innerhalb eines Lexikons sind die Lexeme von Wörtern definiert, so werden *z.B.* „schreiben“, „schriebst“, „schrieben“ demselben Lexem zugeordnet (siehe [41]). In der Abbildung 5.11 sind für den Bezeichner

[52] Hierbei wird von Domänenontologien ausgegangen, die nicht auf die Repräsentation lexikalischer Zusammenhänge fokussiert sind. Werden RDF Schemata für Lexika, *z.B.* SKOS, verwendet, können Lexeme äquivalent zu den zuvor vorgestellten Lexika-Einträgen erstellt werden.

[53] Die exakte Zuordnung zu einem Lexem und den zugehörigen Semen hängt vom jeweiligen Lexikon ab.

„Kiefer“ zwei Lexeme angegeben. Bezogen auf eine Ontologie werden die mit dem Label[54] „Kiefer“ versehenen Elemente zurückgegeben. Jedem Lexem sind Seme zugordnet, welche die Bedeutung des Lexems beschreiben. Für das erste Lexem sind im Lexikon die Seme „Baum“, „Holzgewächs“, „maximale Höhe 30 Meter“ und „rötliche Farbe“ angegeben. Das zweite Lexem wird beschrieben durch „Knochen“, „Zähne“, „Gesicht“ und „Kaufunktion“. In einem Lexikon befindet sich nur die Aufzählung der Seme und keine weiteren Informationen. Vergleicht man nun die in der Abbildung ebenfalls dargestellte Repräsentation des Sachverhalts in einer Ontologie, so wird deutlich, dass der Bezeichner eines Ontologieobjekts diesem durch eine spezifische Property (*z.B.* `rdfs:label`) zugeordnet ist.[55] Den Ontologieobjekten können Konzepte, Objekt-Properties und Data-Properties zugewiesen werden. Die dadurch zum Ontologieobjekt zusätzlich beschriebenen Informationen entsprechen den zuvor beschriebenen Semen.

Systematische/Nicht-systematische Polysemie *Systematische Polysemie* ist definiert durch eine wiederkehrende relationale Verknüpfung. Zudem trifft für sie das Kriterium der Multilingualität zu. Dies gilt für das zuvor vorgestellte Beispiel „Glas“, das zum einen ein „Gefäß“ und zum anderen einen „Füllstand“ bezeichnet. Dies galt ebenfalls für „Flasche“, „Eimer“ *etc.* Werden diese Begriffe durch Elemente einer Ontologie repräsentiert, zeichnet diese ein wiederkehrendes strukturelles relationales Geflecht aus. Das heißt, dass die Extensionen dieser Bezeichner miteinander in Beziehung stehen bzw. überlappen.

5.5.3. Fazit

Die Verwendung natürlich-sprachlicher Bezeichner führt in Ontologien unweigerlich zu Mehrdeutigkeit. Diese ist unabhängig von der logischen Definition, die durch und in Ontologien gegeben wird. Die

[54] Hier als „Label“ bezeichnet, um den Unterschied zwischen allgemeinem Bezeichner und Bezeichner eines Ontologieelementes hervorzuheben.

[55] Das Ontologieobjekt selbst repräsentiert die zuvor erwähnte Adresse, *z.B.* via `rdf:id`.

logische Definition ist nach wie vor eindeutig und wird von der linguistischen Mehrdeutigkeit nicht beeinflusst. Die Verwendung von Ontologien im Zusammenhang mit natürlich-sprachlichen Bezeichnern, *z.B.* in Verbindung mit Texten, Gesprächen, beim Designprozess *etc.*, und der damit verbundene Vergleich bzw. dem Auffinden von linguistischen Informationen aus Ontologien führt jedoch zum Problem der Wahl des korrekten Bezeichners je nach gegebener Situation. Innerhalb von Ontologien werden linguistische Bezeichner in einer „Art Lexikon" gespeichert. Im Rahmen dieser Arbeit wird dieses Lexikon (*im Fall, dass kein externes Lexikon gegeben ist*) als Menge der natürlich-sprachlichen Bezeichner von Ontologieelementen basierend auf mindestens einer zuvor festgelegten Data-Property definiert. Auf dem Lexikon basierend entsteht ein verbaler Zugriffsindex für Ontologieelemente.[56] Ein verbaler Zugriffsindex ist jedoch nicht eindeutig (vgl. [104]). In Folge dessen ist lexikalische Mehrdeutigkeit, *d.h.* Homonymie und Polysemie, gegeben. Die konkreten Auswirkungen dieser Mehrdeutigkeiten sind oben dargestellt.

5.6. Zusammenfassung

In Kapitel 4 wurden Entitäten und in Abschnitt 4.3 Entitäten im Kontext des Semantic Web vorgestellt. Beispielsweise sind Namensübertragungen ein typischer Grund für Mehrdeutigkeit, *d.h.* im Fall, dass Plätze nach Personen oder Veranstaltungen benannt werden und beide Teil derselben Ontologie sind. In Deutschland taucht dieser Fall beispielsweise bei Schulen auf, die nach Personen benannt werden, *z.B.* „*Erich-Kästner* Realschule". Im alltäglichen Gebrauch wird auf den Zusatz „Schule" oft verzichtet. Ein weiteres Beispiel ist der Bezeichner „*Opel*", der auf den Firmengründer (Adam Opel) oder die Autofirma (Adam Opel AG) verweisen kann.

Insbesondere kommt Metonymie im Bereich von Entitäten vor, *z.B.* „Deutschland" bezeichnet das Land sowie die politische Führung des Landes. Mit dem Begriff „Spiegel" werden die Zeitschrift, der Inhalt der Zeitschrift sowie ein Arbeitgeber referenziert. Auch Abkürzungen führen zu Mehrdeutigkeit, insbesondere im Falle von

[56] Im Falle von Mehrdeutigkeit existieren mehrere Einträge der gleichen Wortform, die unterschiedlichen Elementen zugeordnet sind.

gängigen, *d.h.* verbreiteten Namen. Beispielsweise können dem Ausdruck „B. Müller“ die Namen „Benigna Müller“, „Benjamin Müller“, „Berta Müller“, „Berenike Müller“ *etc.* zugeordnet werden.

Im Rahmen von Ontologien behalten die zuvor genannten Mehrdeutigkeiten und Beispiele ihre Gültigkeit. Mehrdeutigkeiten, wie im Beispiel „Opel“ angegeben, können hierbei als *polysem* (siehe 5.2.1) klassifiziert werden, da alle Bedeutungen dieses Bezeichners miteinander in Beziehung stehen. Mehrdeutigkeiten, wie im Beispiel der Namensübertragung, sind hierbei als *homonym* (siehe 5.2.2) zu klassifizieren.

Im Allgemeinen hängt die mögliche Handhabung mehrdeutiger Begriffe, genauso wie bei den vorherigen Beispielen basierend auf der Verwendung von gängigen Sprachlexika, vom Inhalt des Lexikons ab und von den darin genannten Begriffen[57]. *D.h.* von der Mehrdeutigkeit, die durch das Lexikon zum Ausdruck gebracht wird (siehe 5.5). Bei Ontologien ist die Mehrdeutigkeit vom Inhalt des Lexikons der Ontologie abhängig. Sie ist jedoch nicht eingeschränkt auf die Nennung von Eigennamen, wie in den bisherigen Beispielen gezeigt wird. Zum Beispiel wird mit dem mehrdeutigen Begriff „Flasche“ kein Eigenname verbunden. Dieser kann jedoch beispielsweise als Bezeichner eines Konzeptes verwendet werden. In Abschnitt 4.3 wurde darauf hingewiesen, dass alle Arten von Ressourcen innerhalb von Ontologien als Entitäten zu betrachten sind. In der Konsequenz werden somit auch Mehrdeutigkeiten, die nicht auf Eigennamen zurückzuführen sind, abgedeckt.

[57] Dies gilt für alle Systeme, die mit Mehrdeutigkeit konfrontiert sind.

Teil II.

Bestimmung von Entitätsreferenzen in RDF-Graphen

6. Verfahren zur Referenzbestimmung

Mehrdeutige Wörter sind in der menschlichen Sprache allgegenwärtig und im Sprachsystem weit verbreitet (vgl. [160]).Ohne Verbreitung würde die Anzahl der Wörter unübersichtliche Dimensionen annehmen (vgl. [102]). Nachdem im Kapitel 5 die Auflistung der verschiedenen Arten von Mehrdeutigkeit erfolgt ist, bildet die Auflösung ambiguer Bezeichner den Schwerpunkt dieses Kapitels. Auflösung eines ambiguen Begriffs bedeutet die Bestimmung dessen korrekter[1] Bedeutung im Rahmen der kontextuellen Einbettung des Bezeichners, *d.h.* die Bestimmung der Referenz. Die Komplexität, die diese Aufgabe beinhaltet, wird durch die Aussage von Judith Klavans [126] deutlich: *„Ambiguity" is „the wild child of language interpretation. Whether from the point of view of the philosopher, linguist, psychologist, lexicographer, or computer scientist, ambiguity problems have relentlessly resisted taming"*.

Der Mensch hat im alltäglichen Austausch eine geeignete Vorgehensweise entwickelt, um mehrdeutige Bezeichner aufzulösen und somit der korrekten Bedeutung[2] zuzuordnen. Die theoretische Aufarbeitung dieses Problems enthält noch unbeantwortete Elemente und beschäftigt daher die Wissenschaft bis heute. Eine Näherung hinsichtlich der geeigneten maschinellen Umsetzung dieser Vorgehensweise zur Auflösung der Mehrdeutigkeit in Bezug zu Texten und Wissensbasen wird in Abschnitt 6.1 vorgestellt. Aufbauend auf der dort vorgenommenen Analyse wird in Abschnitt 6.2 das *prinzipielle Vorgehen zur*

[1] Der Ausdruck „korrekt" bedeutet hier, dass aus der Menge der möglichen Bedeutungen, die mit dem gleichen Bezeichner beschrieben werden, diejenige ausgewählt wird, die im Kontext des vorliegenden textuellen Inhalts oder des Gesprächs die zutreffende ist, *d.h.* dessen Referenz.

[2] Ausgehend von den gegebenen Rahmenbedingungen während einer Äußerung, *z.B.* Ort, Geräusche *etc.*, ist eine Auflösung der Mehrdeutigkeit beziehungsweise eine Einschränkung der Menge der vorhandene Bedeutungen möglich (vgl. Abschnitt 6.1).

Disambiguierung eingeführt. Dieser Abschnitt zeigt die grundlegende Herangehensweise zur Auflösung von Mehrdeutigkeiten, *d.h.* zur Referenzbestimmung. Eine Konkretisierung und Fokussierung dieses Prozesses im Hinblick auf semantische Wissensbasen wird in Abschnitt 6.3 vorgenommen. Abschnitt 6.4 erörtert die Einbettung des hier vorgestellten Problems der Ambiguität in das Forschungsfeld Information Retrieval. Abschließend werden in Abschnitt 6.5 die Voraussetzungen beschrieben, die für ein erfolgreiches Monosemierungsverfahren gegeben sein müssen.

6.1. Grundlagen des Monosemierungsprozess (Disambiguierung)

Das Problem bei mehrdeutigen Wörtern ist die Bestimmung der richtigen Referenz in Anbetracht der gegebenen Bedingungen. Die grundsätzliche Voraussetzung für diesen Prozess wird von Pethö beschrieben als: *„Reference depends on the co-operation between a speaker and a hearer and can be considered successfull if the hearer is able to identify what the speaker has in mind“*[3] [174]. In Übereinstimmung mit dieser Aussage bezeichnet Grice [93] die notwendige Abstimmung zwischen Sprecher und Hörer als *co-operative principle*. Dieses spezifiziert Regeln für den Informationsaustausch zwischen Personen. Zwei dieser Regeln sind *(1)* das Vermeiden von Ambiguität innerhalb des sprachlichen Austauschs und *(2)* die Bereitstellung von so viel Information wie möglich. Vor allem aus der zweiten Regel sowie aus der Aussage von Pethö folgt:

Postulat 6.1.[4] *Ambiguität kann anhand von zusätzlich genannten Informationen aufgelöst werden.* ***Voraussetzung ist, dass die genannten Informationen zueinander in Beziehung stehen*** *(vgl. John Searle [209][5]). Nur*

[3] Sprecher und Hörer kann hierbei mit Autor und Leser gleichgesetzt werden. Auch bei textbasierter Kommunikation gilt dieses Prinzip.

[4] Ein Postulat ist definiert „als Ausgangspunkt, als notwendige, unentbehrliche Voraussetzung einer Theorie, eines Gedankenganges dienende Annahme, These, die nicht bewiesen oder nicht beweisbar ist“ (`http://www.duden.de/rechtschreibung/Postulat#Bedeutung3a` [letzter Zugriff am 12.09.2011].).

[5] *„A necessary condition for the successful performance of a definite reference in the utterance of an expression is that either the expression must be an identifying description or the speaker must be able to produce an identifying description [...].“* [209]

durch diese Zusatzinformation wird die Selektion auf diejenige Bedeutung aus der Menge der möglichen Bedeutungen eingeschränkt, die sich durch die zusätzlichen Informationen in den gegebenen Kontext einbettet.

Schippan bestätigt dies mit der Aussage: *„im Kommunikationsakt“* vollzieht sich *„bei der Sprachverwendung ein Monosemierungsprozess“* [202].

Durch die nachfolgendene Aussage konkretisiert Lewandowski die zuvor gegebene Beschreibung des *Kontexts*, die als entscheidend für den Monosemierungsprozess aufgeführt wurde:

Postulat 6.2. *„Das Eindeutigwerden/Eindeutigmachen der grundsätzlich als polysem[6] (mehrdeutig) zu betrachtenden lexikalischen Einheiten/Wörter in der gesellschaftlichen Kommunikation[7] erfolgt, indem durch Kontext und Situation alle nicht relevanten potentiellen Bedeutungsmöglichkeiten eines Wortes ausgeschaltet, alle relevanten unterstützt beziehungsweise aktualisiert werden.“ [138]*

Lewandowski spezifiziert den in der Hypothese 6.1 verwendeten Ausdruck der „zusätzlich genannte(n) Information“[8] als *Kontext* der Äußerung und *Situation*, in der die Äußerung stattfindet. Dieser Kontext ermöglicht es die Bedeutung eines Wortes zu erschließen. Ausgangspunkt sind die möglichen Bedeutungen, die zum einen der Sprecher und zum anderen der Hörer den Worten zuordnen. Die daraus resultierenden überlappenden Bedeutungen, die sowohl Sprecher als auch Hörer den Worten beimessen, können als ein zugrundeliegendes Lexikon betrachtet werden, das die Mehrdeutigkeit von Worten definiert beziehungsweise beinhaltet (siehe auch Abschnitt 5.3). Lewandowski geht davon aus, dass in

6 Wie in Kapitel 5 angegeben, tendieren einige Autoren dazu generelle Mehrdeutigkeit als Polysemie zu bezeichnen. Beschränkt auf dieses Zitat ist *polysem als allgemein mehrdeutig* und *nicht nur* als *zu Polysemie gehörig* zu verstehen.

7 siehe spätere Definition des Situationskontextes

8 Wie angeführt bezeichnet Searle dies als „identifying description“ [209].

der gesellschaftlichen Kommunikation *keine neuen* Bedeutungen auf Grundlage des Sprachsystems (Langue) entstehen.[9]

Schippan [202] klassifiziert den zur Disambiguierung benötigten Kontext in zwei Unterarten[10]:

Lexisch-semantischer Kontext:

Lexikalischer Kontext: Disambiguierung kann anhand übereinstimmender Merkmale in der Grundbedeutung (Flexion) vorgenommen werden. Enthält beispielsweise ein Satz den Bezeichner „lang“ und das Wort „Raum“, so bedeutet dies „räumlich ausgedehnt“. In Verbindung mit „Zeit“ hingegen bedeutet es „zeitlich ausgedehnt“.

Grammatikalischer Kontext: Eine Disambiguierung kann anhand morphologischer, syntaktischer und konstruktiver Bedingungen vorgenommen werden, *z.B.* „sein Wille ist eisern“ (konsequent) und „die eiserne Tür“ (aus dem Material „Eisen“).

Situationskontext:

Raum-Zeit-Situation: Beschreibt die aktuelle Situation, in der die Äußerung stattfindet, *d.h.* die aktuellen Handlungen der Gesprächspartner, die Umgebung, in der das Gespräch stattfindet *etc.*

Gemeinsames Wissen: Um eine Äußerung zu verstehen, muss gemeinsames Wissen zugrunde gelegt werden, das für das Verständnis notwendig ist. Auch der geistige, soziale und kulturelle Hintergrund der Gesprächspartner wird als entscheidend erachtet (vgl. [143]).

---

[9] Neue lexikalisch-semantische Varianten hingegen können innerhalb des Sprachgebrauchs entstehen, *z.B.* „Turnier“ wurde im Mittelalter als Ausdruck für einen Zweikampf zwischen Rittern in Rüstung verwendet. Diese historische Bedeutung gilt im übertragenen Sinne auch bei der heutigen Verwendung des Bezeichners „Tennisturnier“ und somit in seiner heutigen lexikalisch-semantischen Variante. Die Bedeutung als Zweikampf ist gegeben, jedoch wird dieser nicht mehr von Rittern und nicht mehr in Rüstungen ausgetragen.

[10] Die gegebene Klassifizierung wurde auf Grundlage von sprachlichen Äußerungen, *d.h.* Gesprächen, verfasst. Hinsichtlich textueller Quellen ist die „Raum-Zeit-Situation“ nicht gegeben.

Alle Arten von Kontexten nehmen explizit Bezug auf den Satzkontext, *d.h.* die Bedeutung eines Wortes kann anhand lexikalischer sowie in der Wissensbasis vorhandener Merkmale erfasst werden. Die Bedeutung des Satzes selbst bzw. Wörter innerhalb des Satzes ist abhängig vom gesagten/geschriebenen Text, der dem Satz vorausgeht (Anapher) beziehungsweise ihm nachfolgt (Katapher).

Der Prozess der Monosemierung/Disambiguierung führt nicht immer zu einem eindeutigen Ergebnis. Mit oder ohne Absicht des Lesers beziehungsweise des Autors kann es vorkommen, dass dieser Prozess ab einer zuvor nicht absehbaren Stelle nicht fortgeführt werden kann, da zusätzliche Informationen für die Bestimmung einer eindeutigen Zuweisung nicht vorhanden sind. In der Konsequenz ist die Zuweisung von verschiedenen Bedeutungen trotzdem möglich (vgl. [203]). Bei der Kontextklassifizierung wurde zwischen lexisch-semantischem und Situations-Kontext unterschieden. Ersteres bedarf einer Textanalyse (siehe Abschnitt 6.1.1), letzteres zunächst einer Erfassung der äußeren Raum-Zeit-Situation und der Informationszusammenhänge innerhalb der Wissensbasis (siehe Abschnitt 6.1.2). Im Rahmen dieser Arbeit wird die Raum-Zeit-Situation nicht näher untersucht.[11] Zudem wird auch der Aspekt der mündlichen Äußerung, *z.B.* Stimmlage, Lautstärke, Hintergrundgeräusche während des Gespräches, nicht weiter vertieft, sondern die Untersuchung auf textuelle Quellen sowie deren Informationszusammenhänge in der Wissensbasis konzentriert.

6.1.1. Text

Die Untersuchung des lexisch-semantischen Kontextes bedarf einer *lexikalischen* sowie *grammatikalischen* Analyse eines gegebenen Textes. Hierzu werden verschiedene Methoden aus dem Bereich der Computerlinguistik verwendet (siehe auch [117]). Die grammatikalische Analyse ermittelt Sätze, Satzstrukturen, Wortstellungen *etc.* Sie ist der Grundstock für weitergehende Analysen. Für diese Analysen ist die Existenz von Hintergrundwissen jedoch Voraussetzung. Die Analyse hinsichtlich Wortarten (Verb, Substantiv, Pronomen *etc.*) und Wortformen (Vergangenheit, Zukunft *etc.*) *benötigt ein Lexikon*, das diese Informationen beinhaltet. Das ist in Abbildung 6.1 durch die

[11] Für weitere Informationen siehe *z.B.* [143] oder [40].

grünen (lexikalische Analyse) und schwarzen (Strukturanalyse[12]) Bereiche dargestellt. Der blaue Bereich stellt einen semantischen Zusammenhang dar (siehe hierzu den nächsten Abschnitt 6.1.2).

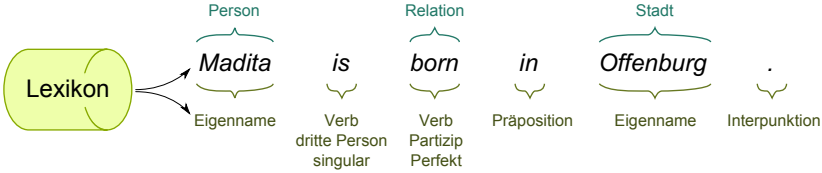


Abbildung 6.1.: Text-basierter Kontext

Aufbauend auf den Wortarten und Wortformen können weitere Analysen erfolgen, *z.B.* der Zusammenhang zwischen Pronomen und Nomen *etc.* Hierfür können grammatikalische Regeln verwendet werden, *z.B.* in Form von Programmlogik. Diese Voranalysen sind Voraussetzung für die Disambiguierung *struktureller, pragmatischer und morphologischer Ambiguität*, da hier auf Faktoren, wie *z.B.* Verb-Substantiv-Unterscheidung (Morphologische Ambiguität), grammatikalisch-basierte Regeln (syntaktische Ambiguität), Feststellen des Pronomens (pragmatische Ambiguität) *etc.*, zurückgegriffen werden muss.

Die Information innerhalb eines Lexikon kann unterschieden werden in: *(a)* Information zu Wortarten und Wortformen und *(b)* Information zur Bedeutung oder Eigenschaften des Wortes (vgl. Abschnitt 5.3). Die in *(b)* genannte Information ist die Grundlage für lexikalische Ambiguität, da hier die polysemen und homonymen Wörter ihre Bedeutungen zugeordnet bekommen. Auch können diese Informationen durch weitere Wissensbasen ergänzt werden, die lexikalischen Bezeichnern Domänen-spezifisches Wissen zuordnen (Wissen zu Instanzen, Relationen und Klassen). Siehe hierzu den folgenden Abschnitt sowie den blauen Bereich in Abbildung 6.1. Für tiefergehende Informationen zur Sprachanalyse hinsichtlich Entitäten siehe Abschnitt 4.2.

[12] Unter Struktur wird hierbei der Aufbau verstanden, *z.B.* Wörter, Sätze, Abschnitte, Kaptiel *etc.* Hier im Beispiel ist die Trennung in einzelne Wörter dargestellt.

6.1.2. Ontologien

Wie bereits im vorherigen Abschnitt im Rahmen der Text-spezifischen Analyse und im Monosemierungsprozess beschrieben (siehe Thesen von Schippan in Abschnitt 6.1), ist der Aspekt des dort vorgestellten Situationskontexts von wesentlicher Bedeutung für die Disambiguierung mehrdeutiger Bezeichner. Der Situationskontexts hebt hervor, dass es gemeinsamen Hintergrundwissens bedarf, um eine Äußerung zu verstehen. Voraussetzung für eine automatische Analyse ist die Repräsentation dieses Wissens in einer geeigneten Struktur, *d.h.* im Kontext dieser Arbeit durch Ontologien.

Ontologien als Hintergrundwissen werden im Bereich der natürlichsprachlichen Entitätserkennung bereits seit längerer Zeit eingesetzt. Chandrasekaran beschrieb bereits 1999: *„Ontologies are useful in Natural Language Understanding [...]“ „[...] as domain knowledge plays a crucial role in disambiguation. A well-designed domain ontology provides the bases for domain knowledge representation“* [43]. Die Verwendung von Domänen-spezifischem Wissen erlaubt es, vorhandene mehrdeutige Bezeichner innerhalb des Textes in der Ontologie zu ermitteln und diese gegebenenfalls anhand der Informationen innerhalb der Ontologie miteinander in Beziehung zu setzen.[13] Auch die Ermittlung der zugeordneten Konzepte (mittels Object-Properties) sowie der Data-Properties, *d.h.* mit dem Objekt in Beziehung stehendes zusätzliches Hintergrundwissen, wird dadurch ermöglicht.

Bezogen auf die linguistische Disambiguierung ist die Aussage von Wolf: *„The sortal proponent can grant all of this, and refine their position by making two claims; (1) the initial inclusion or „baptism“ of a name into the language requires some sortal commitment to disambiguate its referent, and (2) subsequent usage of the name is possible so long as there is some fact about the rule of identity by which the referent should be reidentified, even if some users are not aware of that rule in token instances.“* [240]. Er bezieht sich hierbei auf *sortale Nomen*, *d.h.* Nomen, die zu einer bestimmten Sorte von Objekten zugeordnet werden können. Er beschreibt somit,

[13] Ein mehrdeutiges Wort wird als mehrdeutig klassifiziert, indem diesem in der Ontologie verschiedene Bedeutungen zugewiesen sind, *z.B.* „Schloss“ als Bauwerk bzw. Schließsystem. Wenn in der Definition von Letzterem erwähnt wird, dass es mit einem Schlüssel geöffnet werden kann und Schlüssel ebenfalls im Lexikon definiert ist, ist hierdurch ein Zusammenhang gegeben.

dass bereits bei der initialen Vergabe eines Bezeichners, *z.B.* „Katharina“, eine Zuordnung zu einem Objekt/Klasse, *z.B.* „Person“, erfolgen muss. Er beschreibt weiterhin, dass eine weitere Nutzung dieses Bezeichners (im zuvor definierten Sinne) nur möglich ist, solange es einen Hinweis bzw. eine Regel gibt, die diese Zuordnung ermöglicht. Dieser Hinweis muss nicht in Worten formulierbar sein.

Die Aussage von Wolf beschreibt das Grundvorgehen für die Erstellung, Verwendung und Disambiguierung eines Ontologieelements basierend auf einem linguistischen Bezeichner. Innerhalb der Designphase einer Ontologie werden Elemente erstellt, die bereits bei der Erstellung entsprechenden Klassen zugeordnet werden.[14] Hierbei können diesen Elementen auch linguistische Bezeichner zugeordnet werden.[15] Die Zuordnung dieser Elemente (hier beschränkt auf Instanzen) zu Klassen entspricht der obigen Definition von Michael Wolf.[16]

Kripke [130] spricht im Rahmen eines Beispiels zur Referenzierung der eigentlichen Person, die sich hinter dem Ausdruck *„Jack the Ripper“* [130] verbirgt, die Relevanz von Eigenschaften an, die mit Personen und somit mit Elementen im Allgemeinen verbunden sind: *„It fixes the reference by some contingent marks of the object“* [130]. Auch wenn er sich auf eine *„unique property“* bezieht, die aufgrund ihrer Eindeutigkeit die Identifizierung ermöglicht, wird deutlich, dass vorhandene Informationen über solche Eigenschaften zumindest eine Einschränkung der möglichen Referenzen bewirken können. Abgesehen von der zuvor angeführten Klassifizierung anhand von Konzepten, erfolgt auch die Zuweisung von Eigenschaften zu Ontologieelementen während der Designphase. Hierbei werden Data-Properties und Object-Properties unterschieden. Diese Eigenschaften bilden somit zusätzlich zur Klassenzuordnung die Grundlage für die Objektreferenzierung innerhalb von Ontologien.

Zusammenfassend betrachtet wird der in Abschnitt 6.1 vorgestellte Situationskontext, der zunächst innerhalb des Textes aufgefunden wird,

[14] Dies trifft auch für spätere Aktualisierungen der Ontologie zu.

[15] Dies findet in den meisten Fällen statt, ist jedoch nicht zwingend notwendig.

[16] Nicht für diese Arbeit relevant ist die Tatsache, dass er im weiteren Verlauf des Artikels davon spricht, dass die sortale Zuordnung nicht zuvor bekannt sein muss. Im Rahmen dieser Arbeit werden die sortalen Zuordnungen innerhalb der Ontologie bereits bei der Erstellung von Elementen angegeben und sind somit *a priori* bekannt.

auf eine Ontologie übertragen. Dies geschieht, indem die im Text beschriebenen situativen Zusammenhänge, *d.h.* enthaltene Entitäten, deren Beziehungen *etc.*, versucht werden in der Ontologie aufzufinden und somit den Situationskontext dort zu ermitteln.

Eigenschaften, die mittels Object-Properties in der Ontologie realisiert sind, *zeigen direkte Zusammenhänge zwischen verschiedenen Ontologieelementen auf*. Beispielsweise kann für ein Textfragment „Klaus kennt Andreas“ eine Beziehung `Person` *A* (Klaus) ***`kennt`*** `Person` *B* (Andreas) in der Ontologie existieren. Dies bedeutet, dass der textuelle Situationskontext dem Situationskontext der Ontologie entspricht bzw. in dieser ebenfalls lokalisiert und durch diese somit ausgedrückt werden kann. Eigenschaften, die mittels Data-Properties zugeordnet sind, drücken individuelle Merkmale aus, *z.B.* das Alter einer Person *etc.* Diese Eigenschaften erlauben es nicht, einen unmittelbaren Zusammenhang zwischen verschiedenen Ontologieelementen zu erkennen. Jedoch erlauben diese ebenfalls die Übertragung textuellen Kontexts. Beispielsweise kann der Text „Madita ist 2 Jahre alt“ in einer Ontologie durch „`Person` *C* (Madita) `besitztAlter 2` wiedergefunden werden. Komplexere Rückschlüsse bedürfen einer weitergehenden Analyse, *z.B.* eine Sortierung nach gleichen Attributwerten.[17]

6.2. Prinzipielles Vorgehen zur Disambiguierung

Zur Entwicklung einer grundlegenden Methodik zur Disambiguierung wird zunächst ein Vorbild benötigt, an dem diese ausgerichtet werden kann. Im Rahmen der linguistischen Forschung wurde der Mensch herangezogen. Dieser ordnet zunächst die Wörter, die er liest, den ihm bekannten Bedeutungen zu, um anschließend aus der Summe der Bedeutungen einen Gesamtzusammenhang schlusszufolgern. Diese Vorgehensweise wird in Abschnitt 6.2.1 behandelt und deren linguistische Umsetzung vorgestellt. In Abschnitt 6.2.2 erfolgt im Anschluss die Zuweisung zum allgemein eingeführten

[17] Beispielsweise kann aufgrund des Textes “alle Zweijährigen gehen in den Kindergarten ‘St Anton’“ eine Zuweisung einer Object-Properties „gehen in“ zum Object „St Anton“ für alle Ontologieelemente vorgenommen werden, die vom Typ *Kind* sind und für das Attribut *Alter* den Wert *2* zugewiesen haben.

Modell hinsichtlich der Intension und Extension der konzeptuellen bzw. lexikalischen Ebene. Diesem folgt die Realisierung der Vorgehensweise basierend auf Ontologien.

Dieses Kapitel liefert die konzeptionelle Grundlage des im nächsten Kapitel (Kapitel 7) vorgestellten und vom Autor dieser Arbeit entwickelten Disambiguierungsverfahrens.

6.2.1. Zwei-Ebenen Semantik

Das Modell der *Zwei-Ebenen Semantik* wurde von Manfred Bierwisch 1983 entwickelt [26, 27] und im Laufe der Zeit mehrfach modifiziert. Im Folgenden wird auf der Beschreibung von Pethö [173] aufgebaut, die das Grundmodell modifiziert. Die Vorgehensweise der Zwei-Ebenen Semantik wurde der menschlichen Kognition nachempfunden, *d.h.* den Prozessen des Wahrnehmens und Erfassens von Zusammenhängen. Konkret bedeutet dies, dass bei der Interpretation eines Satzes der Mensch zunächst die Wörter analysiert und die Aussage aus den von ihm erkannten Zusammenhängen zwischen deren Begriffen (vgl. Semiotisches Dreieck 5.1) ableitet. Mehrdeutigkeiten werden hierbei durch die Wahl des korrekten Zusammenhangs ebenfalls aufgelöst.

Wie der Name bereits erahnen lässt, basiert die Analyse von sprachlichen Ausdrücken bei diesem Modell in zwei getrennten Schritten. Diese werden durch zwei verschiedene Ebenen bezeichnet:

a) Lexikalisch-semantische Ebene
   - a_1) Lexemidentifikation
   - a_2) Sembestimmung

b) Konzeptuelle Ebene

Lexikalisch-semantische Ebene Die erste Ebene *(a)* bezieht sich zunächst auf die lexikalischen Informationen, die in einem Satz enthalten sind. Nach der Identifikation der Wörter innerhalb eines Satzes können die zugehörigen Lexeme identifiziert werden. Das jeweilige Lexem wird durch einen Vergleich des vorliegenden Wortes mit den

Wortformen bestimmt, die dem Lexem zugeordnet sind. Diese Identifikation ist notwendig, um im Rahmen der linguistischen Analyse auf den Wörterbucheintrag zugreifen zu können, da diese nach Lexemen aufgeteilt sind. Darauf folgt eine **semantische** Analyse je gegebenem Lexem. Wie in Abbildung 6.2 dargestellt, verfügt jedes Lexem LE_i über die Zuordnung von Bedeutungen, von denen jeder eine Beschreibung von semantischen Eigenschaften (SEM_i) beigefügt ist. Das heißt, dass mittels einer lexikalisch-semantischen Wissensbasis, *z.B.* eines Sprachlexikons, die zugehörigen semantischen Eigenschaften bezüglich jedes Lexems ermittelt werden können.[18] Beispielsweise enthält der Duden für das Lexem „Schule“ die verschiedenen Bedeutungen „Lehranstalt, in der Kindern und Jugendlichen durch planmäßigen Unterricht Wissen und Bildung vermittelt werden“, „Schulgebäude“, „(Zoologie) Schwarm (von Fischen)“ *etc.*[19] Der Bedeutung „Lehranstalt“ sind somit die Domäne „Wissen und Bildung“, die Tätigkeit „unterrichten“ und die Zielgruppe „Kinder“ sowie „Jugendliche“ als weitere semantische Eigenschaften zugewiesen.

Ein Lexikon definiert somit zum einen *(a_1)* Informationen zu Wortarten und Wortformen und zum anderen *(a_2)* Informationen zur Bedeutung und Eigenschaften des Wortes. Zum jetzigen Zeitpunkt wird der Satz A durch die in ihm enthaltenen semantischen Eigenschaften repräsentiert, *d.h.* $(SEM_1 + SEM_2 + ... + SEM_n)_{SEM_A}$.

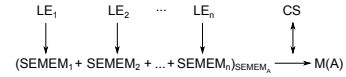


Abbildung 6.2.: Lexikon basierte Satzrepäsentation

In den meisten Fällen ist davon auszugehen, dass die vorhandenen Informationen nicht ausreichen, um den sprachlichen Ausdruck vollständig zu interpretieren beziehungsweise zu disambiguieren. Der Grund hierfür ist, dass die semantischen Eigenschaften der verschiedenen Lexeme keine direkten Übereinstimmungen zeigen und somit kein Zusammenhang geschlussfolgert werden kann. Tritt dieser Fall ein, wird das als eine Unterspezifizierung der Satzbedeutung

[18] Zur Funktion eines Lexikons vgl. Abschnitt 6.1.

[19] Informationen stammen von `http://www.duden.de/rechtschreibung/Schule` [letzter Zugriff am 12.09.2011]

bezeichnet[20] (siehe dazu ebenfalls Abschnitt 6.3). Somit sind weitergehende Informationen notwendig.

Konzeptuelle Ebene Die zweite Ebene, *d.h.* die **konzeptuelle Ebene**, wird nun in Anspruch genommen, um das Defizit an Information aufzulösen. Diese Ebene wird repräsentiert durch das konzeptuelle System CS. Hierbei wird versucht das weitere Vorgehen beim menschlichen Prozess der Disambiguierung nachzubilden. Für den Menschen bilden seine Erfahrungen und sein gespeichertes Wissen die Grundlage für den Prozess. Hierbei handelt es sich um weitergehende Information, *d.h.* semantische Informationen, die über die zuvor erwähnte lexikalische Beschreibung hinausgehen.

Das konzeptuelle System beschreibt eine solche Wissensbasis. Jedes Lexem bzw. die vom Lexikon zugeordneten semantischen Eigenschaften[21] werden in dieser Wissensbasis nachgeschlagen und *es wird in Erfahrung gebracht, welches weitere Wissen damit verknüpft ist*. Somit entsteht ein semantischer Bereich, der umfangreiche Informationen über das ursprüngliche Lexem beinhaltet. Dies erfolgt für alle Lexeme und den diesen zunächst zugeordneten semantischen Eigenschaften. Dieses erweiterte Wissen bildet die Grundlage für die Bestimmung der Satzaussage und somit auch der Auflösung enthaltener Mehrdeutigkeiten, *d.h.* die zuvor vorhandene Unterspezifizierung wird überwunden und die fehlende Information im konzeptuellen System[22] aufgefunden. Damit kann die Interpretation der Satzaussage $M(A)$ vorgenommen werden. Zudem kann weiteres Kontextwissen, *d.h.* innerhalb Anapher oder Katapher, verwendet werden, um die Erweiterung der konzeptuellen Information zu erreichen.

---

20 Der Ausdruck „unterspezifiziert“ wird verwendet, falls *(i)* die Bedeutung nicht aus der Summe der gegebenen semantischen Informationen erschlossen werden kann oder *(ii)* benötigte semantische Informationen im zugehörigen Lexikoneintrag nicht definiert sind. Hervorzuheben ist der Unterschied zur konzeptuellen Ebene (siehe folgender Abschnitt).

21 Ebenfalls können auf das obige Beispiel „Schule“ bezogen zusätzliche Informationen zur Zielgruppe „Jugendliche“ aufgefunden werden. Beispielsweise, dass Menschen zwischen 13 und 19 Jahren als Jugendliche bezeichnet werden.

22 Pethö stellt in seiner Ausarbeitung ein regelbasiertes konzeptionelles System dar. Auf dieses wird im Rahmen dieser Arbeit nicht weiter eingegangen.

6.2.2. Zusammenhang mit dem allgemeinem Modell der Mehrdeutigkeit

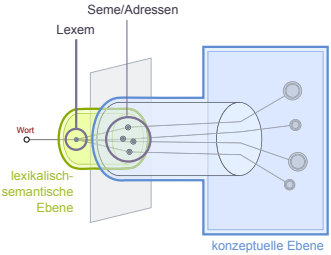


Abbildung 6.3.: Zwei-Ebenen Semantik im allgemeinen Modell (Bild von [174])

Abbildung 6.3 zeigt den entwickelten Zusammenhang mit dem allgemeinen Modell. Zunächst werden die Wörter ihren entsprechenden Einträgen im Lexikon (Intensionen) zugeordnet. Gemäß der oben vorgestellten allgemeinen Vorgehensweise der Zwei-Ebenen Semantik beinhaltet das Lexikon zugleich semantische Eigenschaften zur dort aufgelisteten Bedeutungsvariante. Gemäß Pethö sind Intensionen jedoch nur mögliche Adressen im Lexikon. Die zunächst gegebenen semantischen Eigenschaften SEM_i sind daher bereits ein Teil des konzeptuellen Systems, *d.h.* eine Extension. Für das oben genannte Beispiel „Schule“ wäre somit die Bedeutung „(Zoologie) Schwarm (von Fischen)“ in der Extension für das Lexem gegeben. Eine direkte Bestimmung der Satzaussage ist möglich, falls die initialen Extensionen der verschiedenen Lexeme sich gegenseitig überlappen. Andernfalls besteht eine Unterspezifizierung der Satzbedeutung und somit muss nach Übereinstimmungen in den weiteren semantischen Beziehungen, *d.h.* vergrößerten Extensionen, gesucht werden. Für die Disambiguierung, die Teil der Bestimmung der Satzaussage ist, bedeutet dies:

Postulat 6.3. *Die Disambiguierung durch das konzeptuelle System basiert auf einer teilweisen oder vollständigen Überlappung der Extensionen verschiedener Lexembedeutungen (Intensionen). Dies ist die Grundlage für die Bestimmung relevanter semantischer Eigenschaften und der Rückverfolgung der referenzierten Intension.*

Dies ist zugleich die Grundannahme, worauf der in dieser Dissertation vorgestellte Ansatz aufbaut.

6.3. Ontologie-basierter konzeptueller Disambiguierungsprozess

Bei der allgemeinen Beschreibung der Zwei-Ebenen Semantik (Abschnitt 6.2.1) bleibt die Frage nach der Definition beziehungsweise der Realisierung des konzeptuellen Systems, das für die Monosemierung benötigt wird, offen.[23] Die vom Autor dieser Arbeit vorgeschlagene mögliche Antwort ist der Einsatz einer Ontologie, welche die Definition und gleichzeitige Repräsentation des benötigten Wissens zur Verfügung stellt, *d.h.* die benötigte konzeptuelle Wissensbasis darstellt. Die Ausgangsituation für die Analyse liefert eine zu disambiguierende textuelle Äußerung. Mithilfe des prinzipiellen Disambiguierungsprozesses kann die vorhandene Mehrdeutigkeit aufgelöst, *d.h.* disambiguiert, werden.

Wichtigstes Kriterium für die Disambiguierung mehrdeutiger Begriffe ist die Erfassung von zusätzlichen Informationen, die eine Fokussierung auf einzelne Bedeutungsvarianten ambiguer Bezeichner ermöglicht. Wie oben dargestellt, wird hierbei auf zwei wesentliche Bereiche Bezug genommen:

Textuelle Vorverarbeitung Die Möglichkeiten der textuellen Analyse wurden in Abschnitt 6.1.1 vorgestellt. Hierbei ist hervorzuheben, dass die Erfassung von Bezeichnern (Textausschnitt) und deren Modifikation (Wortform *etc.*) wesentlich für eine Zuordnung zu einem Lexikoneintrag, *d.h.* hier zu dem Bezeichner eines Ontologieelements,

[23] Sowohl Bierwisch [26] als auch Pethö schlagen ein Regelsystem zur Unterscheidung von Bedeutungsvarianten vor. In einer Ontologie können diese Regeln durch Konzeptzuweisungen und die dadurch vorgegebenen Object- und Data-Relationen vorgenommen werden. Pethö führt in seiner Arbeit [173] das Beispiel für „Schule“ ein. Ein schematischer Zusammenhang, dass unter dem Begriff das Gebäude, die Institution, Personengruppe *etc.* gemeint sein können, muss laut Bierwisch und Pethö durch Regeln, die diesen Zusammenhang vorgeben, definiert werden. In einer Ontologie kann dies durch eine Schemadefinition, die diesen Zusammenhang beschreibt, vorgegeben werden. Dies unterscheidet sich jedoch von dem vom Autor dieser Arbeit vorgeschlagenen Disambiguierungsverfahren.

ist. Auch können textuelle Verarbeitungsprozesse zusätzliche Informationen, *z.B.* Data-Properties, Informationen für einen Monosemierungsprozess zur Verfügung stellen (siehe hierzu auch Kapitel 10). Es besteht daher oftmals die Möglichkeit, mit Hilfe einer textuellen Vorverarbeitung zusätzlich zu den Bezeichnern auch Informationen hinsichtlich bestimmter Ontologieelemente zu gewinnen.

Lexikalisch-Semantische Ebene Basierend auf den durch die textuelle Vorverarbeitung bestimmten Bezeichnern wird das Verfahren der Zwei-Ebenen Semantik angewandt. Im ursprünglichen Modell werden anhand der Liste der aufgefundenen Bezeichner die zugehörigen Lexeme bestimmt. Bei der Übertragung des Modells der Zwei-Ebenen-Semantik auf die Ontologie-basierte Disambiguierung wird der Bezug zu Lexemen aufgegeben. Bei Entitätsbezeichnern kann man nicht von einer gemeinsamen Grundform sprechen, wie sie ein Lexem vorgibt (siehe auch 5.5.2). Dies wird am Beispiel der Bezeichner „USA“, „US“, „Amerika“ *etc.* deutlich, die sprachlich keine gemeinsame Grundform aufweisen. In diesem Ansatz wird hierfür die Funktion $f_{Intension}(Bezeichner)$ verwendet. Diese Funktion gibt diejenigen Ontologieelemente der Ontologie zurück, die über einen natürlichsprachlichen Bezeichner verfügen, der mit dem gegebenen Bezeichner übereinstimmt. Die einem Bezeichner a zugeordneten Ontologieelemente sind mögliche *Intensionen* (vgl. Abschnitt 5.4 bzgl. Intensionen) $a_{Int_1}, a_{Int_2}, ..., a_{Int_u}$ für diesen Bezeichner. **Die Monosemierung hat zum Ziel, die im Kontext verwendete Referenz für einen Bezeichner aufzufinden** und aus der Menge der möglichen Ontologieelemente das passende auszuwählen.

Bezogen auf den Instanzgraph einer Ontologie lässt sich die vorhandene Mehrdeutigkeit in verschiedene Kategorien unterteilen.[24]:

- **Instanzen eines Konzepts:** Die Intensionen eines mehrdeutigen Bezeichners beziehen sich auf Instanzen ausschließlich eines Konzepts. *D.h.* die durch einen Bezeichner identifizierten Instanzen sind dem gleichen spezifischen Konzept zugeordnet.
- **Instanzen mehrerer Konzepte:** Die Intensionen eines mehrdeutigen Bezeichners beziehen sich auf Instanzen mehrerer

[24] Diese Art der Unterscheidung wurde vom Autor dieser Arbeit in [257] publiziert.

Konzepte, *d.h.* die durch den Bezeichner identifizierten Instanzen unterscheiden sich teilweise in ihrer Typzuordnung.

- **Außerhalb der Domäne:** Die Intensionen eines vorliegenden mehrdeutigen Bezeichners beziehen sich auf Elemente außerhalb der vorliegenden Ontologie, *d.h.* sind in dieser nicht beschrieben. Eine Ontologie beschreibt die Termini bzw. die Konzepte einer Domäne sowie die Beziehungen dieser Konzepte und beschreibt somit ein domänen-spezifisches Modell. Durch den Bezug zu einer Domäne erfolgt eine Einschränkung auf einen begrenzten Wissensausschnitt im Vergleich zum Allgemeinwissen. Bezeichner können daher auf Dinge verweisen, die nicht innerhalb der Ontologie gespeichert sind beziehungsweise von dieser nicht beschrieben werden.

Konzeptuelle Ebene Das als Referenz identifizierte Ontologieelement ist selbst bereits integraler Bestandteil des durch die Ontologie repräsentierten konzeptuellen Systems. Die Ontologie bietet zudem Informationen über die *Zusammenhänge zwischen Ontologieelementen* und den attributiven und konzeptuellen Eigenschaften. Der konzeptuelle Zusammenhang ist durch eine Überlappung verschiedener Extensionen gegeben, falls eine Verbindung zwischen den einzelnen Ontologieelementen aufgefunden werden kann. Daraus folgt:

Postulat 6.4. *Eine Disambiguierung innerhalb einer Ontologie basiert auf einem ontologischen Zusammenhang. Diesem Zusammenhang gehört mindestens ein Ontologieelement je gegebenen Bezeichner an, das durch diesen Bezeichner benannt werden kann. Dieses Element bildet somit die Referenz des Bezeichners.*

Kann dieser Zusammenhang nicht aufgefunden beziehungsweise kann bei mehreren möglichen Zusammenhängen keine Lösung gegenüber den anderen Lösungen favorisiert werden (falls mehrere Lösungen vorhanden sind), dann war die Disambiguierung nicht erfolgreich. In solchen Fällen kann keine eindeutige Disambiguierung durchgeführt werden, *d.h.* die Identifikation exakt einer Referenz je Bezeichner ist nicht möglich. Oftmals sind die verschiedenen Teilzusammenhänge, die sich aus der Analyse bis zu diesem Zeitpunkt ergaben auch in einem solchen Fall hilfreich, um zumindest eine Einschränkung der Ambiguität zu erreichen.

6.3.1. Beispiel

Abbildung 6.5 zeigt eine mögliche Ontologie-basierte Disambiguierung aufbauend auf dem in Abschnitt 6.1.1 vorgestellten Textbeispiel: „Madita ist geboren in Offenburg“ auf. Zunächst werden die möglichen Bezeichner entsprechend den im Lexikon vorhandenen Bezeichnern aufgefunden beziehungsweise modifiziert (*z.B.* „ist geboren“ zusammengefasst als ein Bezeichner). Dies bedarf einer textuellen Voranalyse. Daraus resultieren die Bezeichner:

- „Madita“
- „ist geboren“
- „in“
- „Offenburg“

Anschließend werden über die Funktion $f_{Intensionen}$ für jeden einzelnen Bezeichner die durch ihn in der Ontologie bezeichneten Elemente erfasst, *d.h.* die Intensionen des Bezeichners. Diese Elemente kennzeichnen zugleich die möglichen Referenzen für den Bezeichner.[25] Das Beispiel ist im allgemeinen Modell in Abbildung 6.4 dargestellt.

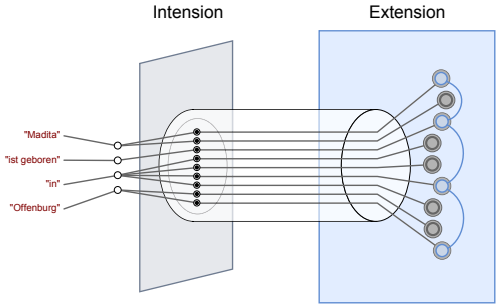


Abbildung 6.4.: Ontologie-basierte Disambiguierung

Da die vorliegende Arbeit sich auf die Graphdarstellung von Ontologien konzentriert und die Analyse anhand des Graphen vollzieht, ist

[25] Nicht in allen Fällen ist ein Element innerhalb der Ontologie für einen gegebenen Bezeichner gegeben. Dieser Umstand wurde in Abschnitt 6.3 beschrieben.

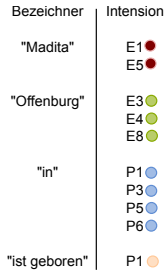
in Abbildung 6.5 das Beispiel anhand eines Ontologiegraphen repräsentiert. Abbildung 6.5(b) zeigt einen Ausschnitt aus dem Ontologiegraph, der einen Zusammenhang der Extensionen der verschiedenen Intensionen repräsentiert.[26] Die Begriffe innerhalb der schwarzen Polygone bezeichnen hierbei Klassen der Ontologie. „E*x*:“ bezeichnet eine Entität, *d.h.* Ontologieinstanz und „P*x*:“ weist auf eine konkrete Property hin. Dieser Graph enthält die Instanzen *E1* und *E2*, die durch $f_{Intensionen}(Madita)$ identifiziert werden konnten. Diese sind in der Abbildung innerhalb der dunkelroten Boxen dargestellt. Die weiteren Zusammenhänge sind in Tabelle 6.5(a) angegebenen. In den folgenden Kapiteln wird näher auf die Auswertung eines solchen Graphen eingegangen.

6.4. Information Retrieval und Disambiguierung

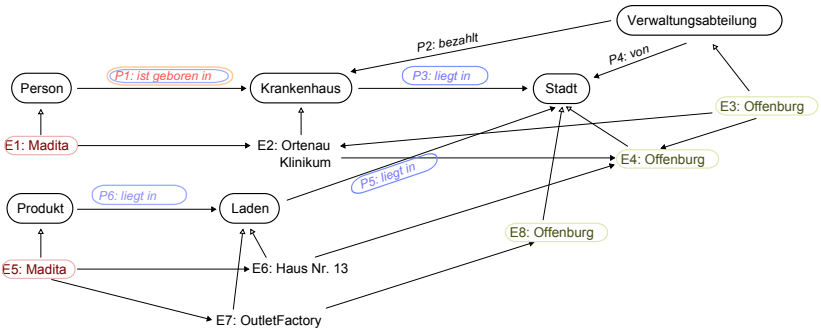
Fabio Crestani et al. beschreiben Information Retrieval (IR) mit Hilfe der Aussage: *„The central problem in IR is the quest to find the set of relevant documents, amongst a large collection, containing the information sought [...] usually expressed by a user with a query. The documents may be objects or items in any medium [...]“* [54]. In dieser kurzen Definition sind die wesentlichen Charakteristika von Information Retrieval benannt. Insbesondere handelt es sich um eine *Suche* nach Elementen mit spezifischen Charakteristika innerhalb einer großen Elementenmenge. Diese Suche hat eine benutzerspezifische Suchanfrage zum Ursprung.

Die Verwendung des Ausdrucks Information Retrieval findet daher meistens im Zusammenhang mit Suchapplikationen statt. Die den Prozess startende Suchanfrage wird in den meisten Fällen von einem menschlichen Benutzer über eine entsprechende Schnittstelle beziehungsweise Benutzeroberfläche eingegeben und somit der Applikation zur Verfügung gestellt. Die Suche bezieht sich auf eine vorhandene natürlich-sprachliche Dokumentenmenge. Die Suchanfrage hat zum

[26] In vielen Fällen enthält dieser Zusammenhang weitere Elemente der Ontologie, deren Bezeichner innerhalb des gegebenen Textes nicht erwähnt wurden.



(a) Bezeichner zu Ontologieelementzuordnung



(b) Ontologie-basierte Disambiguierung

Abbildung 6.5.: Beispiel: Ontologie-basierte Disambiguierung

Ziel diese Menge in einen Teil von relevanten und irrelevanten Dokumenten im Zusammenhang mit der Anfrage aufzuspalten. Diese Dokumente bestehen in der Regel aus natürlich-sprachlichen Texten. Jedoch können ebenfalls nicht natürlich-sprachliche Dokumente sowie Anfragen, die nicht von menschlichen Benutzern stammen, im Information Retrieval Prozess Anwendung finden. Rijsbergen [234] unterscheidet dies ausdrücklich. Er bezeichnet das zuletzt genannte als Data Retrieval und sieht dieses getrennt von Information Retrieval (vgl. Table 1.1 in [234]).

Hervorzuheben ist, dass die Suche auf natürlich-sprachlichem Material eine textuelle Vorverarbeitung (siehe Abschnitt 6.1.1) voraussetzt. Hier findet das Indizieren von initialen Bezeichnern, *d.h.* Worten, statt, die später für die Suche verwendet werden. Diese sind ebenfalls Basis für die ersten Gewichtungsfunktionen, *z.B.* $tf - idf$. Bei mehrdeutigen Bezeichnern innerhalb dieser zugrundeliegenden Dokumente kommt es bereits an dieser Stelle zu falschen Berechnungen, die später das Ergebnis der weiteren Suche beeinflussen. Zernik wies 1991 nach, dass durch eine vorherige Disambiguierung mehrdeutiger Begriffe in der Datenmenge, die von ihm durchgeführte Suche basierend auf einem Suchwort um 50% bessere Resultate erzielte als zuvor [250]. Dies lässt den Schluss zu, dass eine Disambiguierung einen notwendigen Vorverarbeitungsschritt innerhalb des Information Retrieval darstellt, um die darauf aufbauende Dokumentsuche zu verbessern. Insbesondere in Zusammenhang mit „wortarmen" Suchanfragen, *d.h.* mit wenigen Suchworten, führt eine zuvor durchgeführte Monosemierung zu einer Verbesserung der Ergebnisse. Bei der Verwendung von „wortreicheren" Suchanfragen, fügt der Mensch oftmals bereits notwendige Disambiguierungsinformation durch entsprechend zusätzlich genannte Bezeichner in die Suchanfrage mit ein (vgl. Abschnitt 6.1).

Die Disambiguierung selbst kann ebenfalls als Information Retrieval interpretiert werden. Dies erfordert eine Anpassung der Beschreibung des Disambiguierungsverfahrens an die bisher eingeführten Begriffe des Information Retrieval. Der Autor dieser Arbeit stimmt hier mit Sanderson [197] überein, dass ein *Korpus* durch die einem Wort zugewiesenen Bedeutungen dargestellt werden kann. Die innerhalb des natürlich-sprachlichen Satzes enthaltenen *kontextuellen Worte* können als Suchanfrage verwendet werden. Basierend auf dem Suchalgorithmus wird den möglichen Bedeutungen ein Wert der jeweiligen *Wahrscheinlichkeit* zugeordnet, der auch für die *Rangliste des Resultats* verwendet werden kann. Nichtsdestotrotz wird durch diese Beschreibung auch der Unterschied zum klassischen auf die Suche von Dokumenten fokussierten Information Retrieval deutlich.

6.5. Notwendige Voraussetzungen

In diesem Kapitel wird die grundlegende Vorgehensweise zur Disambiguierung beziehungsweise Monosemierung aufgezeigt. Hierbei werden verschiedene Voraussetzungen angesprochen, die für eine vollständige beziehungsweise teilweise Auflösung der Mehrdeutigkeit eines ambiguen Bezeichner benötigt werden und somit zu dessen Referenzbestimmung. Von elementarer Bedeutung ist hierbei das transportierte Kontextwissen, das einem oder mehreren mehrdeutigen Bezeichnern beigefügt ist. Textueller und ontologischer Kontext wird unterschieden, *d.h.* der Kontext eines Bezeichners, der im Text beschrieben wird und den Kontext, den die Ontologie für diesen Bezeichner bereitstellt.

6.5.1. Text

Kontextwissen innerhalb von Texten bedeutet die Erwähnung mehrdeutiger Begriffe in Zusammenhang mit weiteren Begriffen, die es ermöglichen die Referenz des ambiguen Begriffs zu bestimmen. Es ist davon auszugehen, dass der Autor des entsprechenden Textes die Grundannahme vertritt, dass diese Information ausreicht, um eine Monosemierung vorzunehmen.

Die textuelle Analyse muss eine Bestimmung der in einem Satz gegebenen Kontextinformationen ermöglichen (siehe Abschnitt 6.1.1). *Notwendige Voraussetzung* ist somit die Existenz dieser kontextualen Information innerhalb des gegebenen Textes. Insbesondere ist davon auszugehen, dass bei der Disambiguierung von Entitäten die Nennung von mindestens zwei Entitäten erforderlich ist. Jedoch besteht unabhängig davon der in der Zwei-Ebenen Semantik genannte Aspekt der Unterspezifizierung.

6.5.2. Ontologie

„Im Verstehensprozess baut der Rezipient Relationen zwischen den im Satz oder Text genannten Einheiten und Ereignissen auf, indem er auf sein im LZG[27] *gespeichertes Weltwissen zurückgreift."* [208]. Daher geht der Autor beim Erstellen des Textes davon aus, dass die von ihm zur Verfügung gestellte Information es dem Leser ermöglicht, Mehrdeutigkeiten aufzulösen beziehungsweise den Sinn des Satzes in einer Art zu erfassen, wie es die Intention des Autors war. Bei näherer Betrachtung nimmt der Autor an, dass *a)* der Leser über dasselbe Hintergrundwissen verfügt wie er selbst oder *b)* er dem Leser dieses über den Text zur Verfügung stellt, so dass dieser zu den gleichen Schlussfolgerungen gelangen kann. Im Fall *a)* benötigt der Autor nur kontextuelle Information beschränkten Umfangs, um die Mehrdeutigkeit zu überwinden. Im Fall *b)* ist davon auszugehen, dass mehrere zusätzliche kontextuelle Informationen benötigt werden.

Für ein zur Verfügung gestelltes Wissensmodell bedeutet dies, dass das notwendige Instanz- und Konzeptwissen definiert, *d.h.* innerhalb der Ontologie vorhanden, sein muss. Dieses wird benötigt, um die möglichen Bedeutungen der verwendeten Worte zu erfassen. Dieses Kriterium ist unabhängig von der Ausprägung der Mehrdeutigkeit der Begriffe. Die ermittelten Ontologielemente benötigen einen Zusammenhang, der es erlaubt bei mehrdeutigen Bezeichnern das unter den kontextuellen Bedingungen korrekte Ontologieelement, *d.h.* die gesuchte Referenz, zu bestimmen. Somit muss ein Zusammenhang zwischen diesen Elementen bestehen (vgl. Kapitel 7).

[27] LZG $\triangleq$ Langzeitgedächtnis

7. Methodische und technische Verfahrensvoraussetzungen

Der Zugriff auf den Situationskontext stellt einen wesentlichen Faktor für die Referenzbestimmung und somit für die Disambiguierung mehrdeutiger Bezeichner dar (vgl. Kapitel 6). Neben der textuellen Erkennung dieser Bezeichner (Abschnitt 7.3) bildet die Verfügbarkeit und die Analyse von gemeinsamem Wissen die Grundlage für den Prozess der Referenzbestimmung. Dieses Wissen wird im Kontext dieser Arbeit durch Ontologien repräsentiert. Hierbei steht die Darstellung von Ontologien als Graphen im Vordergrund. Abschnitt 7.4 erläutert die Transformation von Ontologien in Graphen, die als Ausgangssituation für die in den Kapiteln 8, 9, 10 und 11 vorgestellten Algorithmen dienen. Voraussetzung für die Disambiguierung ist die Übertragung des im Text beschriebenen Situationskontextes auf die Ontologie. Die ebenfalls in diesem Abschnitt vorgestellte Definition hinsichtlich Steinerbaum-basierten Antwortgraphen findet sich in der Resultatberechnung der Algorithmen wieder. Abschnitt 7.5 erklärt die Vorgehensweise zur Disambiguierung basierend auf der Bestimmung entsprechender Antwortgraphen und setzt diese in Bezug zum in Abschnitt 6.3 beschriebenen, allgemeinen Disambiguierungsprozess. Abschnitt 7.6 erläutert die Technik des „Spreading Activation“, auf der die Vorgehensweise der in den folgenden Kapiteln vorgestellten Algorithmen beruht. Der Zusammenhang zwischen Spreading Activation und Steinerbaum Bestimmung wird in der Vorstellung des Basisansatzes in Kapitel 8 vertieft. Zunächst wird in Abschnitt 7.1 auf die Fokussierung dieser Arbeit auf die Disambiguierung von Entitätsbezeichner eingegangen und im Anschluss daran der vollständig zu durchlaufende Prozess für die Disambiguierung von Referenzen in Abschnitt 7.2 vorgestellt.

7.1. Fokussierung auf die Disambiguierung von Entitätsbezeichnern

Wie bereits in der Einführung (Kapitel 1) beschrieben, konzentriert sich diese Arbeit auf die Disambiguierung von Entitätsbezeichnern (Kapitel 4 bzw. Abschnitt 5.6). Das ist mit der Ausrichtung der Textanalyse auf Eigennamen verbunden (Abschnitt 4.1.1), die innerhalb einer Ontologie als Bezeichner von Konzepten, Instanzen und Properties Anwendung finden (Abschnitt 4.3).

Diese Fokussierung führt dazu, dass eine auf Eigennamen spezialisierte Textanalyse stattfindet. Unabhängig davon sind alle im Rahmen dieser Arbeit vorgestellten Grundlagen sowie das Verfahren zur Referenzbestimmung in Kapitel 6 im Zusammenhang einer allgemeinen Textanalyse ohne Einschränkung gültig. Auch die in den nächsten Kapiteln vorgestellten Verfahrensweisen sind allgemeingültig, jedoch primär auf Entitäten ausgelegt.

7.2. Prozess zur Referenzbestimmung

Der vom Autor dieser Arbeit entworfene Prozess zur Disambiguierung mehrdeutiger Bezeichner ist in Abbildung 7.1 dargestellt. Ausgangspunkt bildet ein zu analysierendes Textdokument, das (ambigue) Bezeichner von Entitäten enthält. Ebenfalls gegeben ist eine Ontologie, die domänenspezifisches Wissen beinhaltet, das im Zusammenhang mit dem Text steht (siehe auch Abschnitt 6.5). Die oberste Zeile innerhalb der Abbildung zeigt die übergeordneten Prozessschritte des Verfahrens. Ausgehend vom gegebenen Text wird der Zusammenhang zwischen textuellen Bezeichnern und Elementen in der Ontologie hergestellt. Jedem Bezeichner wird eine Menge für ihn in Frage kommender Entitäten zugeordnet. Basierend auf der Gesamtheit von möglichen Entitäten erfolgt eine Analyse möglicher Zusammenhänge. Die Monosemierung hat nun zum Ziel, denjenigen Zusammenhang aufzufinden, der die beschriebene Situation innerhalb des

Textes reflektiert und die Anzahl der möglichen Entitäten je Bezeichner *optimal*[1] reduziert.

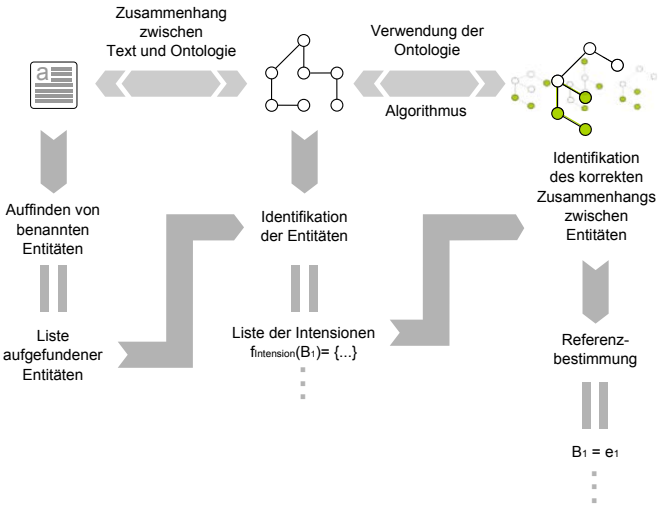


Abbildung 7.1.: Prozess zur Referenzbestimmung
(Bezeichner B_n, Entität e_m)

Die detaillierte Vorgehensweise startet zunächst mit der Analyse des gegebenen Textes. Über geeignete Algorithmen werden aus der Menge von Wörtern eines Textes diejenigen Bezeichner ermittelt, die möglicherweise als Namen von Entitäten fungieren. Dieser Teilprozess ist in Abschnitt 7.3 beschrieben. Ausgehend von der Liste möglicher Entitätsbezeichner werden diejenigen Ontologieelemente ermittelt, die durch diese identifiziert werden können. Als Grundlage dient hierbei die Darstellung der Ontologie in Form eines Graphen (siehe Abschnitt 7.4). Die Ermittlung der in Frage kommenden Ontologieelemente ist anhand der Wörterbuchfunktionen in Abschnitt 7.4.1 beschrieben. Nach der Ermittlung der möglichen Ontologieelemente erfolgt die Monosemierung mehrdeutiger Bezeichner, *d.h.* aus der Menge der Ontologieelemente für einen Bezeichner wird ein Element, *d.h.* Referenz, ausgewählt, das gemäß dem im Text erwähnten

[1] Eine *optimale* Reduktion bezieht sich darauf, dass aus der Menge möglicher Entitäten für einen Bezeichner diejenige Entität gewählt wird, die unter dem Gesichtspunkt des im Satz dargestellten Sachverhalts dessen Referenz darstellt.

Zusammenhang am wahrscheinlichsten ist (siehe Abschnitt 7.5). Die genaue Umsetzung des Prozesses zur Bestimmung dieses Zusammenhangs basiert auf einem Algorithmus unter dem Einsatz von Spreading Activation (Abschnitt 7.6). Die auf dieser theoretischen Grundlage basierende praktische Umsetzung wird in den Kapiteln 8, 9, 10 und 11 vorgestellt.

7.3. Texterkennung

Die Ausgangssituation ist durch die Verwendung von mehrdeutigen Entitäten[2] innerhalb von Dokumenten gegeben. Diese Dokumente können in Form von Nachrichtentexten in Zeitungen, Büchern, Zeitschriften, Online-Dokumenten (*z.B.* Wikipedia) *etc.* vorliegen. Jede dieser verschiedenen Formen besitzt ihre individuellen Eigenheiten, *z.B.* HTML-Verweise zu weiteren Dokumenten *etc.*, die später ebenfalls im Prozess der Weiterverarbeitung berücksichtigt werden können. Abbildung 7.2(c) zeigt einen prozessierten Text, in dem die Entitäten erkannt und annotiert sind.

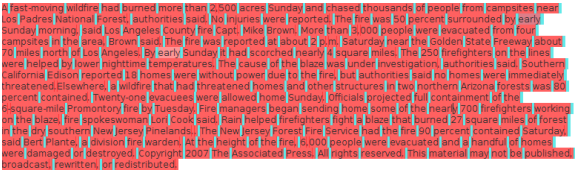
Vorverarbeitung: Zunächst werden verschiedene Schritte zur Vorverarbeitung benötigt, die unabhängig von der nachfolgenden Vorgehensweise zum Auffinden der Entitäten durchgeführt werden müssen. Diese bauen anschließend auf dieser Vorverarbeitung auf.

- **Worterkennung (Tokenization)**: Die einzelnen Wörter im Text werden erkannt (siehe Abbildung 7.2(b) und 7.2(a)). Europäische Sprachen verwenden ein Leerzeichen zur Trennung. Asiatische Sprachen, *z.B.* Japanisch oder Chinesisch, hingegen erfordern einen komplexeren Mechanismus, da die Erkennung der

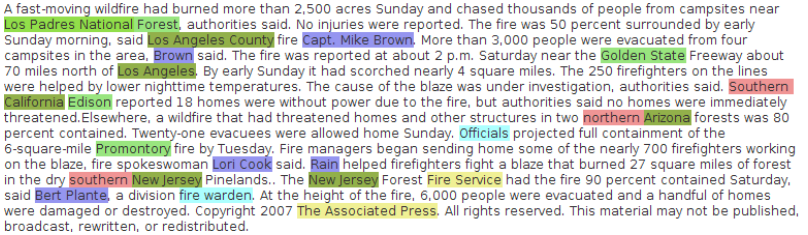
[2] Siehe Kapitel 4 hinsichtlich weitergehenden Informationen zu Entitäten.

[3] General Architecture for Text Engineering, Universität Sheffield (UK) (`http://gate.ac.uk` [letzter Zugriff am 12.09.2011]). Die von Hamish Cunningham et al. entwickelte Plattform ermöglicht die Analyse von Texten, *d.h.* Tokenization (Erkennung von Wörtern (Tokens)), Stemming (Bestimmung der Wortgrundform), Wortarterkennung, Gazetteers (listenbasierte Begriffserkennung) *etc.* Diese verschiedenen Techniken für die Textanalyse können über den Plugin-Mechanismus eingebunden werden (`http://gate.ac.uk/gate/doc/plugins.html` [letzter Zugriff am 12.09.2011]), der zugleich eine einfache Erweiterung durch eigene Programme ermöglicht.

(a) Annotation des Wortes „County“



(b) Einzelne Wörter des Textes erkannt



A fast-moving wildfire had burned more than 2,500 acres Sunday and chased thousands of people from campsites near Los Padres National Forest, authorities said. No injuries were reported. The fire was 50 percent surrounded by early Sunday morning, said Los Angeles County fire Capt. Mike Brown. More than 3,000 people were evacuated from four campsites in the area, Brown said. The fire was reported at about 2 p.m. Saturday near the Golden State Freeway about 70 miles north of Los Angeles. By early Sunday it had scorched nearly 4 square miles. The 250 firefighters on the lines were helped by lower nighttime temperatures. The cause of the blaze was under investigation, authorities said. Southern California Edison reported 18 homes were without power due to the fire, but authorities said no homes were immediately threatened.Elsewhere, a wildfire that had threatened homes and other structures in two northern Arizona forests was 80 percent contained. Twenty-one evacuees were allowed home Sunday. Officials projected full containment of the 6-square-mile Promontory fire by Tuesday. Fire managers began sending home some of the nearly 700 firefighters working on the blaze, fire spokeswoman Lori Cook said. Rain helped firefighters fight a blaze that burned 27 square miles of forest in the dry southern New Jersey Pinelands.. The New Jersey Forest Fire Service had the fire 90 percent contained Saturday, said Bert Plante, a division fire warden. At the height of the fire, 6,000 people were evacuated and a handful of homes were damaged or destroyed. Copyright 2007 The Associated Press. All rights reserved. This material may not be published, broadcast, rewritten, or redistributed.

(c) Text mit annotierten Entitäten

Abbildung 7.2.: Textverarbeitung
(Bilder des Prozessierens mit GATE[3])

einzelnen Wörter hier nicht anhand einfacher Muster vorgenommen werden kann (siehe [101]).

- **Syntaktische Analyse (Part-of-Speech Tagging):** Die einzelnen Wörter werden zu den grammatikalischen Wortklassen, *z.B.* Nomen, Verben, Präpositionen *etc.*, zugeordnet.
- **Morphologische Analyse:** Diese Analyse ermöglicht das Erkennen von Wortteilen, *z.B. Kaffee*kanne. Zudem wird hierbei die Stammform, *z.B. „gehst“* → *„gehen“*, ermittelt und das Erkennen von Präfix und Suffix, *z.B. vor*stellen, durchgeführt.
- **Lexikalische Analyse:** Nachdem die Zeichenketten identifiziert wurden, die Wörter darstellen, können diese in einem Lexikon nachgeschlagen werden. Somit wird ein erster Zusammenhang mit vorhandenem Wissen ermöglicht.

- **Koreferenz Analyse:** Ermittlung des Zusammenhangs zwischen Wörtern, *z.B.* auf welches Nomen bezieht sich ein vorhandenes Pronomen. Beispielsweise bezieht sich das Pronomen *„es“* im Satz *„Das Essen war noch warm und es hat gut geschmeckt“* auf *„Essen“*.

Diese Schritte bauen aufeinander auf und werden zur grundlegenden grammatikalischen Analyse eines gegebenen Textes verwendet. Nachdem die Wörter erkannt wurden, werden Eigennamen anhand von Nomen und anhand der morphologischen Analyse ermittelt. Die lexikalische Analyse ermöglicht die Bestimmung der Intensionen, während die Koreferenzanalyse auf Entitäten innerhalb des Ansatzes der lokalen Kohärenz Anwendung findet, der in Kapitel 10 vorgestellt wird.

Vorgehensweisen: Basierend auf der vorhergehenden Analyse existieren verschiedene Algorithmen zur Erkennung von benannten Entitäten innerhalb von Texten (*z.B.* Wortlisten (Gazetteer) [158], Conditional Random Fields [150] *etc.*). Im Hinblick auf die Vorgehensweise dieser Algorithmen ist es möglich, drei verschiedene Verfahrensweisen hervorzuheben:

- **Listenbasiert:** Verwendung eines oder mehrerer umfangreicher Verzeichnisse (*z.B.* Datenbank), *die alle zu erkennenden Entitäten und deren Namen sowie alternativer Namen enthalten*. Das Nachschlagen eines Wortes in einem solchen Verzeichnis ermöglicht das Auffinden von Entitäten. Es fungiert hierbei als Lexikon. Bezogen auf eine Ontologie kann für einen gegebenen Bezeichner die vorhandene Intension bestimmt werden, welche die entsprechenden Adressen enthält. Die diesen Adressen zugeordneten Ontologieelemente sind eingebettet in die Ontologie und somit kann auf deren Extensionen zugegriffen werden (vgl. Abschnitt 5.5).
- **Regelbasiert:** Erstellen von Regeln, die ein *Erkennen von Entitäten* ermöglichen, die zuvor nicht im lexikalischen Wissen enthalten waren. Zum Beispiel kann bei einer Geographieontologie aus der Erkennung eines *Nomens*, das auf das Wort „County“ folgt oder diesem vorausgeht, geschlossen werden, dass dieses Nomen ein Bezeichner einer Entität darstellt. Diese Entität kann

zudem dem Typ „County“ zugeordnet werden. Ein Beispiel ist in Abbildung 7.2(c) zu sehen. „Los Angeles“ steht hierbei vor „County“ und kann daher als Entität identifiziert werden.
Siehe hierzu auch die Arbeit des Autors dieser Arbeit basierend auf Regeln mit Signalwörtern [259], die ebenfalls in Abschnitt 7.3.1 ausgeführt wird.

- **Lernend:** Verwendung von Methoden des maschinellen Lernens (siehe [155] und [30] für nähere Informationen zu maschinellem Lernen). Diese Methoden benötigen einen Trainingskorpus[4], der bereits annotierte Entitäten enthält. Basierend auf der Struktur des Trainingskorpus, werden die benötigten Kriterien für eine *automatische Erkennung von zuvor unbekannten Entitäten*[5] trainiert (*z.B.* [195]). Zum Einsatz kommen Techniken, wie *z.B.* Hidden Markov Modelle, Maximum Entropie, Conditional Random Fields, Support Vector Machines *etc.* Hier ist der vom Autor dieser Arbeit erstellte Ansatz [257] einzuordnen (siehe Abschnitt 7.3.2).

Die listenbasierte Vorgehensweise ist Ausgangspunkt dieser Arbeit. Wie im vorhergehenden Kapitel 5 bereits erwähnt, liegen die in einer Ontologie definierten Entitäten im Fokus dieser Arbeit. Die Mehrdeutigkeit hat ihren Ursprung in der mehrfachen Verwendung von gleichen Entitätsbezeichnern (siehe Abschnitt 5.5.2). Somit stellt die Menge der Bezeichner innerhalb einer Ontologie die Liste dar, die für die Disambiguierung verwendet wird. Abschnitt 7.4.1 erläutert den Zugriff auf die Bezeichnerliste in RDF(S)-Graphen und somit die graphbezogene listenbasierte Vorgehensweise.

Regelbasierte und maschinell-lernende Vorgehensweisen können zum einen auf eine vorhandene Liste von Entitäten bezogen sein. Hier ermöglicht die Verwendung von Regeln oder Verfahren des maschinellen Lernens eine Relevanzbewertung von möglichen Entitäten. Diese Vorgehensweise kann somit ergänzend zur listenbasierten Verfahrensweise eingesetzt und gegebenenfalls auch in, das im Rahmen der Arbeit vorgestellte Verfahren, eingebettet werden. Zum anderen können

[4] Ein Korpus (lat. corpus (Körper)) bezeichnet eine Sammlung von Texten.

[5] Der im Rahmen dieser Arbeit vorgestellte Ansatz ermöglicht momentan nicht die Erkennung zuvor unbekannter Entitäten. Ein Verfahren, dass sich der Erkennung zuvor unbekannter Entitäten annimmt, kann jedoch eingebettet werden und ist somit Teil möglicher Erweiterungen.

diese Verfahren eingesetzt werden, um unabhängig von oder ergänzend zu Listen mit Entitätsnamen neue Entitäten zu entdecken, die zuvor unbekannt waren.

7.3.1. Exkurs: Regelbasierte Monosemierung

Der Basisansatz der Erkennung von Entitäten verwendet ein zu Beginn des Verfahrens bereitgestelltes Lexikon. Hierbei erfolgt ein direkter Bezeichnervergleich, *d.h.* das Wort im Text wird verglichen mit den Bezeichnern in der Entitätenliste. Im Vergleich zu diesem Ansatz zur Erkennung von Entitäten ermöglicht die Anwendung von Regeln einen selektiveren Zugriff auf die zugrunde liegende Liste. Das Ziel einer regelbasierten Anwendung ist es entweder *1)* eine Erkennung einer Entität ohne direktes Verwenden eines Entitätsbezeichners, *d.h.* das hierfür verwendete Muster enthält keinen Bezeichner einer Entität oder *2)* bei Verwendung des Entitätsbezeichners erfolgt eine Einschränkung der Menge der durch diesen direkt identifizierbaren Entitäten durch die Anwendung einer Regel. Fall *1)* nimmt nicht direkt Bezug zum Lexikon.[6]

2007 entwickelte der Autor dieser Arbeit zusammen mit Raphael Volz und Wolfgang Müller einen Ansatz zur regelbasierten Monosemierung [259]. Dieser umfasst zum einen eine automatische Anreicherung möglicher natürlich-sprachlicher Bezeichner von Entitäten und zum anderen einen Ansatz zur automatischen Einschränkung der Menge der möglichen im listenbasierten Verzeichnis identifizierbaren Entitäten. Für die Erweiterung des natürlich-sprachlichen Vokabulars wurde ein zusätzliches[7] Wörterbuch in die vorhandene Ontologie integriert. Hierzu wurde Wordnet [153, 76] verwendet, das über eine linguistische Ordnungsstruktur durch die enthaltenen Synonym-, Holonym-, Meronym-, Hyponym- und Hypernymrelationen *etc.* verfügt (siehe hierzu auch Abschnitt 5.5.1). Diese können verwendet werden, um Bezeichner einander zuzuordnen und somit mit den lexikalischen Synonymen, die in einem zusätzlichen Wörterbuch definiert wurden, die ursprünglich vorhandenen natürlich-sprachlichen

[6] In Fall *1)* kann die Abarbeitung einer Regel ebenfalls die Selektion von Entitäten aus Mengen beinhalten.

[7] Ein zusätzliches Wörterbuch ist unabhängig vom ontologischen Wörterbuch (siehe 7.4). Für eine Verwendung der im Zusatzlexikon enthaltenen Information muss dieses in das Ontologiewörterbuch integriert werden.

Bezeichner von Ontologieelementen anzureichern (siehe Artikel [259] für weitere Informationen).

Für die Disambiguierung, *d.h.* Einschränkung beziehungsweise Referenzbestimmung, mehrdeutiger Bezeichner sieht der Ansatz ein dreistufiges Verfahren vor. Im ersten Schritt erfolgt die Erkennung möglicher Entitäten basierend auf gegebenen Bezeichnern. Im zweiten Schritt wird eine Vorauswahl möglicher Teilmengen aus der Menge von Entitäten je Bezeichner vorgenommen und im dritten Schritt erfolgt die Erstellung einer Rangliste der in Frage kommenden Entitäten je Bezeichner.

Schritt 1: In der grundlegenden Textanalyse erfolgt die Darstellung eines Dokuments D durch die geordnete[8] Menge der in ihm enthaltenen Terminale $D = (t_1, ..., t_n)$. Die Erkennung von Entitäten erfolgt über einen Bezeichnervergleich. Mit diesem Vergleich werden zum einen Instanzen der Ontologie $i \in I$ mittels $cand(t_i)$[9] und zum anderen Klassen der Ontologie $c \in C$ mittels $con(t_i)$[10] identifiziert.

Schritt 2: Dieser Schritt sieht eine Iteration über die Terme des Dokumentes vor. Hierbei wird jeweils eine Teilmenge von Knoten einer vorgegebenen Größe betrachtet. Im Ansatz wurden zwei Teilschritte vorgestellt:

1. Der erste Teilschritt untersucht jeweils zwei aufeinander folgende Terme (t_i, t_{i+1}) (siehe Abbildung 7.3) und ist spezifisch für die Domäne der Geographie, die im Ansatz verwendet wird[11]. Die verwendete Heuristik sieht vor, dass im Fall, dass t_i als Bezeichner einer Instanz des Konzepts *administrative feature* und t_{i+1} als Bezeichner des Konzepts *administrative region* (unter anderem) in Schritt 1 identifiziert wird et vice versa. Im Folgenden wird die Menge möglicher Entitäten für Bezeichner t_i auf diejenigen eingeschränkt, die in dieser *administrative region* liegen. Zum Beispiel führt die

[8] „Geordnet" bedeutet, dass die Reihenfolge der Wörter (Terminale) im Text beibehalten wird.

[9] $cand : \mathcal{W} \rightarrow 2^I$

[10] $con : \mathcal{Q} \rightarrow 2^C$

[11] Ausgangspunkt ist die Ontologie Geonames `http://www.geonames.org/ontology` [letzter Zugriff am 12.09.2011].

Bearbeitung des Textes „Paris, Frankreich“ mit dieser Analyse zur Auswahl der französischen Hauptstadt aus der Menge der möglichen Entitäten für „Paris“.

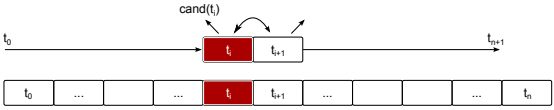


Abbildung 7.3.: Beziehungen zwischen Kandidaten

2. Im zweiten Teilschritt wird eine Menge von elf[12] aufeinander folgenden Termen verwendet $t_{i-5}, ..., t_i, ..., t_{i+5}$ (siehe Abbildung 7.4). Hier wird versucht die Menge der Entitäten der in diesem Abschnitt gefundenen Bezeichner, je Bezeichner auf Entitäten zu begrenzen, die einem ebenfalls in diesem Abschnitt erkannten Konzept als Instanz zugeordnet sind. Bei mehreren Konzepten entscheidet die Distanz zwischen Konzept und Entität. Im Beispiel „Karlsruhe ist eine `city` (Stadt), welche dem Konzept `administrative region` (Bundesland) Baden-Württemberg“ zugeordnet ist, erfolgt somit die Untersuchung von Karlsruhe im Zusammenhang zum Konzept `city` und von Baden-Württemberg zum Konzept `administrative region`. Diese Zuordnung basiert jeweils auf den kürzesten Distanzen. Falls kein Konzept lokalisiert werden wird, kann keine Einschränkung vorgenommen werden.

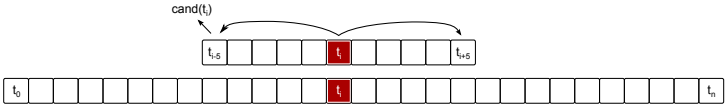


Abbildung 7.4.: Einschränkung möglicher Kandidaten

Schritt 3: In diesem Schritt wird eine Gewichtung der möglichen Instanz je nach Konzeptzugehörigkeit vorgenommen. Der Ansatz verwendet dafür eine Domänen-spezifische Gewichtung von Konzepten. Die Gewichtung eines Konzeptes wird transitiv

[12] Mehrere Tests ergaben diese Kontextfenster als optimale Größe im Bezug zum analysierten Testdatensatz.

an seine Unterklassen weitergegeben und jeweils abgeschwächt. Die Gewichtung wird an die Instanzen weitergegeben. Es werden zudem Domänen-spezifische Heuristiken verwendet. Hier wurden Städte mit höherer Einwohnerzahl gegenüber Städten mit niedriger Einwohnerzahl bevorzugt. Beispielsweise wurde in Folge dessen für „Lancaster“ die Instanz, welche die Stadt Lancaster in Kalifornien (129000 Einwohner) beschreibt gegenüber der Instanz bevorzugt, welche die Stadt Lancaster in Großbritannien (39000 Einwohner) beschreibt.

Testläufe	I	II	III
Precision	40,1%	68,9%	67,9%
Recall	86,7%	86,7%	86,7%
Gewichtungen			
WordnetSenses	0	-20000	-10000
AdministrativeRegion	1000	1000	1000
County	3000	3000	-3000
Hypsography	10	1	10
Locality	1000	500	1000
PopulatedPlace	3000	10000	3000
Road	5	10	5
SpotPlace	10000	2500	1000
Hydrography	10	10	10
Undersea	-10	1	10
Vegetation	10	20	10

Tabelle 7.1.: Evaluation regelbasierte Monosemierung

Bei der Evaluation des Verfahrens (siehe auch [259]) wurden bei der Untersuchung von 255 handanotierten Dokumenten des Reuters21578 Corpus die in Tabelle 7.1 angegebenen Ergebnisse erzielt. Die angegebenen Gewichtungen wurden durch verschiedene zuvor durchgeführte Testläufe bestimmt und für diese Tests vorgegeben.

7.3.2. Exkurs: Regel erlernende Monosemierung

Beim vorgestellten Ansatz wurden die anzuwendenden Regeln vom Autor vorgegeben. Auch für die vorgenommene Auswertung

der Dokumentbereiche[13] und die anschließende Gewichtung wurde die jeweilige Vorgehensweise vorgegeben, *d.h.* die Regeln zuvor explizit definiert. Die Verwendung eines Regel erlernenden Ansatzes sieht dem hingegen das automatische Erstellen einer Regelmenge vor. Diese wird basierend auf einer Menge vorhandener Daten, die exemplarisch die Datengrundlage der später durch die Regeln zu analysierenden Daten darstellen, erlernt.

Im Kontext von Computer-Algorithmen spricht man von maschinell lernenden Algorithmen zur automatischen Regelerstellung. 2009 wurde vom Autor dieser Arbeit im Artikel „Ontology-based Entity Disambiguation with Natural Language Patterns“ [257] ein Ansatz zur Disambiguierung basierend auf maschinellem Lernen vorgestellt.

Im Zentrum des Ansatzes liegt die Problematik, dass ein gegebener Bezeichner auf mehrere Instanzen verweist, die unterschiedlichen Konzepten angehören, *d.h.* verschiedenen Konzepten zugeordnet sind. Diese Art erfordert als primären Schritt zur Auflösung der Ambiguität die Ermittlung des korrekten[14] Konzepts im Satzkontext. Eine weitergehende Analyse ist notwendig, falls durch die Identifikation des Konzepts nicht bereits Eindeutigkeit erreicht wird, *d.h.* der Bezeichner ebenfalls auf mehrere Instanzen dieses Konzeptes verweist. Zur Auflösung der Mehrdeutigkeit folgt der Ansatz der Beschreibung der systematischen Polysemie (Abschnitt 5.2), die davon ausgeht, dass die korrekte Instanz durch eine Regel beschrieben werden kann. Hinsichtlich der zunächst erwähnten Mehrdeutigkeit, *d.h.* die Zuordnung zu verschiedenen Konzepten, können vier Arten von Regeln identifiziert werden:

1. *Regeln, die Instanzen eines Konzeptes identifizieren*, *d.h.* die Reduktion der Intensionen zu Instanzen eines Typs ermöglichen. Beispielsweise die Nennung des Konzeptnamens in direkter textueller Nähe kann als Regel formuliert werden (vgl. Abschnitt 7.3.1).

2. *Regeln, die Instanzen einer Untermenge von Konzepten identifizieren*. Diese finden vor allem Anwendung, falls die Instanzen

[13] Ein Dokumentbereich wurde definiert durch die den jeweiligen Ontologiebezeichner umgebende vorgegebene Anzahl an Termen.

[14] „Korrekt“ bedeutet hier, dass dieses Konzept derjenigen Instanz zugeordnet wird, die im analysierten Textbereich als gültige Referenz des Bezeichners auszuwählen ist.

von Unterkonzepten eines Konzeptes ebenfalls in der Resultatmenge enthalten sein sollen. Beispielsweise kann durch die Nennung von Konzepteigenschaften im Bezeichnerumfeld eine solche Zuordnung ermöglicht werden. Die Nennung von `Inhabitants` würde bei einer entsprechenden Ontologie die Zuordnung zu `Populated Place` und somit zu den Unterkonzepten `City`, `Village` *etc.* ermöglichen.

3. *Regeln, die Object-Properties zwischen Konzepten erkennen*. Zum Beispiel ermöglicht eine Regel „`City` *is located near* `City`“, die konzeptuelle Zuordnung von Instanzen für „Karlsruhe“ und „Rastatt“ zum Konzept `City` ausgehend von einem Textausschnitt „Karlsruhe *is located near* Rastatt“.

4. *Regeln, welche die Erkennung von Data-Properties von Konzepten ermöglichen.* Beispielsweise ermöglicht die Verwendung eines Textmusters „*x people live in* `City`“ die Erkennung der Data-Property `Inhabitants` des Konzepts `City`. Somit kann die Menge an möglichen Referenzen auf Instanzen, die diesem Wert für die gegebene Data-Property besitzen, eingeschränkt werden.

Der Ansatz selbst besteht aus drei Prozessschritten: *1)* Erkennung textueller Bezeichner, die auf Ontologieelemente verweisen (*d.h.* mit deren Namen übereinstimmen), *2)* Suche und Identifikation von Textmustern in der direkten Nachbarschaft eines in Schritt 1 erkannten textuellen Bezeichners und *3)* Gewichtung der Textmuster hinsichtlich ihrer Bedeutung für die konzeptuelle Zuordnung, *d.h.* der Reduktion der Menge möglicher Instanzen zu der Teilmenge der Instanzen für dieses Konzept.

Für den ersten Schritt wird ein sogenannter „Gazetteer“ eingesetzt, der den Vergleich eines im Text identifizierten Wortes mit einer Wortliste der Entitätsbezeichner vornimmt. Hierdurch werden die möglichen Intensionen des Bezeichners bestimmt. Die für den zweiten Schritt benötigten Textmuster werden durch das in diesem Ansatz vorgestellte Lernverfahren automatisch erstellt. Der Algorithmus, der dabei angewendet wird, basiert auf dem Apriori-Algorithmus [3]. Der Apriori-Algorithmus wurde für die automatische Erkennung häufiger Kombinationen in Daten entwickelt. Ein Beispiel hierfür ist die Analyse von Einkäufen von Kunden, *z.B.* einer Einkaufsplattform wie

Amazon[15]. Der Apriori Algorithmus ermöglicht die Analyse von Produkten, die häufig zusammen, *d.h.* kombiniert, gekauft werden. Dazu werden zunächst einzelne Produkte mit einer zuvor vorgegebenen Mindestanzahl an Verkäufen identifiziert. Aus diesen werden anschließend mögliche Kombinationen als Kandidaten für häufige Käufe zweier Artikel erstellt, die dann anschließend überprüft werden. Das Verfahren geht iterativ vor, bis keine Kombination häufiger Artikel mehr nachgewiesen werden kann.

Der Apriori Algorithmus ist in Algorithmus 1 dargestellt. Ausgangspunkt ist der vorliegende Textkorpus. Die Datenbank DB, die im Algorithmus verwendet wird, beinhaltet die Sätze des Korpus. Die in Schritt 1 erkannten Bezeichner von Ontologieelementen werden in diesen Sätzen ersetzt. Dieses Vorgehen basiert auf der Betrachtung der möglichen Typzuordnungen der Instanzen innerhalb der Menge der Ontologieelemente mit diesem Bezeichner. Für jedes Konzept in dieser Menge von Konzepten wird ein Satz in die Datenbank eingefügt. Diese künstliche Erhöhung der Anzahl der Wörter muss bei der nachfolgend beschriebenen Berechnung der Häufigkeit der Wörter berücksichtigt werden. Das Alphabet A besteht aus den im Korpus verwendeten Wörtern. Der Algorithmus beginnt bei allen Wörtern, die aus dem Alphabet A der Menge U, welche die Textmuster enthält, zugefügt werden. Die Menge U enthält zunächst das ganze Aphabet A, da der Algorithmus mit Textmustern der Länge $n := 1$ beginnt. Diese ändert sich jedoch im Verlauf, da die Bedingung für die Verwendung eines Textmusters ist, dass das Muster u in der Datenbank mindestens in der Anzahl s_{min} vorkommt, *d.h.* $sup_{DB} \geq s_{min}$ (sup = support). Ist dies der Fall, so wird dieses Muster bei der Generierung neuer, *d.h.* erweiterter Textmuster verwendet. Im Gegensatz zu der oben dargestellten Kombination von Teilmustern zu neuen Kandidaten, muss im Zusammenhang mit der Textanalyse die im Satz vorhandene Reihenfolge der Wörter eingehalten werden. Das bedeutet, dass bei der Erweiterung eines Musters, wie beispielsweise „the `City`“ nur Muster mit „the“ als letztes Wort innerhalb des Musters oder Muster mit „`City` als erstes Wort“ zur Erweiterung verwendet werden dürfen. Dies hat den Effekt, dass das bisherige Textmuster, das einen korrekten Ausschnitt[16] eines Satzes repräsentiert, nicht auseinander getrennt werden kann.

[15] `http://www.amazon.com` [letzter Zugriff am 12.09.2011]

[16] „Korrekt“ bedeutet hier, dass dieses Satzfragment innerhalb des Korpus vorkommen muss. Dies ist durch die Bedingung der Mindesthäufigkeit s_{min} gegeben. Voraussetzung hierfür ist $s_{min} \geq 1$.

Algorithmus 1 : Erweiterter Apriori

Input : *Alphabet A*, *Database DB*, *MinimumSupport* $s_{\min}$
Output : F
$U := \{\{a\} | a \in A\}$
$n := 1$
while $U \neq \emptyset$ **do**
  compute $sup_{DB}(u), \forall u \in U$
  $F_n := \{u \in U | sup_{DB}(u) \geq s_{\min}\}$
  $U := \{F_n \oplus_{n-1} F_n\} := \{f \oplus_{n-1} g | f, g \in F_n$ and
  $|f \cap g| = (n-1) - overlapping\}$
  $n := n + 1$
end
$F := \{\emptyset\} \cup \bigcup_{n \in \mathbb{N}} F_n$

Die erstellten Muster werden zur Zuordnung von Bezeichnern zu Konzeptinstanzen benötigt. Infolgedessen werden aus der Resultatmenge der häufigen Muster all diejenigen entfernt, die keinen Konzeptnamen beinhalten. Für die verbleibenden Textmuster wird eine Gewichtung vorgenommen, die bei der späteren Analyse Auskunft über die Aussagekraft dieses Musters und somit den Grad der Zuordnung zu dem durch das Muster repräsentierten Konzept gibt. Diese wird anhand der Formel $Rank_{gi \to c_i} := (\frac{f_{c_i}}{f_d}) * l_p$ bestimmt. Hierbei gibt die Variable f_{c_i} an, wie oft das Muster in der Datenbank im Zusammenhang mit dem Konzept c_i vorkommt, f_d ist die generelle Häufigkeit des Musters in der Datenbank und l_p bezeichnet die Länge des Musters, *d.h.* die Anzahl der Terme aus denen es besteht. Somit steigt die Gewichtung durch die Länge und und der Häufigkeit des Musters im Zusammenhang mit dem Konzept c_i.

Zusätzlich zum vorgestellten Ansatz ist auch die vom Autor betreute Diplomarbeit von Jiayi Shen [213] zu erwähnen, die sich mit der Verwendung maschineller Lernalgorithmen, insbesondere mit Support-Vector-Machines, beschäftigt. Der Ansatz ist hierbei, von der textuellen Umgebung eines Bezeichners auf das zugehörige Konzept der durch ihn bezeichneten Instanzen zu schließen. Zunächst erfolgt eine Textanalyse und eine Vektorgenerierung. Anhand eines Trainingsdatensatzes wird das Lernmodell entsprechend aufgebaut und auf neue Datensätze angewendet. Eine Evaluation des Verfahrens ist in der zuvor erwähnten Arbeit aufgeführt.

Das vorgestellte Verfahren ermöglicht die Zuordnung eines ambiguen Entitätsbezeichners zu Instanzen des durch die Analyse von Textmustern bestimmten Konzepts. Durch diese Zuordnung reduziert sich die Ambiguität des Bezeichners. Dies kann zur Bestimmung der Maße des im Abschnitt 7.3.1 vorgestellten Verfahrens bzw. aller Verfahren mit ähnlicher Zielgebung angewandt werden. Die identifizierten Textmuster können als Ausgangspunkt für weitere Analysen verwendet werden, *z.B.* eine kategorische Einteilung von Texten (*z.B.* in Politik, Geographie *etc.*).

7.4. Graphrepräsentation von RDF

In Abschnitt 6.1 wurden die Grundlagen des Monosemierungsprozesses vorgestellt. Neben dem lexikalisch-semantischen Kontext wurde auch der Situationskontext eingeführt. Das gemeinsame Hintergrundwissen, das Autor und Leser teilen, ist ein wesentlicher Faktor zur Disambiguierung mehrdeutiger Bezeichner. Erst dieses gemeinsame Hintergrundwissen ermöglicht eine Einordnung von Informationen. Die Art und Weise dieser Einordnung der durch den Autor gegebenen Information in dieses Hintergrundwissen und insbesondere die direkt mit ihm verknüpften Informationen stellen wertvolles Wissen für die Referenzauflösung dar.

In Abschnitt 6.2 wurden die Grundlagen für den konzeptuellen Disambiguierungsprozess beschrieben. Dargestellt wurden zum einen die konzeptuelle Repräsentation des Hintergrundwissens sowie zum anderen das vorhandene Wissen über die enthaltenen Entitäten. Im Rahmen dieser Arbeit werden Ontologien, die die Möglichkeit bieten dieses Wissen zu repräsentieren, verwendet. In Abschnitt 3.2.1 wurde im Zusammenhang mit der Einführung von Ontologieformaten ebenfalls RDF(S) vorgestellt. Die durch RDF(S) beschriebenen Daten werden durch RDF-Tripel repräsentiert. Ein „Tripel“ entspricht der Struktur: Subjekt, Prädikat und Objekt, *z.B.* $< urn{:}123, rdfs{:}label, 'Madita' >$ (siehe Abbildung 7.5).

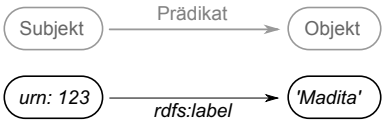


Abbildung 7.5.: RDF-Tripel

7.4.1. RDFS- und RDF-Graph

Der W3C[17] Standard beschreibt die Möglichkeit, mithilfe dieser Tripel-Information einen *„directed, labelled graph“* zu erstellen *„where edges represent a named link between two resources, represented as graph nodes“* [128]. Eine RDF-Ontologie kann somit in ihrer Gesamtheit als Graph dargestellt werden ohne eine Transformation des darin enthaltenen Wissens vorzunehmen, *d.h.* die Darstellung als Graph ist mit keinem Informationsverlust verbunden. Nachfolgend ist die Graph-Definition für RDFS-Ontologien gegeben (siehe auch Erklärungen in [255, 256] sowie [108, 247, 128, 38]).

Definition 7.1 (RDF-Graph)**.** *Eine Menge von RDF-Tripeln wird als RDF-Graph bezeichnet. Die einzelnen Elemente eines Tripels besitzen URI's, Literale oder Blank-nodes[18] als Werte.* `Subjekt` *ist hierbei die beschriebene Ressource,* `Prädikat` *die Property und somit die Art der Verbindung und* `Objekt` *der Wert der Property. Ein* `Subjekt` *kann hierbei nicht durch ein Literal dargestellt werden und ein* `Prädikat` *muss durch eine URI gekennzeichnet sein.*

Alle Werte in den Tripeln eines RDF-Graph $\mathcal{R}$, die keine Blank-nodes darstellen können mit der Funktion $vocabulary(\mathcal{R})$ und die Größe eines RDF-Graphen in Tripeln kann durch $size(\mathcal{R}) = |\mathcal{R}|$ abgefragt werden. Mit den Funktionen $subjekt(\mathcal{R}), prädikat(\mathcal{R}), objekt(\mathcal{R})$ kann jeweils die Menge der Werte, die sich an den jeweiligen Tripel Positionen befindet, ausgegeben werden. Literale werden nicht als Knoten im Graph dargestellt, sondern als Eigenschaften den jeweiligen Subjektknoten zugewiesen.

---

17 W3C = **W**orld **W**ide **W**eb **C**onsortium

18 Sogenannte „Blank nodes“ ermöglichen die Erzeugung mehrstelliger Relationen. Sie können weder durch ein Literal noch durch eine URI identifiziert werden. Zum Beispiel kann mittels `<bsp:Joachim bsp:hatTelefonNr _:b>`, `<_:b bsp:number '01'>` und `<_:b bsp:number '02'>` die Zuweisung zweier Telefonnummern über diese mehrstellige Relation erzeugt werden.

Im Folgenden wird eine angepasste Version[19] der Definition von Xiao et al. [247] verwendet, die eine Trennung des Gesamt-RDF Graphen in den konzeptuellen Part, *d.h.* RDF-Schema Graph, und die Instanz-Beschreibungen, *d.h.* RDF-Instanz Graph, ermöglicht.

Definition 7.2 (RDF-Schema Graph)**.** *Ein RDF-Schema Graph $\mathcal{R}$ über dem Alphabet $\mathcal{A}_\mathcal{R}$ ist ein gerichteter und bezeichneter Graph $\mathcal{R} = (V, E, \lambda)$. V bezeichnet hierbei die Menge von Knoten und E die Menge der Kanten. „Gerichtet" bedeutet, dass jede Kante $e \in E$ genau einen Anfangsknoten (oder Startknoten) $v_i \in V$ mit genau einem Endknoten (oder Zielknoten) $v_j \in V$ verbindet und nicht-symmetrisch ($E \neq E^{-1}$) ist. Die Menge an Knoten V teilt sich in Klassen C, Properties P und Datentypen L auf. $E = \{(v_i, v_j) | v_i, v_j \in V\}$ definiert eine Menge von bezeichneten Kanten, während die Bezeichnungsfunktion durch $\lambda : V \cup E \mapsto \mathcal{A}_\mathcal{R}$ gegeben ist. Es gilt:*

- *$\forall v \in P$ gilt $domain(v) \in C, range(v) \in C \cup L$ und $\lambda((v, domain(v))) =$ „rdfs:domain" und $\lambda((v, range(v))) =$ „rdfs:range"*
- *$\forall e = (v_i, v_j) \in E$ gilt im Fall von v_i und $v_j \in C$, dass $\lambda(v_i, v_j) =$ „rdfs:subClassOf"" und im Fall v_i und $v_j \in P$, dass $\lambda(v_i, v_j) =$ „rdfs:subPropertyOf""*

Bezogen auf ein Tripel $< v_1\ v_p\ v_2 >$ bezeichnet $domain(v_p)$ den Ursprungsknoten der Kante, *d.h.* v_1, und $range(v_p)$ den Zielknoten der Kante, *d.h.* v_2. Das Prädikat „rdfs:subClassOf" bezeichnen eine Unterklasse gemäß der Klassen-Hierarchie und das Prädikat „rdfs:subPropertyOf" beschreibt eine Spezialisierung zwischen zwei Properties, *z.B.*
$<$ example:hasToDoWith, rdfs:subPropertyOf, example:isTaughtBy $>$
(für eine vollständige Erklärung des RDF- beziehungsweise RDF-Schema Vokabulars siehe [38, 128]).

[19] Xiao et al. entwarfen diese Version für das Mapping von XML-Daten auf RDF-Daten beziehungsweise die Transformation von XML-Daten zu RDF-Daten. Insbesondere durch die strukturell geordnete Darstellung von komplexen Objekten in XML ist für den Zusammenhang zwischen hierachisch übergeordnetem Element und untergeordnetem Elementen beim Mapping keine spezifische Relation vorhanden. Diese wird von Xiao mit „rdfx:contained" angenommen. Ein Beispiel hierfür ist das komplexe XML-Element `<Adresse><Straße/></Adresse>`. Hier würde im RDF-Mapping das Tripel `<Adresse rdfx:contained Straße>` generiert. Zuletzt Genanntes ist jedoch für die vorliegende Arbeit *nicht von Relevanz*.

Neben dem RDF-Schema-Graph enthält die gegebene RDF-Ontologie die instanzspezifische Beschreibung durch den RDF-Instanz Graph. Die Datentypen sind in dieser Definition auf Literale eingeschränkt.

Definition 7.3 (RDF-Instanz Graph)**.** *Aufsetzend auf einem RDF-Schema Graph $\mathcal{S} = (V_\mathcal{S}, E_\mathcal{S}, \lambda_\mathcal{S})$, mit $V_\mathcal{S} = C \cup P$, kann ein RDF-Instanz Graph $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G}, \tau, \lambda_\mathcal{G})$ durch die Beschreibung klassenspezifischer Objekte erzeugt werden. Γ beschreibt eine endliche Menge an Konstanten und U eine endliche Menge an URI's. Weiterhin bezeichnet $V_\mathcal{G}$ eine Menge von Knoten und $E_\mathcal{G}$ eine Menge von Kanten. Die Bezeichnungsfunktion $\lambda_\mathcal{G}$ ist definiert durch $\lambda_\mathcal{G} : V_\mathcal{G} \cup E_\mathcal{G} \mapsto \mathcal{A}_\mathcal{G} \cup U \cup \Gamma$*[20] *und τ definiert eine Typisierungsfunktion $\tau : V_\mathcal{G} \cup E_\mathcal{G} \mapsto V_S \cup \{$„rdf:Property“$\} \cup \{$„rdfs:literal“$\}$.*[21]

$\forall e = (v_i, v_j) \in E_\mathcal{G}$ gilt:

1. *Falls $\tau(e) =$ „rdf:Property“, dann $\lambda_\mathcal{G}(e) =$ „rdfs:subClassOf“, $\lambda_\mathcal{G}(v_i)$ und $\lambda_\mathcal{G}(v_j) \in U$, $\tau(v_i)$ und $\tau(v_j) \in C$ und $(\tau(v_i), \tau(v_j)) \in E_\mathcal{S}$*
2. *Falls $\tau(e) \in P$, dann $\lambda_\mathcal{G}(e) = \lambda_\mathcal{S}(\tau(e)), \lambda_\mathcal{G}(v_i) \in U, \tau(v_i) \in C, \lambda_\mathcal{S}((\tau(e)), (\tau(v_i))) =$ „rdfs:domain“, $\lambda_\mathcal{S}((\tau(e)), (\tau(v_j))) =$ „rdfs:range“ und*
   - *$\lambda_\mathcal{G}(v_j) \in U$, falls $\tau(v_j) \in C$*
   - *$\lambda_\mathcal{G}(v_j) \in \Gamma$, falls $\tau(v_j) =$ „rdfs:literal“*

Übergreifend, *d.h.* für Schema- und Instanzgraph, ist eine Bindung von Ontologieelementen zu natürlich-sprachlichen Bezeichnern vorhanden. Diese existiert zusätzlich zu ihrer Kennzeichnung durch URI's. Basierend auf diesen natürlich-sprachlichen Bezeichnern existiert daher ein ontologiespezifisches Lexikon.

[20] Beispielsweise stützt sich die Bezeichnungsfunktion $\lambda_\mathcal{G}(Karlsruhe123) =$ 'Karlsruhe' auf die vorhandene Bezeichnerzuordnung durch das Tripel <*Karlsruhe123 rdfs:label 'Karlsruhe'*>, *d.h.* hier $\lambda_\mathcal{G}(Karlsruhe123) \in \Gamma$.

[21] Beispielsweise stützt sich die Typfunktion $\tau(Karlsruhe123)$ auf die vorhandene Typzuordnung durch das Tripel <*Karlsruhe123 rdf:tpeOf City*>.

Definition 7.4 (Wörterbuch)**.** *Ein Wörterbuch $\mathcal{D}$ einer Ontologie enthält eine endliche Menge an Wörtern, die als Bezeichner Ontologieelementen zugeordnet sind. Die Relation(en), die zur Zuordnung verwendet wird, ist abhängig vom Benutzer und muss zuvor gegeben sein (z.B. „rdfs:label“[22]). Es existiert eine Funktion $l : V \rightarrow 2^{\mathcal{D}}$, die Bezeichner eines gegebenen Ontologieelementes auflistet. Hierbei handelt es sich um eine Einschränkung der zu berücksichtigenden Properties, d.h. um eine Teilmenge von λ_S bzw. λ_G. Ebenfalls können mit $p : \mathcal{D} \rightarrow 2^V$ die Elemente zu einem gegebenen Bezeichner aufgelistet werden. Dies entspricht der in Abschnitt 6.2.2 angesprochenen Identifikation der Intension mittels $f_{Intension}$.*

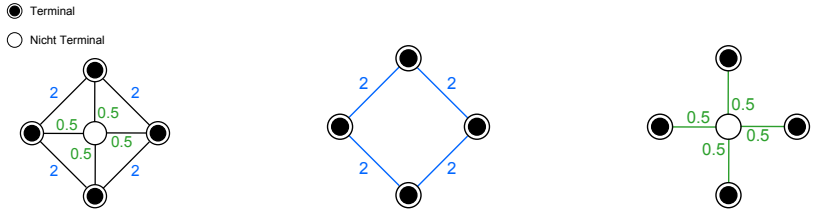
Unabhängig von der Teilung von Schema und Instanz Graphen gilt:

Definition 7.5 (Weitere Graph Funktionen)**.**

- *Zwei Kanten sind* nachfolgend*, falls der Endknoten der einen Kante der Startknoten der anderen Kante darstellt. Im Beispiel der Kanten $e_1 = (v_i, v_j)$ und $e_2 = (v_j, v_l)$ gilt dieser Umstand für v_j.*
- *Ein Pfad (*pfad*) beschreibt eine Sequenz von Kanten $e_1, e_2, ..., e_i, ..., e_n$. Für jede dieser Kanten gilt, dass e_i auf e_{i-1}* nachfolgend *ist.*
- *Zwei Knoten x und y sind* zusammenhängend*, falls ein* pfad$(e_1, ..., e_n)$ *zwischen ihnen existiert mit $x \in e_1$ und $y \in e_n$.*
- *Die Länge (*länge*) des Pfades entspricht der Anzahl seiner Kanten.*

OWL-Full [204] baut auf der Semantik von RDF auf. Demzufolge kann eine Ontologie, die in OWL-Full vorliegt, als Graph repräsentiert werden. Somit gelten die oben angegebenen Definitionen ebenfalls für dieses Ontologieformat. OWL-DL [33] Ontologien lassen sich ebenfalls in RDF-Ontologien übertragen und sind somit ebenfalls als Graph darstellbar. Dies bedeutet, dass ebenfalls OWL-Full-Ontologien sowie OWL-DL-Ontologien als Graph dargestellt und somit vom im Rahmen dieser Arbeit präsentierten Verfahren als Hintergrundwissen verwendet werden können.

[22] *„rdfs:label is an instance of rdf:Property that may be used to provide a human-readable version of a resource's name.“* `http://www.w3.org/TR/rdf-schema/#ch_label` [letzter Zugriff am 12.09.2011]



(a) Ausgangsgraph

(b) Minimaler Spannbaum ($c(e) = 8$)

(c) Steinerbaum ($c(e) = 2$)

Abbildung 7.6.: Spannbäume: Minimaler Spannbaum vs. Steinerbaum

7.4.2. Bestimmung von Teilgraphen

Die Darstellung einer Wissensbasis als Graphmodel $\mathcal{G} = (V, E)$ bietet verschiedene Möglichkeiten des Zugriffs. Zuvor wurde gezeigt, dass es möglich ist, Pfade zwischen Knoten zu bestimmen. Hierbei können Knoten verschiedene Entitäten $v \in V$ darstellen, die über Kanten $e \in E$ miteinander verbunden sind. Mithilfe der Pfade lassen sich semantische Beziehungen zwischen zwei gegebenen Entitäten entdecken und analysieren. In dem Fall, dass zwei Knoten über einen Pfad, jedoch nicht direkt[23] miteinander verbunden sind, können weitere Knoten, die auf diesem Pfad liegen, als zusätzliche Informationsquellen verwendet werden. Im Abschnitt 6.3 des vorherigen Kapitels wurde darauf hingewiesen, dass innerhalb der konzeptuellen Abbildung Zusammenhänge zwischen verschiedenen Entitäten existieren. Im Folgenden werden die Grundlagen für die Bestimmung von Teilgraphen des Ontologiegraphen vorgestellt, die Zusammenhänge zwischen Knoten repräsentieren. Im darauf folgenden Abschnitt 7.5 wird die darauf aufbauende graphbasierte Referenzbestimmung vorgestellt.

Minimaler Spannbaum Ausgangspunkt für die Suche nach einer Verbindung zwischen einer zuvor gegebenen Menge an Terminalknoten ist die Bestimmung des minimalen Spannbaums eines Graphen.

[23] „Direkt“ entspricht einem Pfad der der Länge 1.

Ein minimaler Spannbaum ist definiert als (vgl. [245, 151]):

Definition 7.6 (Minimaler Spannbaum)**.** *Ausgangspunkt bildet ein ungerichteter Graph $\mathcal{G} = (V, E)$. Ein minimaler Spannbaum beschreibt eine Verbindung zwischen einer zuvor definierten Teilmenge von Knoten $L \subseteq V$, diese Menge wird ebenfalls als Terminalknotenmenge bezeichnet. Minimal bedeutet hierbei die Minimierung einer Kostenfunktion $c(e)$ mit $e \in E$ und $c : E \to \mathbb{R}_+$, d.h. die Kosten sind hierbei den Kanten zugewiesen und sind von der Anwendung vorgegeben. Der minimale Spannbaum $\mathcal{M} = (L, T)$ ist somit definiert durch eine Menge von Kanten $T \subseteq E$, so dass für diesen Baum eine Minimierung von $c(T) : \sum_{e \in T} c(e)$ gilt.*

Es ist ein wesentliches Merkmal eines minimalen Spannbaums, dass zum einen alle Knoten der Terminalknotenmenge $v \in L$ darin enthalten sein müssen und dass zu minimalen Kosten eine Verbindung zwischen ihnen existieren muss. Zum anderen dürfen *keine weiteren Knoten* $v \in L \cap V$ im Spannbaum vorkommen (siehe mittlerer Graph in Abbildung 7.6) (vgl. [65][24]). Der minimale Spannbaum wird beginnend mit einem Wurzelknoten w (der willkürlich aus der Menge seiner Knoten, *d.h.* $w \in L$, ausgewählt werden kann) mit zusammenhängenden Knoten aufgebaut.

Steinerbaum Im Gegensatz hierzu ist die Verwendung von *nichtterminalen* Knoten zur weiteren Minimalisierung der oben genannten Konstenfunktion bei Steinerbäumen möglich. Voraussetzung hierfür ist die Möglichkeit einer weiteren Minimierung hinsichtlich der für die Verbindung der Terminalknoten auflaufenden Kosten. Die initiale Motivation für einen Steinerbaum, berichten Courant und Robbins [52], war das Problem drei Städte A, B, C über ein System von Straßen von minimaler Länge zur verbinden. Das Steinerbaum-Problem ist eine Generalisierung des Problems des minimalen Spannbaums und ist definiert als (vgl. [115, 181, 245]):

Definition 7.7 (Steinerbaum)**.** *Ausgangspunkt bildet ein ungerichteter Graph $\mathcal{G} = (V, E)$. Ein Steinerbaum beschreibt einen minimalen Spannbaum $\mathcal{B} = (U, T)$, der eine vorher gegebene Knotenmenge L als Teilmenge*

[24] *„A tree connecting“ L „without using any Steiner points is called a spanning tree and a shortest spanning tree is called a minimal spanning tree.“* [65]. Steiner points beschreiben zusätzliche Knoten, die nicht in der vorgegebenen Knotenmenge $L \subseteq V$ enthalten sind.

der ihm zugeordneten Knotenmenge U enthält; $L \subseteq U \subseteq V$. Die Kantenmenge $T \subseteq E$ erfüllt ebenfalls die Bedingung der Minimierung der Kostenfunktion $c(T) : \sum_{e \in T} c(e)$. Der Steinerbaum ist somit ein minimaler Spannbaum hinsichtlich der Knotenmenge L. Für das Erreichen einer optimalen Minimierung der Kostenfunktion kann dieser Baum jedoch zusätzliche Knoten $U \cap L$ enthalten.

Abbildung 7.6 zeigt den Unterschied zwischen Steinerbaum (Abbildung 7.6(c)) und minimalem Spannbaum (Abbildung 7.6(b)). Während beim minimalen Spannbaum die Verbindungen auf die Verwendung von Terminalknoten beschränkt sind ($c(e) = 8$), können beim Steinerbaum zusätzliche Knoten berücksichtigt werden ($c(e) = 2$). Bezogen auf das hier abgebildete Beispiel können die Gesamtkosten durch die zusätzliche Berücksichtigung von Verbindungen über Nichtterminalknoten weiter minimiert werden (Kosten im Verhältnis $4 : 1$), *d.h.* der Steinerbaum ist im Rahmen dieses Beispiels dem minimalen Spannbaum vorzuziehen.

Steiner-Gruppen-Problem Das Steiner-Gruppen-Problem (engl. Group-Steiner-Problem) baut auf dem Ausgangsproblem des Steinerbaums auf, *d.h.* auf dessen Ziel, für eine gegebene Terminalknotenmenge den optimalen Verbindungsgraph zu bestimmen. Hierbei beruht die Aufgabe auf einer Menge von Mengen von Knoten, wobei je Knotenmenge jeweils ein Knoten gewählt werden muss. Diese Wahl basiert auf der Aufgabe einen Steinerbaum anhand dieser Auswahl zu erzeugen, der hinsichtlich der Kostenfunktion das minimale Ergebnis ermöglicht. Dieses wird als Steiner-Gruppen-Problem (vgl. [115, 181]) bezeichnet, das als Erweiterung des Steinerbaum Problems betrachtet werden kann. Das Steiner-Gruppen-Problem ist definiert als:

Definition 7.8 (Steiner-Gruppen-Problem)**.** *Ausgehend von einem ungerichteten Graphen $\mathcal{G} = (V, E)$, einer Menge von Gruppen (Knotenmengen) $I_1, I_2 ... I_n$ und unter der Berücksichtigung einer Kostenfunktion für die gegebenen Kantengewichtungen $c(e)$, ist es die Aufgabe einen minimalen Steinerbaum zu berechnen, der von jeder Gruppe mindestens einen Knoten enthält. Ohne Verlust der Allgemeinheit kann davon ausgegangen werden, dass es einen Knoten v gibt, der die Wurzel eines solchen Baums repräsentiert. Gilt $|I_i| = 1$, d.h. jede Gruppe besteht nur aus einem Knoten, so handelt es sich um das zuvor beschriebene „klassische" Steinerbaum-Problem.*

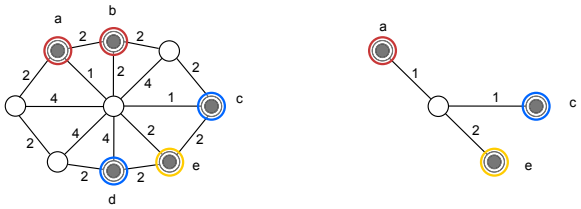


Abbildung 7.7.: Beispiel Steiner-Gruppen-Problem

In Abbildung 7.7 ist ein Beispiel für dieses Problem aufgezeigt. Es sind drei verschiedene Knotenmengen gegeben, $S_1 = \{a, b\}$, $S_2 = \{c, d\}$ und $S_3 = \{e\}$. Aus der Menge der möglichen Steinerbäume, welche die verschiedenen Teilmengen miteinander verbinden, wird derjenige mit den minimalen Kosten bestimmt. Dieser ist innerhalb der Abbildung rechts dargestellt.

7.5. Graphbasierte Referenzbestimmung

Die nachfolgende Vorgehensweise konkretisiert die in Abschnitt 6.2.1 vorgestellte Übertragung des Modells der Zwei-Ebenen Semantik auf Ontologien. Insbesondere wird der im Postulat 6.4 angesprochene graphbasierte Zusammenhang dargestellt.

Das Problem der Disambiguierung lässt sich übertragen auf das Steiner-Gruppen-Problem (siehe Definition 7.8). Für die Intension sowie Extension des allgemeinen Modells der Zwei-Ebenen Semantik (vgl. Abbildung 6.3) wird ein RDF-Instanz Graph $\mathcal{G} = (V, E)$ (siehe Definition 7.3) verwendet. Die Disambiguierung selbst basiert nur auf dem Instanzgraphen, während der Schemagraph in der Vorprozessierung, *z.B.* zur Berechnung von Knotengewichtungen, Anwendung finden kann (siehe Kapitel 12). Der Instanzgraph wurde als gerichteter Graph definiert. Wird das Steiner-Gruppen-Problem mit einem zugrunde liegenden Instanzgraph angewandt, so können

dessen Richtungsangaben vernachlässigt werden.[25] Für diesen Graph $\mathcal{G}^{\mathcal{U}} = (V, E^U)$[26] gilt somit $E^U = E$, jedoch gilt ebenfalls eine symmetrische Kantenrelation, *d.h.* $E^U = E^{U^{-1}}$.

Basierend auf der Graphrepräsentation erfolgt die Referenzbestimmung gemäß der Zwei-Ebenen Semantik:

Lexikalisch-semantische-Ebene Ausgehend von einem Satz $\mathcal{A}$, der eine jeweilige Aussage besitzt, werden zunächst die Bezeichner und anschließend die Intensionen (Seme) bestimmt (siehe Abbildung 7.8).

- Basierend auf der textuellen Vorverarbeitung wird bei der Disambiguierung eines Satzes $\mathcal{A} = \{w_1, ...w_n\}$ mit den Worten w_i[27] als Eingabemenge übergeben. Die Menge der zu berücksichtigenden Wörter für die Disambiguierung von Entitäten wird eingeschränkt auf diejenigen Terme, die als Bezeichner von Entitäten verwendet werden können. Die Bezeichner der Entitäten werden durch die vorausgehende textuelle Analyse erkannt (siehe Abschnitt 7.3).
- Die Bestimmung der Intensionen erfolgt über die Selektion der Ontologieelemente, die dem jeweiligen Bezeichner entsprechen. Somit ist die Menge möglicher Knoten je Bezeichner innerhalb eines Satzes $\mathcal{A}$ gegeben durch die Menge von Intensionen, $Int_i, 1 \leq i \leq n$.

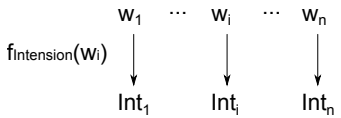


Abbildung 7.8.: Zusammenhang zw. Bezeichner und Intension (vgl. Abbildung 6.2)

[25] Jede gerichtete Property des RDF-Instanz Graphen besitzt eine implizite, *d.h.* nicht in der Ontologiedefinition enthaltene, Umkehrproperty. Beispielsweise besitzt die Beziehung `<geo:Karlsruhe geo:liegtIn geo:BadenWürttemberg>` die implizite Umkehrbeziehung `<geo:BadenWürttemberg geo:enthältStadt geo:Karlsruhe>`.

[26] U = ungerichtet

[27] Bei der allgemeinen Disambiguierung wird hierbei auf Lexeme verwiesen.

Konzeptuelle Ebene und Disambiguierung *Die Übertragung auf das Steiner-Gruppen-Problem stellt die Bedingung, dass die aufzufindenden Seme*[28] *der verschiedenen Bezeichner miteinander verbunden sind.* Dies zeigt den konzeptuellen Zusammenhang zwischen ihnen auf und erlaubt somit die Darstellung eines Graph-Modells für einen Satz $\mathcal{A}$.

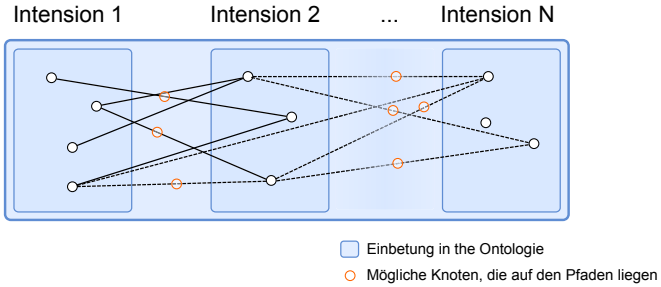


Abbildung 7.9.: Zusammenhang zwischen mehreren Adressen verschiedener Intensionen innerhalb der Wissensbasis

Dies ist in Abbildung 7.9 dargestellt. Je analysiertem Bezeichner w_i existiert eine Gruppe von Adressen, welche die Intension Int_i für diesen Bezeichner darstellt. Diese Intensionen zeigen jeweils einen Ausschnitt des gesamten konzeptuellen Systems, *d.h.* der Ontologie. Der Zusammenhang zwischen den Intensionen wird über deren Extensionen dargestellt, *d.h.* deren Einbindung in die Ontologie und somit deren Einbindung in den Graphen. Hierbei können sich die verschiedenen Extensionen überlappen bzw. es kann eine Verbindung zwischen diesen existieren. Es kann jedoch davon ausgegangen werden, dass es viele unterschiedliche Graphen gibt, die einen Zusammenhang für denselben Satz darstellen (wie in der Abbildung angedeutet). Ein Abwägen der Relevanz eines Graphen für den vorliegenden Satz muss daher durch zuvor festgelegte Kriterien ermöglicht werden. Im Kontext des Steinerbaums erfolgt dies durch die zuvorvorgestellte Kostenfunktion für die Kanten. Die Interpretation

[28] In Abschnitt 5.2.1 wurde der Begriff „Sem“ erstmals verwendet und als die kleinste Einheit der Bedeutung sprachlicher Zeichen definiert. Im Zusammenhang mit der Ontologie-basierten Disambiguierung entspricht dessen übertragene Bedeutung der einer initialen Adresse in der Intensionsmenge, *d.h.* die Seme des Bezeichners „A“ referenzieren die initialen Adressen der Ontologieelemente in der Intension dieses Bezeichners. Der zusätzliche Begriff wird eingeführt, um den Bezug zu diesen *initialen Adressen* festzulegen.

des vorliegenden Satzes, *d.h.* die Aussage des Satzes, ist definiert durch:

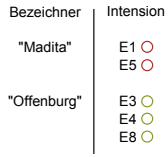
Postulat 7.9 (Satzaussage)**.** *Ausgehend von einem ungerichteten RDF-Instanz-Graphen $\mathcal{G}^u = (V, E^u)$, einer Menge von Intensionen, d.h. $Int_1, Int_2 ... Int_n$ für die gegebenen Entitätsbezeichner im Satz und dem Kantengewicht $c(e)$ wird* ***die Satzaussage*** *$M(A)$* ***bestimmt durch den minimalen Steiner-Gruppen Graph****. Dieser berechnet je Intension diejenige Adresse, die im Rahmen der Aussage des vorliegenden Satzes die gültige Referenz für den gegebenen Bezeichner darstellt. Dies bedeutet, der Ergebnisgraph repräsentiert den Satz und stellt somit dessen semantische Aussage dar.*

Das kann ebenfalls auf Texte bzw. Dokumente übertragen werden. Hier bildet das gesamte Dokument den Kontext, *d.h.* die Menge der Intensionen enthält alle Entitätsbezeichner innerhalb des Dokuments.

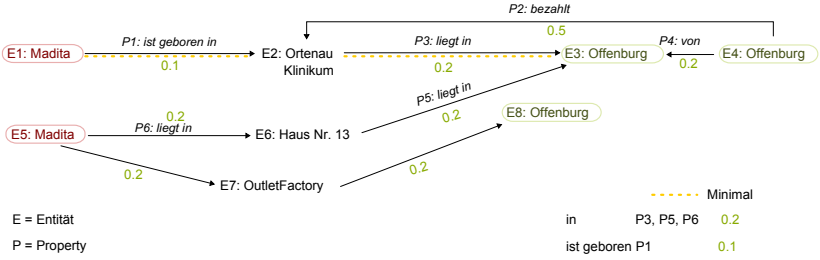
7.5.1. Beispiel

Abbildung 7.10 zeigt die Anwendung der zuvor vorgestellten Vorgehensweise auf. Dieses Beispiel ist eine Übertragung des in Abschnitt 6.3.1 gezeigten Beispiels im Hinblick auf die Referenzbestimmung aus der Menge der möglichen Intensionen.

Ausgangspunkt ist ebenfalls der Satz: *„Madita ist geboren in Offenburg“*. Basierend auf der Vorprozessierung, *z.B.* des Zusammenhanges zwischen Text und Kantenbezeichnern, kann eine Gewichtung für diese vorgenommen werden (siehe Abbildung 7.10(b) rechts unten). In dem gezeigten Beispiel erfolgt eine Beschränkung auf die Entitätsbezeichner *„Madita“* und *„Offenburg“*. Dem Bezeichner *„Madita“* ist die Instanzmenge $Int_{\text{Madita}} = [i_1, i_2]$ mit $i_1 =$ E1:Madita und $i_2 =$ E5:Madita zugeordnet. Für den Bezeichner *„Offenburg“* wurde $Int_{\text{Offenburg}} = [i_3, i_4, i_5]$ bestimmt mit $i_3 =$ E3:Offenburg, $i_4 =$ E4:Offenburg und $i_5 =$ E8:Offenburg (siehe Abbildung 7.10(a)). Basierend auf einer Anwendung des vorgestellten Disambiguierungsprozesses wird i_1 als Referenz für *„Madita“* und i_3 für *Offenburg* bestimmt. Dies ist darin begründet, dass der (gelbe) Graph, der diese beiden Referenzen enthält, im Verhältnis zu allen anderen möglichen Graphen die minimalen Kosten gemäß der Kostenfunktion besitzt.



(a) Bezeichner zu Ontologie-element-zuordnung



(b) Ontologie-basierte Disambiguierung

Abbildung 7.10.: Beispiel: Zusammenhang zwischen Zwei-Ebenen Semantik und Ontologie

7.6. Spreading Activation

Der Begriff Spreading Activation geht zurück auf die Doktorarbeit von Allan Quillian [185]. Die Ursprünge hierzu beschrieb er bereits 1962 im Rahmen einer Arbeit zur maschinellen Übersetzung. Er befasste sich mit dem Informationsmodell, das für eine erfolgreiche Übersetzung von natürlich-sprachlichen Texten benötigt wird. Er erstellt ein eigenes Modell, das die menschliche Art und Weise der Informationsverarbeitung abbildet. Konfrontiert mit einem natürlich-sprachlichen Wort oder einer Aussage beginnt der Mensch intuitiv dieses mit gelerntem Wissen in Verbindung zu bringen, *d.h.* dem Konzept[29] das er damit in Verbindung bringt. Wird er beispielsweise mit dem Wort *„VW Käfer“* konfrontiert, werden diverse Gedankengänge angestoßen, *z.B.* es dem Konzept *Auto* zuzuordnen.

[29] Ein Konzept kann eine oder mehrere Eigenschaften eines Objekts, dessen Beziehungen und Fähigkeiten beschreiben.

Ein *Auto* verfügt über Räder, es wird durch einen Motor angetrieben *etc.* Nach diesem ersten Impuls beziehungsweise diesen ersten Gedanken hinsichtlich direkt zugeordnetem Wissen, fängt der Mensch an diesen Impuls mit weitergehendem Wissen in Beziehung, zu bringen, *z.B.* dass Räder von Reifen umgeben und diese mit Luft gefüllt sind. Er beginnt ebenfalls abzuwägen, ob Bereiche davon in Anbetracht der erhaltenen Information von Bedeutung sind. Daraus folgt, dass innerhalb dieses Modells, das von Quillian als „Semantic Memory" bezeichnet wird, dass die Konzepte nicht für sich allein, sondern miteinander in Beziehung stehen. Somit verfügt jedes Konzept über Eigenschaften, die als Relationen modelliert sind. In ihrer Gesamtheit bilden diese Konzepte ein Netz mit einer unterschiedlichen Dichte an Verknüpfungen. Diese hängt vom entstandenen Zusammenhang zwischen den Konzepten ab, *d.h.* dem vorhandenen Wissen. Quillian spezifiziert weiterhin, welche schematischen Aufgaben diese Verknüpfungen zusätzlich haben. Er unterscheidet fünf Arten von Relationen: *1)* unter- und übergeordnete (is a) Relationen *2)* Relationen zur Modifizierung *3)* Relationen zur Trennung von Mengen *4)* Relationen zur Verbindung von Mengen und *5)* Klassen von Relationen.

Collins und Quillian [50] versuchen diese Theorie mit psychologischen Tests zu untermauern und gleichzeitig mehr über den Aufbau dieses Wissensmodells zu erfahren. So stellen sie beispielsweise bei der Untersuchung des konzeptuellen Aufbaus einer Wissensbasis fest, dass die Probanden innerhalb eines Tests wesentlicher schneller die Frage: *„Kann ein Kanarienvogel fliegen?"* beantworten konnten als die Frage *„Besitzt ein Kanarienvogel eine Haut?"*. Daraus zogen sie den Schluss, dass das Konzept „Tier" mit der Eigenschaft „Haut" dem Konzept „Vogel" mit der Eigenschaft „Federn" übergeordnet ist, *d.h.* die Reaktionsfähigkeit über die Struktur der Wissensbasis beeinflusst wird (siehe auch [217], S. 250). Der Theorie von Quillian als generelles Wissensmodell wird von Tulving [231] widersprochen. Er betrachtete das vorgeschlagene Modell als zu restriktiv für organisiertes Wissen hinsichtlich der Verarbeitung von Sprache und anderen Symbolen, *d.h.* die Zuordnung von Wörtern zu Referenzen beziehungsweise die Bedeutung der Wörter sowie die Beziehungen zwischen diesen. Pfuhl [175]

beschreibt dieses als *„geistigen“* Thesaurus[30]. Für die Abbildung des menschlichen Wissensverarbeitungsprozesses ist der Meinung von Tulving folgend ein zusätzliches epistemisches Modell notwendig. Ein epistemisches Modell beschreibt autobiographische Vorgänge beziehungsweise spezifische Ereignisse und Episoden. So wäre eine Frage, die das epistemische Gedächtnis anspricht *„Wo waren Sie und was haben Sie gemacht, als die Bundeskanzlerin Fr. Merkel in Offenburg war?* . Im Gegensatz dazu würde die Frage *„Wo hat die Bundeskanzlerin Fr. Merkel am 16.03.2011 eine Wahlkampfrede gehalten?“* das semantische Gedächtnis ansprechen. Schank [199] verwirft die Theorien von Quillian und Tulving. Er führt die Conceptual Dependency Theorie ein, die episodisches und semantisches Wissen zusammenführt. Semantisches Wissen wird hier durch Episoden dargestellt, *z.B.* beim Satz *„Madita isst ein Eis“* wird *„isst“* dem Prozess *„essen“* zugeordnet. Dieser enthält den Einsatz des Mundes *etc.* Nur abstraktes Wissen, *d.h.* Wissen ohne Erfahrungen, wird in der von Quillian vorgeschlagenen Form gespeichert.

Selbst in den Anpassungen und Änderungen von Tulving und Schank sind die Ideen von Quillian für die Modellierung von abstraktem, *d.h.* allgemein gültigem Wissen enthalten. Auffallend ist auch die Ähnlichkeit seiner Theorie zu den Thesen, auf denen das Semantic Web aufbaut (vgl. Kapitel 3). Die von Quillian definierten „Konzepte“ werden im Semantic Web in *Instanzen* und schematische *Konzepte (Klassen)* unterschieden. Die von ihm vorgeschlagenen Properties werden im Semantic Web weiter unterschieden in Data- und Object-Properties. Die Verbindung über Typzuordnungen zu Instanzen wird auch in seinem Modell bereits berücksichtigt. Die Erstellung von eigenen Relationen, deren Typisierung, Erzeugung von konjunktiven und disjunktiven Gruppen sind Eigenschaften, die in beiden Modellen zu den Grundlagen gehören. Zudem ist in beiden Modellen die Darstellung von epistemischem Wissen nicht berücksichtigt, *d.h.* eine Trennung in der oben genannten Form ist nicht möglich.

[30] Ein Thesaurus baut auf einem kontrollierten Vokabular auf, *d.h.* einer endlichen Menge von zuvor hinzugefügten Wörtern. Er bezeichnet ein Wortnetz, das diese Begriffe miteinander in Beziehung setzt. Eine Ontologie bietet ebenfalls die Möglichkeit über den Zusammenhang von Wort zu Intension auf das extensional konzeptuelle System zuzugreifen und hier Beziehungen zwischen den Adressen zu ermitteln. Der Unterschied liegt hierbei in der definierten Semantik, die ein Thesaurus nicht bietet.

Zusammenfassend kann festgestellt werden, dass Quillian der Idee der Wissensmodellierung bereits 1962 eine eigene Form durch das von ihm entwickelte Modell gegeben hat. Dieses basiert auf seiner Interpretation des menschlichen Prozesses mit Wissen umzugehen und es zu verknüpfen. Das von ihm entwickelte Graphmodel lässt sich auch in RDF teilweise wiedererkennen und es lässt sich somit ein Zusammenhang zum Semantic Web herstellen. Er verfolgt mit seinem Modell das Ziel *„a medium appropriate to serve as a mechanical equivalent of human understanding"* [184] zu kreieren.

Die ursprüngliche Idee von Quillian findet in der Psychologie (*z.B.* [49, 8]) Anklang und natürlich weiterhin in der Informatik (*z.B.* [180]; siehe hierzu ebenfalls Kapitel 14).

7.6.1. Zusammenhang mit der menschlichen Informationsverarbeitung

Joachim Diedrich [61] formuliert im Zusammenhang mit der menschlichen Informationsverarbeitung und Spreading Activation, dass *„it is assumed that cognitive processes are best described by parallel relexation methods or the straigth-forward propagation of activation throughout a memory network"*. Unabhängig von der vagen Beschreibung von Spreading Activation kann diese somit als mögliche Technik mit dem Ziel der Nachbildung der menschlichen kognitiven Verarbeitung angesehen werden. Collins bezeichnet Spreading Activation sogar als *„theory of human semantic processing"* [49].

Bei der Suche nach einer Möglichkeit, sich dem Prozess der menschlichen Informationsverarbeitung anzunähern, wird versucht dessen Vorgehensweise zu imitieren. So wird vermutet, dass der Impuls, der durch die Konfrontation beziehungsweise der Eingabe eines Objektes, *d.h.* Wort oder Satz, erzeugt wird, eine bestimmte Hirnregion anregt, in der relevante Informationen gespeichert sind. Dieses Ansprechen von gespeicherter Information wird durch einen Aktivierungswert nachgebildet, *d.h.* die in dieser Hirnregion gespeicherte Information wird aktiviert. Die weitere Annahme ist nun, dass diese Aktivität sich im menschlichen Gehirn ausbreitet und die mit den zuerst aktivierten Konzepten in Beziehung stehenden Konzepte erfasst

und ebenfalls aktiviert werden, *d.h.* die Aktivität des zunächst aktivierten Objekts weitergegeben wird. Übertragen auf das oben dargestellte Hintergrundmodell, fließt die Aktivität somit über die modellierten Kanten zu in Beziehung stehenden Konzepten.

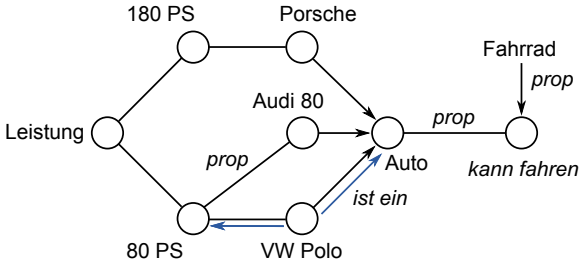


Abbildung 7.11.: Beispiel zur menschlichen Informationsverarbeitung

Ein Beispiel ist in Abbildung 7.11 gegeben. Ausgehend von *VW Polo* wird das Konzept *Auto* aktiviert und die Property *80 PS*. Es muss jedoch abgewogen werden, wie wichtig jedes weitere Konzept ist und wieviel Aktivität damit übertragen wird. Ausgehend vom vorliegenden Beispiel stellt sich die Frage, ob es im gegebenen Kontext von Relevanz ist, dass neben einem Auto auch ein Fahrrad in der Lage ist zu fahren. Der Kontext ist hierbei der Ausgangspunkt, *d.h. VW Polo*. Oftmals wird die Stärke des Zusammenhangs zweier Konzepte über deren Entfernung im Hintergrundmodell ausgedrückt, *d.h.* längere Wege deuten auf einen geringen Zusammenhang hin.

Als allgemeiner Oberbegriff wird das verwendete Hintergrundmodell durch den Ausdruck „Assoziatives Netz“ bezeichnet. Ein assoziatives Netz ist definiert als ein Netz mit Knoten, die Informationsobjekte repräsentieren während die Kanten des Netzes den Zusammenhang zwischen diesen darstellen. Das einer Kante zugeordnete Gewicht repräsentiert die Stärke des Zusammenhangs zwischen den Informationsobjekten. Crestani definiert ein assoziatives Netz als:

Definition 7.10 (Assoziatives Netz)**.** *„This is a generic network of information items in which information items are represented by nodes, and links express sometimes undefined and unlabeled associative relations among information items.“ [53]*

Ein assoziatives Netz fungiert als grundlegende Definition und gibt somit den äußeren Rahmen eines Hintergrundmodells vor. Verglichen mit den Modellen des Semantic Web (siehe Abschnitt 3) wird deutlich, dass eine Ontologie ein konkretes assoziatives Netz darstellt. Konkret, da die Kanten eine semantische Bedeutung zwischen zwei Informationsobjekten beschreiben. Der im vorigen Abschnitt vorgestellte Zusammenhang zwischen dem Modell von Quillian und den Thesen des Semantic Web gibt auch diese Konkretisierung wieder. Der Zusammenhang Spreading Activation und assoziatives Netz wird ebenfalls in [18, 200] behandelt. Die obige Nennung von Konzepten bedeutet jedoch keine Beschränkung auf den Schema-Part einer Ontologie, sondern es kann ebenfalls auf Instanzen Bezug genommen werden.

Betrachtet man die dargestellte menschliche Informationsverarbeitung unter dem Gesichtspunkt, dass ein assoziatives Netz als Hintergrundmodell verwendet wird, so erfolgt bei der Konfrontation mit einem Ereignis zunächst die initiale Aktivierung einer Teilmenge von Knoten. Anderson [8] bezeichnet diese als *Primary Units*. Anderson beschreibt, dass nun ausgehend von diesen Kanten sich die Aktivierung über das Netzwerk ausbreitet. Es gibt zwei mögliche Abbruchbedingungen, die in Abschnitt 7.6.2 vorgestellt werden.

7.6.2. Technik

Die Ausbreitung der Aktivierung erfolgt in den meisten Fällen automatisch (*d.h.* nach Aktivierung des Knotens wird die Aktivität sofort weitergegeben), ungerichtet (*d.h.* die Richtung der Kante ist nicht von Bedeutung) und parallel (*d.h.* alle aktivierten Knoten geben gleichzeitig ihre Aktivierung weiter) (vgl. [61]). Der Prozess selbst ist hierbei frei von Interpretation, d.h. es werden nur nicht-symbolische Nachrichten übertragen. Die Auswertung möglicher Resultate bedarf jedoch einer externen Komponente, des Path-Evaluator [111].

Ein mögliches Resultat eines Spreading Activation Algorithmus wird durch einen Pfad ausgedrückt, der entweder eine Menge von miteinander in Beziehung stehenden aktivierten Elementen beinhaltet oder ein beziehungsweise mehrere „Marker“ besitzt (siehe *Digital* or *Analog* Spreading Activation unten). Die Generierung eines solchen Pfades ist als Resultat eines Propagation-Schrittes möglich,

d.h. einem Schritt, in dem die Aktivierungen zu den benachbarten Knoten weitergegeben werden. Waltz teilte 1985 Spreading Activation Algorithmen in zwei wesentliche Klassen von Algorithmen ein. Er unterscheidet zwischen „Digital Spreading Activation“ und „Analog Spreading Activation“ (vgl. [238]):

Definition 7.11 (Digital Spreading Activation)**.** *„Digital Spreading Activation is a class of marker-passing algorithms which perform a breadth-first search for shortest paths on a relational network.“ [238]*

„Digital“ bedeutet hier, dass während der Initialisierung verschiedene *Marker* Knoten aktiviert werden. Im nachfolgenden Spreading-Activation-Prozess wird ein diskreter Wert je Marker an seine Nachbarknoten weitergegeben und somit die Information des Markers durch das Netz ausgebreitet. Treffen sich die Marker in einem Knoten, so wird der Pfad zurückgegeben, der den Zusammenhang der Marker aufzeigt. Normalerweise existieren mehrere mögliche Pfade. Die Kanten der verwendeten Netzwerke zeigen symbolische Zusammenhänge zwischen den Knoten auf. Der Zusammenhang selbst wird hierbei nicht gewichtet. Lycan [142] beschreibt im Rahmen seiner Causal-Historical-Theorie ein Beispiel, das die Verwendung von Markern wiedergibt. Er beschreibt, dass die Probleme, mit welchen der Prozess der Referenzerkennung konfrontiert ist (im konkreten Fall einer Person) bereits mit der initialen Namensvergabe auftreten. Bei einer Person erfolgt dies normalerweise bei der Geburt. Dies ist im übertragenen Sinne die Vergabe des Markers an den initial zugewiesenen Knoten des Wissensnetzwerks. Er beschreibt ebenfalls, dass bei der Weitergabe des Personennamens die aktuelle Person über diesen durch die vorhergehende Person informiert wird. Dadurch ergibt sich eine Kette der Weitergabe, die sich auch auf mehrere Pfade verteilen kann. Dieses Beispiel kann direkt auf die Weitergabe von markerspezifischer Information durch das Spreadingverfahren übertragen werden. Bereits Lycan weist auf die Mehrdeutigkeit von Namen hin, *d.h.* in dem von ihm vorgestellten Fall sind das Namen, die mehrere Personen tragen.[31] Dieses Problem wird in Kapitel 8 behandelt.

[31] Hierbei wird an das Eingangs erwähnte Beispiel der Namenvergabe „Marie“ für Neugeborene verwiesen (siehe Kapitel 1).

Definition 7.12 (Analog Spreading Activation). *„Analog Spreading Activation takes place on a weighted netwok of associations, where "activation energy" is distributed over the network based on some mathematical function of the strength of connections. [238]*

„Analog" bedeutet hierbei eine initiale Aktivierung basierend auf einem numerischen Wert, der basierend auf Kantengewichtungen und weiteren Einflussfaktoren während des Spreading verändert wird. Die aktivierten Knoten bleiben oftmals durchgehend aktiviert, wodurch das Spreading Aktivierungspfade, die einen numerischen Zusammenhang repräsentieren, aufgezeigt werden. Diese Aktivierungspfade sind das Resultat eines analogen Ansatzes. Analoge Ansätze bieten gegenüber den digitalen Ansätzen die Möglichkeit, Ähnlichkeiten zwischen Objekten numerisch darzustellen und somit auch durch das Spreading auf weitere Knoten implizit zu übertragen. Als Resultat werden Aktivierungspfade, die Datenmuster beinhalten, ausgegeben. Ein Beispiel des analogen Spreading zeigt die Untersuchung von Elman [66] basierend auf einem Korpus mit 10000 Sätzen. Das Experiment hatte zur Aufgabe, basierend auf einem analogen Spreading, die Vorhersage des nächsten Wortes des Satzes vorzunehmen. Bei der Analyse der Muster stellte er fest, dass das Aktivierungsmuster für „woman" im Graphen nahezu dem von „girl" entsprach. Bei „man" and „woman" war eine Ähnlichkeit ebenfalls auffindbar. Weitere Informationen zu den „Connectionist Models", die Spreading Activation auf gewichteten Netzen behandeln, sind im Technical Report von Thomas und McClelland [228] zu finden.

Aktuelle Ansätze (*z.B.* [180, 105, 230],[255, 256]) verwenden durchweg eine Kombination beider Ansätze. Innerhalb dieser Ansätze wird jedoch zum Teil eine Aufteilung in Teilprozessschritte vorgenommen, die einer der beiden Verfahrensweisen zugeordnet werden können. Eine Kombination beider Klassen wurde von Hendler [111] erstmals vorgestellt. Er verwendet eine Erweiterung eines digitalen Ansatzes durch eine numerische Auswertung von Ähnlichkeiten zwischen Objekteigenschaften.

Die grundlegenden Prozessschritte, die beide Varianten gemeinsam haben, formuliert Collins [49] in den von ihm aufgestellten Annahmen zur Aktivitätsausbreitung. Im Folgenden ist ein Auszug an Regeln wiedergegeben, die in den Ansätzen (*z.B.* [118, 105, 230],[255, 256, 257]) in Teilen wiederzufinden sind:

1. Bei Prozessieren eines Konzeptes wird dessen Aktivität an die mit diesem in Beziehung stehenden Konzepte weitergegeben.
2. Die Weitergabe der Aktivität kann je Knoten nur zu einem Zeitpunkt erfolgen.
3. Die Aktivität nimmt mit der ansteigenden Länge des Pfades ab, *d.h.* wird mit jedem Schritt geringer.
4. Der Wert der Aktivierung kann als eine variable Größe interpretiert werden. Wird ein Konzept von verschiedenen anderen Konzepten aktiviert, können diese Aktivierungen aufaddiert werden. Erreicht ein Konzept einen zu definierenden Schwellwert, deutet dies auf eine Relevanz des dafür verantwortlichen beziehungsweise damit in Beziehung stehenden Pfades hin.

Spreading Activation Algorithmen können abgesehen von der oberen Unterscheidung nach vier Kriterien klassifiziert werden (vgl. [61]):

- *gerichtet - ungerichtet*: Vorhandene Kantenrichtungen im Graphen bestimmen den Fluss des Spreading Verfahrens oder werden ignoriert.
- *best first - breadth first*: Im ersten Fall gibt im nächsten Spreading-Schritt der am höchsten aktivierte Knoten seine Aktivierung an seine Kindknoten weiter. Andernfalls werden von allen aktivierten Knoten die jeweiligen Aktivierungen an die Kindknoten weitergegeben. Im letzteren Fall findet somit eine Breitensuche statt.
- *dampening - decay*. „Dampening“ erzeugt eine Reduzierung der Aktivierung basierend auf zuvor festgelegten Kriterien, *z.B.* Kantentyp. „Decay“ beschreibt eine Abnahme des weitergegebenen Aktivierungswertes von Knoten zu Knoten, *d.h.* mit der Länge des Aktivierungspfades nimmt die weiterzugebende Aktivierung ab.

- *source specific - pulse specific*. Diedrich beschreibt die erste Klasse als eine Auswahl gegebener Knoten, die zuerst aktiviert werden und ständig Aktivierung an die mit ihnen in Beziehung stehenden Knoten abgeben. Bei *Pulse specific* hingegen wird die Aktivierung gleich einem Pulsschlag auf einmal durch das ganze Netz verteilt.

Die gegebenen Klassifikationen sind nicht klar voneinander getrennt. Sie können auch teilweise gleichzeitig in der Spreading Activation Methode zum Einsatz kommen. Eine Verfeinerung der oben genannten *grundlegenden* Kriterien findet sich ebenfalls in der Dissertation von Scott Preece [180] und den Arbeiten [49, 8].

Feststellung eines Resultats des Spreading Algorithmus

Analoges Spreading Activation wird oft im Zusammenhang mit einem Lernprozess eingesetzt, *d.h.* die Erstellung der Aktivitätsmuster ist hierbei ein Ziel, das durch das Lernverfahren erreicht werden soll. Die Terminierung ergibt sich durch das Ende des Lernprozesses, *z.B.* eine benutzerspezifische Bedingung (Beispielsweise durch Erreichen eines zuvor festgelegten Schwellwertes). Somit besteht das Endresultat in der Erzeugung des Aktivierungsmusters beziehungsweise das Auslesen von knotenspezifischen Aktivierungswerten.

Im Falle von digitalem Spreading beziehungsweise Spreading mit Markern kommen sogenannte *Path-Evaluators* zum Einsatz. Der Begriff wurde von James Hendler [111] geprägt und bildet einen der Schwerpunkte im Design von Marker-Passing-Systems (siehe seine Dissertation [110]). Die initial festgelegten Marker werden bei jedem Spreading Schritt an die jeweiligen Kindknoten weitergegeben. Im Verlauf des Spreading kommt es somit zu einem Aufeinandertreffen von verschiedenen Markern in diversen Knoten. Treffen alle Marker aufeinander, so handelt es sich um eine potentielle Lösung des durch das Netzwerk, die Marker und die Aktivierungen festgelegten Problems. Die Güte beziehungsweise Qualität einer möglichen Lösung bedarf jedoch einer Analyse. Diese Analyse wird durch den sogenannten „Path-Evaluator“ vorgenommen. Die Analyse des Path-Evaluator fokussiert auf zuvor festgelegte Kriterien, deren Güte anhand des aufgefundenen Pfades bestimmt werden. Genügt der Pfad den Anforderungen nicht, so wird dieser nicht als mögliche Lösung akzeptiert.

Die genannten Kriterien werden über programmspezifische Heuristiken ausgewertet, die den Pfad untersuchen.

Hendler unterscheidet zwei Arten der Mitteilung des Aufeinandertreffens von Markern: *(1)* Senden des ganzen Pfades *(2)* binäre Mitteilung, ob ein Aufeinandertreffen stattfand. Die Wahl hängt von der Aufgabe des Algorithmus ab.

Aus Effizienzgründen empfiehlt sich eine Abwägung, welche und wieviele Markerkollisionen überprüft werden sollen. Zu viele bedeutet einen Effizienzverlust, während zu wenige die Gefahr bergen, dass die „beste Lösung" sich in den *nicht untersuchten* Kollisionen verbirgt.

Formale Vorgehensweise

Crestani [53] definiert folgenden Ablauf für Spreading Activation Algorithmen (siehe Abbildung 7.12)[32]:

Die Phasen der Vorbereitung und der Nachbereitung bieten die Möglichkeit, die Aktivierung anhand äußerer Umstände, *d.h.* situationsabhängig, zu modifizieren. Beide Manipulationen sind optional und werden oftmals eingesetzt, um eine Abschwächung („decay") der Aktivierung zu vollziehen. Beide ermöglichen sowohl die Aktivierung des Knotens als auch den Verlauf des Spreading die Aktivierung weiterer Knoten und somit eine Beeinflussung des gesamten Knotennetzwerkes. Auch kann hier angegeben werden, welche der mit dem Knoten in Verbindung stehenden Knoten für die Weitergabe der Aktivierung ausgewählt werden (siehe hierzu auch Abschnitt 7.6.2).

Die eingehende Aktivierung I_j als Resultat eines Aktivierungspuls wird berechnet durch:

$$I_j = \sum_{i=0}^{n} (O_i \cdot w_{ij}) \tag{7.1}$$

[32] Diese grundlegende Vorgehensweise von Crestani hat bis heute Bestand, wie die Verweise auf diese Vorgehensweise in aktuellen Dissertationen belegen, *z.B.* Scheir [200] und Berger [18].

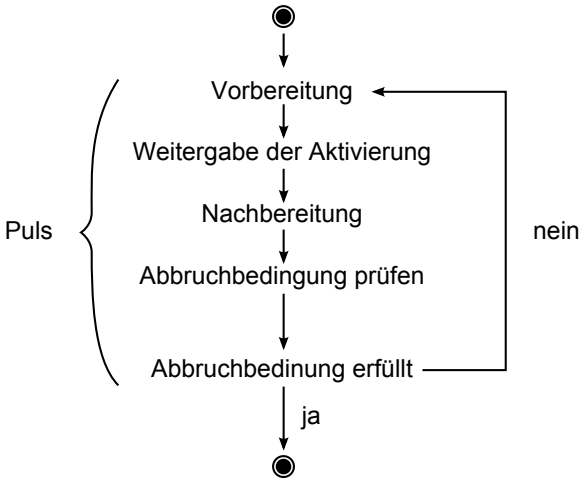


Abbildung 7.12.: Ablaufschema für Spreading Activation Algorithmen

In diesem Aktivierungsimpuls geben alle[33] n Knoten, die über eingehende[34] Verbindungen beim Knoten j verfügen ihre jeweiligen Ausgangsaktivierungen O_i an diesen Knoten weiter. Die Aktivierung wird beeinflusst durch das Kantengewicht w_{ij} der jeweiligen Kante, über welche die entsprechende Aktivierung weitergegeben wird.

Die Ausgangsaktivierung O_i eines Knoten i und somit die Aktivierung, die dieser Knoten weitergibt, wird berechnet durch:

$$O_i = f(I_i) \tag{7.2}$$

[33] Hierbei können bei modifizierten Algorithmen auch nur Ausschnitte der Menge der eingehenden Knoten berücksichtigt werden.

[34] Im Standardfall ist die Aktivierungsweitergabe nur über ausgehende und somit beim Knoten j eingehende Verbindungen möglich. Innerhalb einer spezifischen Algorithmusumsetzung kann von dieser Einschränkung abgewichen werden, *z.B.* durch ungerichtete Kanten.

Die Funktion $f(I_i)$ kann eine beliebige Funktion darstellen, *z.B.* eine lineare Funktion oder eine Schwellwertfunktion. Letztere wird sehr oft verwendet und ist definiert durch:

$$O_i = \begin{cases} 0 : I_i < k_i \\ 1 : I_i \geq k_i \end{cases}$$

Der Parameter k_i gibt hierbei den Schwellwert vor, der übertroffen werden muss damit eine Aktivierung weitergegeben wird. Zudem handelt es sich in der dargestellten Funktion um die Weitergabe binärer Aktivierungswerte.

Weitere Einzelheiten und Varianten zu der Durchführung von Vor- und Nachverabeitungsschritten ist in der Dissertation von Preece [180] beschrieben.[35]

Der Zusammenhang zwischen Spreading Activation und Neuronalen Netzwerken wird auf Grundlage der Definition von Neuronalen Netzen im Buch von Rummelhart und McClelland [194] sowie im Artikel von Berthold et al. [23] dargestellt.

Einschränkung des Spreading Prozesses

Eine unkontrollierte Durchführung des Spreading Activation Ansatzes birgt Risiken, die abgewogen werden müssen. Beispielsweise führt ein unbeschränktes Spreading zu einer kompletten Verbreitung der Aktivierung durch das ganze Netz. Dies erschwert zum einen das Auffinden von Aktivierungsmustern oder Pfaden und zum anderen bringt es hohe Bearbeitungszeiten mit sich. Auch werden Zusatzinformationen im Sinne der Verwendung von Labels, *d.h.* Knoten- beziehungsweise Kantennamen und semantischen Beschreibungen nicht analysiert.

[35] Er unterscheidet zwischen quellen- und zielspezifischer Beeinflussung der Spreadingwerte, *z.B.* „full strength-, unit-“ und „equal-distribution-spreading“ im Fall der Quelle und „summation, inverse destination frequency spreading“ sowie „decay factors“ im Fall des Ziels.

Dies wird deutlich in der Beschreibung von Anderson [8], der ebenfalls den Zusammenhang zur menschlichen Informationsverarbeitung aufzeigt. Seine Arbeit beschreibt die psychologische Verarbeitung von semantischen Informationen und weist bei dieser auf einen grundlegenden Fehler hin, der bei dieser Verarbeitung auftreten kann. Die Tendenz des Menschen zur „Overinclusion", beschreibt das Phänomen, dass eine Information beziehungsweise Instanz der falschen semantischen Klasse zugeordnet wird. Dies geschieht, obwohl die Information nicht vollständig dieser zugeordnet werden kann, jedoch in primären Charakteristika dieser entspricht. Ein typisches Beispiel ist seiner Meinung nach die Zuordnung von Schmetterlingen zu Vögeln. Diese Zuordnung erfolgt aufgrund der Tatsache, dass Schmetterlinge fliegen können. Jedoch handelt es sich bei Schmetterlingen um Insekten und bei Vögeln um Wirbeltiere. Daher ist eine Übereinstimmung im primären Merkmal „fliegen" ist gegeben. Diese stellt sich jedoch – bei näherem Blick – als falsch heraus. Die Auffassung des Autors dieser Arbeit kann dies insbesondere im Rahmen von Spreading-Algorithmen passieren, da diese in den meisten Fällen bereits eine Informationsverbindung ausnutzen ohne einen vollständigen Vergleich aller Assoziationen zwischen Klassen[36] durchzuführen.

Folgende Kategorien wurden entworfen, um sich den oben dargestellten Problemen anzunehmen. Diese Einschränkungen stützen sich ebenfalls auf die Arbeiten von Crestani und Preece [53, 180].

Einschränkung des Abstands Nach einem zuvor festgelegten Abstand wird keine Aktivierung mehr weitergegeben. Diese Beschränkung folgt der Annahme, dass nach einer gewissen Entfernung kein ableitbarer Zusammenhang von Relevanz mehr zwischen dem initialen Ursprungsknoten der Aktivierung und dem jetzigen Zielknoten vorhanden ist.

Einschränkung der ausgehenden Verbindungen Die Knoten, die über eine hohe Anzahl von Kanten verfügen, geben keine Aktivierung ab. Dieser Umstand kann auch auf Ontologien übertragen werden. Je nach Aufgabe des Algorithmus ist die Weitergabe der Aktivierung eines Konzeptknotens an alle seine zugehörigen Instanzknoten kontraproduktiv.

[36] Dies kann mit Algorithmen für Reasoning erreicht werden. Für nähere Informationen zu Reasoning siehe [35, 219]. Ebenfalls kann über semantische Kantengewichtungen darauf Einfluss genommen werden (siehe Kapitel 12).

Pfadeinschränkungen Der Pfad darf nur aus Knoten bestehen, die zuvor festgelegten Kriterien entsprechen.

Aktivierungseinschränkungen Hier handelt es sich um Schwellwerte, die ein weiteres Ausbreiten der Aktivierung vom gegebenen Knoten verhindern, falls deren Werte überschritten werden. Diese können zuvor vorgegeben werden oder durch den Algorithmus dynamisch bestimmt werden.

Einschränkungen sind oftmals mit einer Reduktion des Umfangs des zu durchsuchenden Graphen verbunden. Diese Reduktion bewirkt somit gleichzeitig eine Laufzeitverbesserung. Im vom Autor dieser Arbeit entwickelten Ansatz kommen die Einschränkung des Abstands, der ausgehenden Verbindungen und der Aktivierung zum Einsatz (siehe Kapitel 8 und 13). Mögliche Pfadeinschränkungen hängen von der gegebenen Ontologie ab. Ob diese eingesetzt werden ist individuell zu entscheiden. Einschränkungen des Abstands und der Aktivierung erwirken die angesprochene Reduktion des Umfangs. Eine Einschränkung der ausgehenden Verbindung betrifft im Rahmen dieser Arbeit die Data-Properties, da eine Übereinstimmung von Werten von Basisdatentypen keine semantische Assoziation im Sinne der Ontologie darstellt. Zudem besteht die Möglichkeit, den Verbindungen (Objectproperties) Kostenwerte beizufügen, die ebenfalls für eine Auswahl verwendet werden (siehe Kapitel 12).

7.7. Zusammenfassung

Zurückkommend auf den in Abschnitt 7.2 vorgestellten Prozess zur Referenzbestimmung, der auch in der untenstehenden Grafik 7.13 nochmals gezeigt wird, erfolgt zunächst in Phase 1 die Textanalyse, die in Abschnitt 7.3 vorgestellt wird. Diese identifiziert die Phrasen, *d.h.* Wortfolgen, die innerhalb des untersuchten Textes Namen von Entitäten repräsentieren. Aufbauend auf diesen erfolgt in Phase 2 die Identifikation der Ontologieinstanzen, deren Benamung mit den im Text erkannten Entitätsbezeichnern in Zusammenhang gebracht wird, *z.B.* über vollständige oder teilweise Überlappung der Zeichenfolgen. Die Entitäten innerhalb des Dokumentes bezeichnen die Aussage des Dokuments (siehe Postulat 7.9). Insofern birgt die Ermittlung der Satzaussage (Abschnitt 7.5) eine gleichzeitige Disambiguierung der

Entitäten. Jeder Bezeichner einer Entität definiert hierbei eine Gruppe im Steiner-Gruppen-Problem (Abschnitt 7.4.2). Hierbei enthält die jeweilige Gruppe die Intension des Bezeichners, *d.h.* die Adressen der oben genannten Instanzen. Im Bild wird die Ermittlung der Gruppe durch $f_{Intension}(B_i)$ dargestellt, während die in Phase 3 vorgenommene Auflösung des Gruppen-Problems in der rechten Spalte visualisiert ist. Die Lösung des Steiner-Gruppen-Problems erfolgt über einen Algorithmus zur Graphexploration, der den Resultatgraph, *d.h.* die Lösung des Steiner-Gruppen-Problems, mittels eines auf Spreading Activation (Abschnitt 7.6) basierenden Verfahrens ermittelt. Dieser Algorithmus wird im folgenden Kapitel 8 vorgestellt.

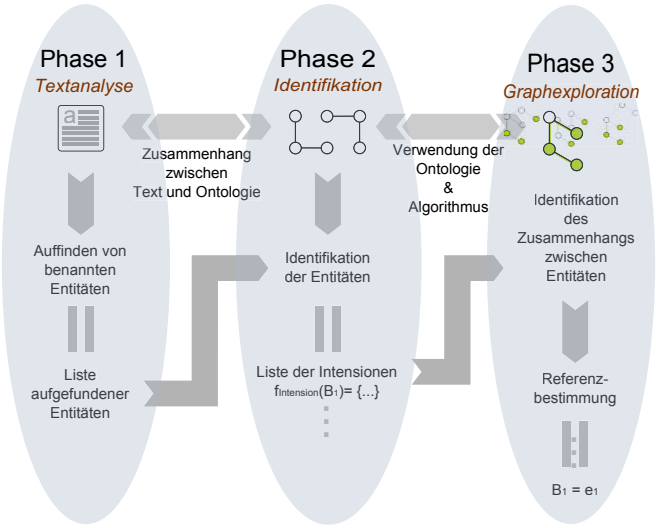


Abbildung 7.13.: Phasen des Prozesses zur Referenzbestimmung

8. Basisansatz

Bereits 1975 wies Collins auf die grundlegende Bedeutung des Verständnisses der menschlichen Sprache im Bereich von rechnergestützten Dienstleistungen durch die Aussage *„For information retrieval, and for social science, the implications of having a computer program able to reproduce the essentials of human understanding of language would seem to be of no small importance“* [49] hin. Um dieses zu ermöglichen muss zuvor eines der größten Hindernisse auf dem Weg zum Verständnis der menschlichen Sprache, die Mehrdeutigkeit von Begriffen, ausgeräumt werden.

Nachdem im Teil I der Dissertation die verschiedenen Arten von Ambiguität behandelt und in der bisherigen Ausführung von Teil II die technischen Grundlagen für die Umsetzung eines Algorithmus zur Referenzauflösung dargestellt werden, wird innerhalb des nun folgenden Kapitels der im Rahmen dieser Arbeit erstellte Basisansatz zur Referenzbestimmung mehrdeutiger Begriffe vorgestellt. Dieser Ansatz basiert auf der im vorausgehenden Kapitel vorgestellten Technik des Spreading Activation (Abschnitt 7.6). Der konkrete Zusammenhang von Spreading Activation und der Vorgehensweise innerhalb eines kognitiven Modells zur Auflösung von ambiguen Begriffen, *d.h.* der Referenzbestimmung, wird in Abschnitt 8.1 vorgestellt. Die approximative Bestimmung der Satzaussage $M(A)$ (siehe Postulat 7.9) in Form eines Steiner Baums (siehe Abschnitt 7.4.2) wird in Abschnitt 8.2 diskutiert. Der darauf folgende Abschnitt 8.3 bringt diese approximative Bestimmung in den Zusammenhang mit der Spreading Activation Technik, *d.h.* der grundlegenden Verfahrensweise des vorgestellten Ansatzes. In Abschnitt 8.4 wird das im Rahmen dieser Arbeit entwickelte Verfahren zur Disambiguierung mehrdeutiger Entitäten mittels Spreading Activation eingeführt.

8.1. Spreading Activation und Ambiguität

Die grundlegende Idee zur informationstechnischen Lösung des Problems der Referenzbestimmung mehrdeutiger Begriffe orientiert sich an der menschlichen Kognition, *d.h.* am Prozess der bei der Referenzbestimmung vom Menschen vollzogen wird.

Die Idee der Verwendung von Spreading Activation zur Disambiguierung mehrdeutiger Begriffe wird bereits von Deane 1988 [60] verfolgt. Wenngleich Deane keine konkrete Umsetzung der Disambiguierung und des Wissensmodells im Sinne eines Algorithmus beschreibt, bieten die von ihm vorgenommenen Untersuchungen Einsicht in eine theoretisch beschriebene, auf Spreading Activation basierende Vorgehensweise zur Referenzbestimmung. Er definiert hierzu zwei Spezialisierungen der referentiellen[1] Polysemie, die auch von Pethö [174] übernommen (siehe hierzu auch Abschnitt 5.2.5) und ursprünglich von Fauconnier [73] in Teilen beschrieben wurden. Dean und Pethö beschreiben hierbei die zwei Phänomene referentieller Polysemie. Zunächst wird der Fall der möglichen Referenzbestimmung (Closed Referential Polysemy) und anschließend der Fall, in welchem die Referenz nicht eindeutig bestimmt werden kann, beschrieben (Open Referential Polysemy) :

Postulat 8.1 (Closed Referential Polysemy[2])**.** *Durch die asymmetrische Verteilung der Aktivierung im konzeptuellen Netzwerk ist es immer möglich, die primäre Referenz, auf die sich das ambigue Wort bezieht, zu bestimmen. Gleichzeitig darf dem Wort nur die gleiche Bedeutung im Kontext des Textes zugewiesen sein.*[3]

Diese Form der Referenzbestimmung basiert auf der Möglichkeit Pfade, innerhalb des semantischen Netzwerkes anderen vorzuziehen und

[1] Der Ausdruck „referentielle Polysemie“ beschreibt die nicht eindeutig mögliche Zuweisung der korrekten Referenz zum polysemen Ausdruck.

[2] Definition in [174]: *„Closed referential polysemy derives from an asymmetric spread of activation in a conceptual network, which accounts for the facts that 1) with closed polysemy, it is always obvious which the primary, 'direct' referent of a word is and which is derived and 2) closed polysemy disallows crossed-sense anaphora (or more generally, leads to zeugma).“*

[3] Dean weist darauf hin, dass dies auch zu Zeugma transformiert werden kann. Unter Zeugma versteht man das Zusammenfügen zweier oder mehrere Teile eines Satzes mittels eines einfachen Verbs oder Nomens, *z.B.* „Er *nahm* seinen Mantel und seinen Hut“.

dadurch die primäre Referenz zu bestimmen, die als mögliche Bedeutungen des zu disambiguierenden Wortes sich auf dem bevorzugten Pfad befindet. Durch diese Pfadauswahl findet eine asymmetrische Aktivierung innerhalb des konzeptuellen Netzwerks statt, indem bestimmte Pfade anderen vorgezogen werden können. Deane und Pethö geben hierfür zwar Beispiele an, beschreiben jedoch nicht, wie diese Pfadauswahl technisch durchgeführt werden kann.

Postulat 8.2 (Open Referential Polysemy[4])**.** *Durch die symmetrische Verteilung der Aktivierung kann keine primäre Referenz bestimmt werden. Dadurch können für ein Wort auch nach dem Spreading noch mehrere Bedeutungen in Frage kommen. Dieser Zustand kann zum einen durch symmetrische konzeptuelle Relationen ausgehend von einem Knoten erreicht werden, zum anderen durch mindestens zwei asymmetrische Operationen, die sich im selben Knoten treffen.*

Bei dieser Art von Polysemie kann der primäre Referent nicht bestimmt werden. Dies geht darauf zurück, dass notwendige Informationen, welche die Bevorzugung von einzelnen Pfaden und somit einzelnen Referenzen ermöglichen würden, entweder nicht im zugrunde liegenden Text genannt oder aufgrund der Struktur des assoziativen Netzes nicht erreichbar sind. Somit findet eine symmetrische Verteilung der Aktivierung innerhalb des konzeptuellen Netzwerks statt.

Die Einordnung beider Arten von Polysemie im Verhältnis zur Homonymie, Eindeutigkeit und den anderen Arten von Polysemie wird von Pethö durch folgende Abbildung charakterisiert (siehe Abbildung 8.1):

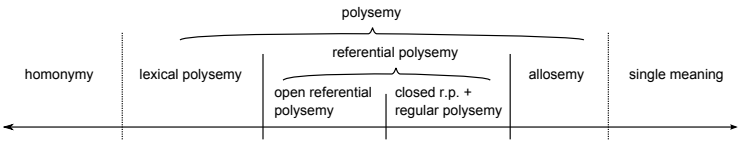


Abbildung 8.1.: Open und Closed Referential Polysemie (entnommen aus Pethö [174])

[4] Definition in [174]: *„Open referential polysemy on the other hand derives from a symmetric spread of activation, which accounts for the phenomena opposite to what has been observed with closed polysemy: 1) no primary referent can be established; 2) no zeugma arises."*

Pethö ordnet hierbei regulärer Polysemie (vgl. systematische Polysemie 5.2) *Closed Referential Polysemy* bei. Diese Zuordnung basiert darauf, dass mögliche asymmetrische Aktivierungsausbreitung auf das Umsetzen einer Regel(n) zurückgeführt werden kann. Dies bedeutet die Möglichkeit zur selektiven Pfadwahl anhand vorhandener Regeln, die Informationen zur erfolgreichen primären Referenzbestimmung bieten. Diese Zuordnung erfolgt nicht zu *Open Referential Polysemy*. Sondern erfolgt eine symmetrische Aktivierungsausbreitung und somit kann keine primäre Referenz aufgrund fehlender zusätzlicher Information innerhalb des Textes vorgenommen werden. Lexikalische Polysemie betrachten Deane und Pethö als nicht auflösbar, da die Aktivierungen von verschiedenen Pfaden ausgehend aufeinander treffen und es so zu Überlappungen von möglichen Bedeutungsvarianten kommt. Wenngleich Pethö referentielle von lexikalischer Polysemie trennt, so können die Bedeutungen lexikalisch polysemer Begriffe innerhalb des kontextuellen Rahmens eines Textes über die jeweiligen Referenzen miteinander in Beziehung gesetzt werden. Iraide Ibarretxe-Antuñano beschreibt dies durch ihre Aussage: *„They need the help of the semantic content of other lexical items in order to obtain those polysemous senses“* [116]. Somit sind diese nicht als vollständig getrennt zu betrachten.

Die Umsetzung der Referenzbestimmung anhand eines mathematischen Modells beziehungsweise einer algorithmischen, prozessorientierten, Vorgehensweise wurde von Pethö und Deane jedoch nicht entwickelt.

8.2. Grundlagen zur Bestimmung der Satzaussage

Die Orientierung an den kognitiven Prozessen des Menschen und somit der menschlichen Art und Weise der Auflösung von Mehrdeutigkeit setzt neben der Umsetzung der kognitiven Analyse auch die Feststellung eines Resultats voraus, *d.h.* eine Untermenge möglicher Referenzen. Zwar werden von Deane [60] auf Spreading Activation basierende Mechanismen im Kontext von referentieller Polysemie vorgestellt, jedoch keine algorithmische Vorgehensweise, die

es ermöglicht mittels eines durchgeführten Spreadings die im Kontext gemeinte Referenz zu bestimmen.

Basierend auf der deklarativen Wissensstruktur einer Ontologie wird in Abschnitt 7.5 die theoretische Vorgehensweise zur graphbasierten Referenzbestimmung vorgestellt. Im Postulat 7.9 wird das Konzept einer Satzaussage vorgestellt, *d.h.* die Bestimmung eines Teilgraphen basierend auf den Grundlagen der Zwei-Ebenen Semantik, der diese Aussage repräsentiert. Für jedes Wort eines Satzes wird die zugehörige Intension bestimmt. Je Intension muss im Resultatgraph mindestens eine Adresse enthalten sein. Der Resultatgraph zeigt die im Kontext des Satzes verwendete Bedeutung eines jeden Wortes auf und somit die gesamte Bedeutung des Satzes, *d.h.* die Satzaussage.

Diese Vorgehensweise birgt Ähnlichkeiten zu klassischen Suchalgorithmen, die ausgehend von einer Menge von Schlüssel- bzw. Suchbegriffen, Daten, die unter diesen Begriffen aufzufinden sind, zurückliefern. Auch die Bestimmung der referenzierten Adressen baut daher auf den Grundlagen einer solchen Suche auf. Jeffrey Yu, Lu Qin und Lijung Chang beschreiben in ihrer wissenschaftlichen Veröffentlichung [249] und ihrem Buch [248] theoretische Grundlagen zu Vorgehensweisen hinsichtlich *„Keyword Search in Relational Databases"*. Die Bestimmung eines Suchergebnisses basiert hierbei auf der Extraktion des korrekten Teilgraphen T innerhalb des Datengraphen G_D, der die Antwort auf eine Suchanfrage mit l Suchbegriffen birgt. Die Autoren beschreiben dies als „tree-based-semantics", *d.h.* innerhalb des Resultatgraphen T müssen Knoten $v_i \in V(T)$ existieren für die gilt, dass v_i für einen Suchbegriff k_i steht (für $1 \leq i \leq l$). Die Blattknoten dürfen nur diese Knoten darstellen, *d.h.* $leaves(T) \subseteq \{v_1, ..., v_l\}$. Siehe auch Abschnitt 7.4.2, in dem die grundlegenden Konzepte zur Teilgraphenbestimmung bereits eingeführt wird.

Yu, Qin und Chang unterscheiden verschiedene Rückgabestrukturen der Suchverfahren. Hierbei handelt es sich zum einen um Baumstrukturen und zum anderen um Teilgraphen, welche die Resultate der Suchanfrage repräsentieren. Auf Bäume als Rückgabestruktur wird im Folgenden genauer eingegangen. Zur Bestimmung der Güte des Resultats der Anfrage in einer Baumstruktur unterscheiden die Autoren zwei mögliche Kostenfunktionen[5] STBS (Definition 8.3) und

[5] *„Weights are assigned to edges to reflect the (directional) proximity of the corresponding tuple"* [249]

DRS (Definition 8.4) als Grundlage von Verfahren zur Teilgraphenbestimmung:

Definition 8.3 (Steiner Tree-Based Semantics[6] (STBS))**.** *Die zur Anwendung kommende Kostenfunktion $c_{STBS}(T) = \sum_{e \in T} c(e)$ berechnet die Gesamtkosten auf **ALLEN** innerhalb des Graphen vorkommenden Kanten. Die Kostenfunktion wird vom verwendeten Algorithmus vorgegeben.*

Definition 8.4 (Distinct Root Semantics (DRS))**.** *Anstatt der Berechnung der Kostenfunktion basierend auf allen Kantengewichten wird eine terminalspezifische Gewichtung eingesetzt. Die Kostenfunktion $c_{DRS}(T) = \sum_{i=1}^{l} dist(root(T), k_i)$ bestimmt jeweils das Gewicht hinsichtlich des kürzesten Abstands des jeweiligen Blattknotens k_i zur Wurzel $root(T)$ des Teilgraphen.*

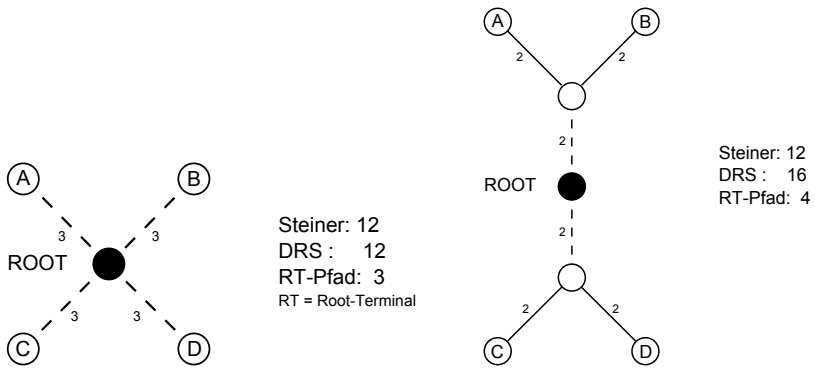
Die Bestimmung von Teilgraphen basierend auf einer Minimierung der in der Definition 8.3 vorgestellten Kostenfunktion $c_{STBS}(T)$ entspricht der in Abschnitt 7.4.2 vorgestellten Definition eines Steinerbaums 7.7. Das Problem der Bestimmung des optimalen Steinerbaums ist der Komplexitätsklasse **NP** zugeordnet (vgl. [119]). Zugleich entspricht die Bestimmung einem Aufwand von $O(2^m)$ (m bezeichnet die Anzahl der Kanten), da eine exponentielle Anzahl an möglichen Bäumen berechnet werden muss. Wird der Teilgraph jedoch mittels der Minimierung der Kostenfunktion $c_{DRS}(T)$, *d.h.* der Distinct Root Semantics (Definition 8.4), bestimmt, so birgt dies einen Aufwand von $O(l(n \log n + m))$, mit $n = |V(T)|$ und n möglichen Bäumen, *d.h.* jeder Knoten kann als potentielle Wurzel eines Baumes betrachtet werden (vgl. [249]).

In Anbetracht der unterschiedlichen Komplexitäts- und Aufwandswerte wird deutlich, dass algorithmische Verfahren, die eine Lösung gemäß des Prinzips der Steiner-Tree-Based-Semantics suchen, geringere Effizienz aufweisen. Hingegen ermöglicht die Teilgraphenbestimmung mittels der Kostenfunktion der Distinct-Root-Semantic eine effizientere Bestimmung der Näherungslösung. Bei dieser handelt sich jedoch um eine Näherungslösung im Vergleich zur „optimalen" Lösung eines Steinerbaums. Dieser Zusammenhang zwischen STBS

[6] Der Begriff „Semantics" wird von den Autoren nicht näher bestimmt. Der Autor dieser Arbeit übernimmt diesen Begriff, um den korrekten Bezug zu den ursprünglichen Autoren zu gewährleisten.

und DRS wird dadurch deutlich, dass im Falle von kantendisjunkten Verbindungen zu den gesuchten Begriffen beide Kostenfunktionen übereinstimmen, *d.h.* $c_{STBS} = c_{DRS}$ (siehe Abbildung 8.2(a)). Jedoch für Graphen, die dieses kantendisjunkte Merkmal nicht aufweisen, gilt $c_{STBS} < c_{DRS}$ (siehe Abbildung 8.2(b)). Somit gilt:

Satz 8.5. *Der Zusammenhang zwischen Steiner Tree-Based Semantics und Distinct Root Semantics ist durch* $c_{STBS} \leq c_{DRS}$ *gegeben.*



(a) Kantendisjunkte Blattknotenverbindungen

(b) Kantenüberlappende Blattknotenverbindungen

Abbildung 8.2.: Zusammenhang Steiner-Tree-Based und Distinct-Root-Semantics

Im Kontext dieses Zusammenhangs ist ebenfalls auf die Optimierung des Resultats in Verbindung mit einer Minimierung der Kostenfunktion hinzuweisen. Somit kann davon ausgegangen werden, dass bei Distinct Root Semantics der Teilgraph T mit den minimalen Kosten $\min(c_{DRS}(T))$ eine Näherung beziehungsweise gegebenenfalls eine exakte Übereinstimmung zum gesuchten Steinerbaum darstellt.

Im Zusammenhang mit der zuvor angesprochenen Bestimmung der Satzaussage ist darauf hinzuweisen, dass sich zusätzlich die Komplexität des Steiner-Gruppen-Problems erhöht, da es sich hier um Mengen handelt, die eine Selektion des zu verwendenden Blattknotens (*d.h.* der Adresse) je Intension des jeweiligen Suchbegriffs

erfordern. Hierbei kann eine Näherung über die Verwendung von Distinct-Root-Semantics erzielt werden.

Zwischen Verfahren zur Bestimmung der Satzaussage und der von Yu et al. beschriebenen Suche, *d.h.* zwischen den Suchalgorithmen basierend auf Schlüsselworten und denen zur Disambiguierung, besteht jedoch ein wesentlicher Unterschied. Im Zusammenhang mit einem vorliegenden Satz besteht bei der Disambiguierung die Möglichkeit des Zugriffs auf Kontextinformationen innerhalb des Satzes zusätzlich zu den ambiguen Begriffen. Auf diese impliziten Informationen kann zur Disambiguierung ebenfalls zurückgegriffen werden.

8.3. Bestimmung von Steinerbäumen mittels Spreading Activation

In Kapitel 7.6 wird die Methode Spreading Activation zur Simulation menschlicher kognitiver Prozesse vorgestellt. Es wurde des Weiteren auf die Verwendung von markerbasierten Spreading Methoden auf Grundlage der von Hendler [111] entwickelten Technik hingewiesen und insbesondere die Definition 7.11 des Digital Spreading vorgestellt, die auf der Verwendung von Markern basiert und ein prinzipielles Vorgehen zur Bestimmung von Lösungen vorgibt.

Die Verwendung von Markern heftet ausgewählten Knoten eine Information an. Diese Information wird in jedem Spreading Schritt weitergegeben und die zur Information assoziierte Aktivierung gemäß der Formel des Spreading Activation Algorithmus angepasst. Meist impliziert diese einen Decay-Parameter und somit eine distanzgebundene Abschwächung des Aktivierungswerts. Basierend auf der iterativ (*d.h.* im Pulstakt und Activation Decay; siehe Abschnitt 7.6.2) erfolgenden Weitergabe von markerspezifischer Information können Aktivierungswerte verschiedener Marker innerhalb eines Knoten zusammenkommen. Somit verfügt dieser Knoten über die Information der Marker und den diesen zugeordneten Aktivierungswerten.

Überträgt man diesen Vorgang auf das Problem der Steinerbaum-Bestimmung basierend auf einer initial gegebenen Menge von Entitätsbezeichnern, so wird zunächst ein Marker $m_i \in M$ für jeden dieser

Bezeichner individuell erstellt und dieser initial dem Knoten, der eine Entität mit einem solchen Bezeichner repräsentiert und somit jeder Adresse in der Intension des Bezeichners zugewiesen. Hierbei findet auch eine initiale Zuweisung des zugehörigen Aktivierungswerts für den Marker statt.[7] Im Laufe des Spreading werden die markerspezifischen Aktivierungen weitergegeben. Treffen im Verlauf des Spreadingvorgangs alle Marker in einem Knoten aufeinander, *d.h.* der Knoten besitzt Aktivierungswerte für jeden gegebenen Marker, so repräsentiert dieser eine mögliche Lösung in Form eines Teilbaumes, dessen Wurzel durch diesen Knoten repräsentiert wird. Ein solcher Teilbaum wird mittels der Kostenfunktion $c_{SP}(T) : \sum_{i=1}^{|M|} a_{m_i}$ beurteilt, welche die Summe der markerspezifischen Aktivierungenswerte, die dem Wurzelknoten zugeordnet werden, berechnet (siehe Abschnitt 7.6.2; Kriterium für Path-Evaluator). Die Weitergabe einer Markeraktivierung vom Ursprungsknoten hin zu diesem Knoten beschreibt den Verbindungspfad (Funktion $pfad$, siehe Def. 7.5) zwischen diesen Knoten. Bei der Weitergabe der Aktivierung des Markers m_i zwischen zwei benachbarten Knoten v und g wird die Distanz durch die Verwendung eines Decay-Faktors und der zusätzlichen Gewichtung der Kante berücksichtigt (*z.B.* $a_{g,i} = a_{v,i} \cdot d \cdot w_{v,g}$; Decay-Faktor d). Insofern hängt der markerspezifische Aktivierungswert eines Knotens g ab vom Abstand zwischen diesem Knoten und dem Knoten der eine Intension des Markers repräsentiert. Die Funktion $dist(root(T), k_i)$ beschreibt die Distanz vom Wurzelknoten ausgehend zum Intensionsknoten des Markers k_i bei der Rückverfolgung der höchsten Aktivierungswerte bzgl. dieses Markers. In der Konsequenz resultiert daraus: $\min(c_{DRS}(T)) \widehat{=} \max(c_{SP}(T))$. Ein algorithmisches Suchverfahren mittels Spreading Activation ist somit Distinct-Root-Semantics zuzuordnen und somit gleichzeitig als Verfahren zur approximativen Bestimmung des das optimale Resultat beschreibenden Steinerbaumes zu betrachten. Hierbei gilt: Je höher die Gesamtaktivierung des Rootknotens, desto besser die approximative Näherung (vgl. vorheriger Abschnitt 8.2).

Wird diese Vorgehensweise ohne zusätzliche algorithmische Modifikationen ausgeführt, so führt diese „a breadth-first search for shortest paths on a relational network“ (Definition 7.11) durch. Diese

[7] Dieser ist abhängig von der Zuordnung des Bezeichners zum Ontologieelement und gegebenenfalls weiterer textueller Gegebenheiten im zugrundeliegenden Text (siehe Kapitel 12 zur Bestimmung des initialen Akivierungswertes).

M	Menge der Marker für alle Entitätsbezeichner
$c_{SP}(T)$	Summe der markerspezifischen Aktivierungen im Teilbaum T
d	Decay-Faktor
$w_{v,w}$	Kantengewicht zwischen Knoten v und Knoten w
$dist(root(T), k_i)$	Distanz zwischen Wurzel des Teilbaums T und Intension des Markers k_i bei Rückverfolgung des Pfads der höchsten Aktivierung.
$c_{DRS}(T)$	Definition 8.4

Tabelle 8.1.: Variablen zur Bestimmung von Steinerbäumen mittels Spreading Activation

impliziert eine Äquivalenz zur Suche nach dem Prinzip des Breadth-First Algorithmus von Dijkstra [51, S. 595]. Dieser Zusammenhang ist anhand der Speicherung des maximalen Aktivierungswerts per Marker im jeweiligen Knoten ersichtlich, welches dem Prinzip der Speicherung des minimalen Abstands von Djikstra folgt. Dies setzt voraus, dass alle Knoten ihre Aktivierungen innerhalb eines Pulses weitergeben und kein Priorisieren von Knoten stattfindet.[8] Ein Beispiel eines solchen Spreading Vorgangs ist in Abbildung 8.3 gegeben.

Der Ausgangsgraph ist durch die grauen Kanten dargestellt. Ziel ist es Referenzen für die Bezeichner a, b, c zu finden. Im Beispiel wurde nur eine Intension je Bezeichner verwendet. Auf der rechten Bildhälfte ist die Bestimmung des Steinerbaums mittels Spreading Activation dargestellt. Die Aktivierungsweitergabe (Spreading) wurde bereits vollständig ausgeführt und alle berechneten Aktivierungswerte sind in der rechten oberen Abbildung dargestellt. Der Wert $\max(c_{SP}(T))$ wird für den Teilbaum mit der Wurzel r mit der Gesamtaktivierung von $a_r = 2,4$ erzielt. Zusammen mit der Rückverfolgung der Marker (*d.h.* je Bezeichner die Intension mit dem kürzesten Abstand und der höchsten Aktivierung) ergibt sich der in der rechten unteren Bildhälfte dargestellte Resultatgraph.

[8] Der Vergleich mit Djikstra setzt eine systematische Verteilung der Aktivierung voraus. Der Einsatz eines Priorisierens und somit asymmetrischer Verteilung der Aktivierung widerspricht diesem. Eine Beschreibung von symmetrischer und asymmetrischer Aktivierung ist in Abschnitt 8.1 enthalten.

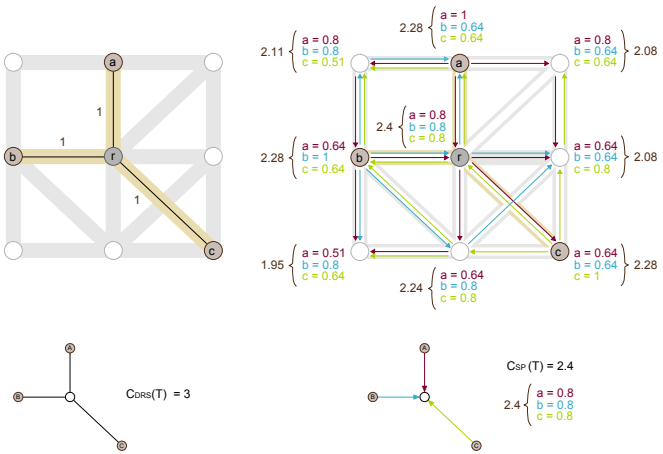


Abbildung 8.3.: Beispielsuche mit Hilfe von Spreading Activation

Die Berechnung des Steinerbaums (siehe Abschnitt 7.7) mittels Distinct Root Semantics ($c_{DRS}(T) = 1 + 1 + 1 = 3$) ist in der linken Bildhälfte dargestellt. Hierbei ergibt sich eine Übereinstimmung des bestmöglichen Resultgraphs mittels Distinct Root Semantics und dem Spreading Activation Ansatz. In diesem Beispiel ist ein Kantengewicht von 1 und ein Decay-Faktor von 0,8 gegeben. Ein Verlust dieser Übereinstimmung kann bei der Verwendung anderer Kantengewichtungen erfolgen. Für das Beispiel wurde 1 als Gewichtungswert für jede Kante verwendet.

8.4. Algorithmus

Zunächst wird in die allgemeine Vorgehensweise des Algorithmus in Abschnitt 8.4.1 eingeführt, bevor der Algorithmus selbst in Abschnitt 8.4.2 vorgestellt wird. In Abschnitt 8.4.3 wird dieser anschließend zusammengefasst. Die Problematik der nicht zusammenhängenden Teilbäume wird in Abschnitt 8.4.5 besprochen und im Anschluss daran der Algorithmus an einem Beispiel in Abschnitt 8.4.6 erklärt.

8.4.1. Textuelle Analyse

Der Disambiguierung selbst geht die textuelle Analyse voraus, die es ermöglicht, ontologiespezifische Entitätsbezeichner (*z.B.* Bezeichner von Konzepten, Instanzen *etc.*) im Text zu erkennen (siehe hierzu Abschnitt 7.3). Eine Erkennung aller ontologischen Merkmale ermöglicht eine kontextbezogene Übertragung über verschiedene Informationsmedien hinweg, *d.h.* textuell repräsentiertes Wissen wird anhand einer Auswahl von Merkmalen (*d.h.* den erkannten Bezeichnern) auf das in einer Ontologie modellierte Wissen übertragen. Eine direkte[9] Vorgehensweise, um diese beiden Welten zu vereinen, ist die Suche nach Ontologieelementen innerhalb des Wörterbuchs der Ontologie, die namentlich mit den Bezeichnern übereinstimmen (vgl. Abschnitt 7.4). Die Vorgehensweise des Algorithmus basiert auf dem Prinzip der Zwei-Ebenen Semantik, die in Abschnitt 6.2.1 eingeführt wurde und hier zum besseren Verständnis der algorithmischen Vorgehensweise nochmals besprochen wird. Zunächst wird für einen durch die textuelle Analyse erkannten Bezeichner die zugehörige Intension bestimmt ($f_{Intension}(B)$), *d.h.* die Adressen (Seme) der Instanzen in der Ontologie. Ambiguität definiert sich hierbei aus Intensionen, die mehr als eine Adresse beinhalten. Im Rahmen der konzeptuellen Analyse sind alle Intensionen relevant. Dies impliziert ebenfalls eindeutige Bezeichner-zu-Ontologieelement-Zuordnungen, *d.h.* Intensionen, die nur eine Adresse beinhalten, da diese Informationen zur Bestimmung des Kontextumfelds von Adressen anderer Intensionen von Belang sind. Die konzeptuelle[10] Einbettung eines Ontologieelements innerhalb der Ontologie wird als dessen Extension bezeichnet. Dies ist in Abbildung 8.4 exemplarisch mit den Bezeichnern „A“, „B“ und „C“

[9] Die textuelle Analyse bietet eine Vielzahl von Möglichkeiten des Bezeichnervergleiches. So können Informationen, wie *z.B.* textuelle Distanzen zwischen Parametern, Informationsgewinnung mit zusätzlichen Lexika, indirekte Referenzen über Pronomina, satzspezifische Aussagen durch lokale Kontextbeschränkung *etc.* zur Gewichtung der Güte der Bezeichnerübereinstimmung verwendet werden. Diese Gewichtung kann zur Berechnung einer individuellen initialen Bezeichneraktivierung für den aktuellen Knoten verwendet werden. Die zuvor genannten Informationsauswertungen gestaltet sich jedoch schwieriger als der direkte, *d.h.* zeichenbasierte, Wortvergleich.

[10] „Konzeptuell“ bezieht sich hierbei nicht auf die Schema-Ebene einer Ontologie sondern auf die relationalen Zusammenhänge der Instanzen innerhalb der vorgegebenen Ontologie.

dargestellt. Der obere Teil der Abbildung zeigt zunächst die einzelnen Extensionen der Adressen separat voneinander.

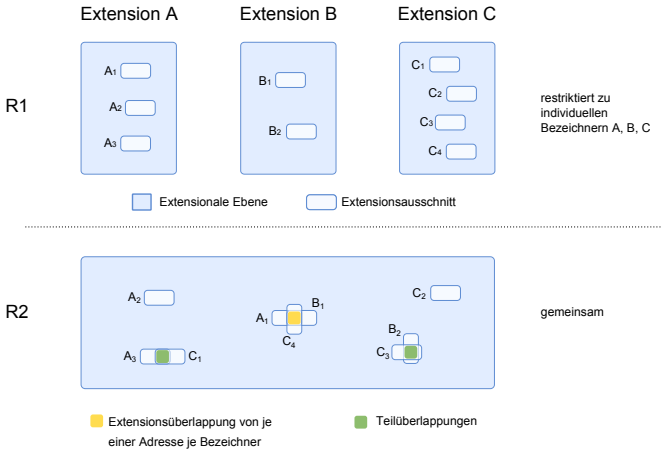


Abbildung 8.4.: Darstellung der Extensionen bzw. der Extensionsüberschneidungen

Der vorliegende Ansatz versucht die in Abschnitt 6.2 vorgestellte Überlappung der Extensionen (vgl. unterer Teil der Abbildung 8.4) zur Referenzbestimmung der mehrdeutigen Begriffe zu nutzen. Übertragen auf eine graphbasierte Darstellung einer Ontologie wird eine Extensionsüberlappung zweier Instanzen durch die Existenz eines Pfades zwischen diesen ausgedrückt. Demzufolge wird eine Extensionsüberlappung mehrerer Instanzen durch einen Teilbaum inmitten des Ontologiegraphen repräsentiert.

Das **Ziel** des Algorithmus ist die Bestimmung des Wurzelknotens gemäß der vorgestellten Teilgraphenbestimmung basierend auf der Distinct-Root-Semantic (siehe Abschnitt 8.2). Diese ist zugleich eine Näherung an die kostenminimale Wurzel des Steinerbaumes. Wie im vorhergehenden Abschnitt 8.3 gezeigt, ist hierbei im Zusammenhang der Spreading Activation Technik die zugewiesene Aktivierung ausschlaggebendes Kriterium für die Bewertung des Wurzelknotens.

Variable	Bedeutung
Q	Nach absteigender Aktivierung geordnete Liste der zu prozessierenden Knoten
Y	Liste der prozessierten Knoten
I	Menge der Intensionen
$depth_u$	Explorationstiefe Knoten u
Int_l	Intension für den Entitätsbezeichner l
$P_{u,l}$	Der erste Knoten auf dem Pfad, der hinsichtlich einer Referenz (Adresse in der Intension) des Bezeichners l ausgehend vom Knoten u am besten bewertet wurde
a	Aktivierung
l	Bezeichner l
L	Liste der Bezeichner
P_u	Prozessierte Knoten, die vom Knoten u aus erreichbar sind
w	Zählvariable zur Bestimmung des Abbruchkriteriums $maxDeviance$

Tabelle 8.2.: Variablen des Algorithmus

8.4.2. Basisalgorithmus

Die semantische Datengrundlage des Algorithmus (siehe Alg. 2, 3, 4, 5 und Tabelle 8.1) basiert auf dem Instanzgraphen $\mathcal{G}$ einer RDF-Ontologie (siehe Abschnitt 7.3), welcher das im Kontext der Disambiguierung notwendige Hintergrundwissen repräsentiert.[11] Die innerhalb dieses Graphen repräsentierten Zusammenhänge und Beziehungen zwischen Instanzen beschreiben den konkreten Zustand der durch die Ontologie repräsentierten Domäne. Nach erfolgter Identifikation der initialen Entitätsbezeichner durch die textuelle Analyse wird für jeden Bezeichner $l \in L$ mittels $f_{Intension}(l) = Int_l$ die zugehörige Intension Int_l bestimmt. $I = \cup_{l \in L} Int_l$ beschreibt somit die Gesamtmenge der initial verfügbaren Information im Rahmen des

[11] Terminologische bzw. schematische Zusammenhänge in Verbindung mit dem Schemagraphen (siehe Abschnitt 7.2) können unabhängig davon in der Voranalyse zur Gewichtung von Instanzinformationen Anwendung finden und somit Informationen des Instanzgraphen hinsichtlich algorithmischer Verwertung beeinflussen.

gegebenen konzeptuellen Umfeldes. Jeder einen Bezeichner repräsentierende Knoten $v \in Int_l$, *d.h.* jede Adresse, definiert den Ursprung einer Extension, *d.h.* jeder Teilgraph, der diesen Knoten enthält, kann als eine von dessen möglichen Extensionen betrachtet werden. Dies ist exemplarisch im oberen Part der Abbildung 8.5 dargestellt. Zu überprüfen sind nun mögliche Steinerbäume innerhalb der Ontologie, die mindestens eine Adresse je Bezeichner enthalten Der untere Part der Abbildung 8.5 visualisiert solche Extensionüberlappungen ausgehend vom Aufeinandertreffen einzelner Extensionen, die durch Pfade zwischen den Adressen und dem sich daraus ergebenden Teilgraphen repräsentiert werden.

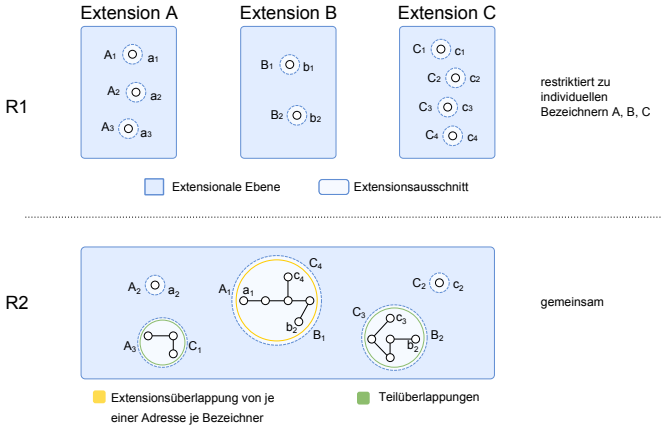


Abbildung 8.5.: Überlappung der Extensionen auf Grundlage eines Ontologiegraphen

Initialisierung Nach der oben vorgestellten Adressbestimmung (Seme), also der Identifikation der Knoten $v \in Int_l$, die im Rahmen des Algorithmus von besonderer Bedeutung sind. Desweiteren erfolgt die Zuweisung einer initialen, dem repräsentierten Bezeichner $l \in L$ zugeordneten Aktivierung $a_{v,l}$ je Knoten. Diese *initiale* Aktivierung wird von zwei Faktoren beeinflusst. Zum einen erfolgt eine Bewertung der linguistischen Übereinstimmung des textuellen und des ontologischen Bezeichners und zum anderen wird ein knotenspezifisches

Maß[12] verwendet. Darauf folgend wird das Distanzmaß des der Intension Int_l zugeordneten Knotens mit $d_{v,l} = 0$ initialisiert, da jeder dieser Knoten selbst den Bezeichner l repräsentiert. Weiterhin besitzt jeder Knoten v eine zugeordnete Menge $P_{v,l}$, die jeweils den nächsten Knoten auf dem aus der Sicht des Algorithmus optimalen[13] Pfades bzw. der optimalen Pfade (bei gleicher Bewertung) zu einer Adresse des Bezeichners l beschreibt.[14] Zu Beginn sind dieser Menge keine Knoten zugeordnet.

Algorithmus 2 : Basisalgorithmus Teil 1 (Initialisierung und Prozessablauf)

```
1  Initialisation Q ← L; Y = ∅; R = ∅; ∀u ∈ I : depth_u = 0;
2  ∀l, ∀u ∈ I : if u ∈ Int_l then d_{u,l} ← 0, P_{u,l} ← u else d_{u,l} ← ∞, P_{u,l} ← ∅,
   a_max = 0, depth = 0, P_u ← ∅, w = 0, act = 0;
3  while Q is non-empty do
4  |  get node v, with highest overall activation, from Q and add v to Y;
5  |  if IS-FULLCONNECTOR(v) then ADDRESULT(v);
6  |  foreach (u, e) ∈ GETPATHSTEPSOFINTEREST(v) do
7  |  |  ANALYSECONNECTION(v, e, u);
8  |  |  if ((u ∉ Y) and (a_u > a_min) and (depth_u < depth_max)) then
9  |  |  |  add v to Q with depth_u = depth_v + 1;
10 |  |  end
11 |  end
12 |  depth = depth + 1;
13 end
```

Im Gegensatz zu der in Abschnitt 8.3 vorgestellten symmetrischen und gleichzeitigen Exploration erfolgt eine sukzessive priorisierte, *d.h.* asymmetrische, Exploration basierend auf der Gesamtaktivierung a_v, die dem einzelnen Knoten v zugeordnet ist. Alle Knoten innerhalb der Vereinigungsmenge der Intensionen $v \in I$ werden zunächst in die Queue Q eingefügt, ($Q \leftarrow I$). Diese ist absteigend geordnet nach den Gesamtaktivierungen der innerhalb der Liste enthaltenen Knoten, *d.h.* der Knoten, der die maximale Gesamtaktivierung $\max(a_v)$ besitzt, befindet sich an der Spitze der Liste.

[12] Dieses beschreibt die Einbettung des Ontologieelements, welches der Knoten repräsentiert, in die Ontologie (siehe Kapitel 12).

[13] Je nach Variante beschreibt entspricht „optimal“ entweder der kürzesten Distanz oder dem höchsten Aktivierungswert.

[14] Für einen gegebenen Pfad $u - n - m - x$ besitzen die Knoten demzufolge die Zuweisungen $P_{u,l} = \emptyset, P_{n,l} = u, P_{m,l} = n$ und $P_{x,l} = m$.

Prozessbeginn Nach Beendigung dieser initialen Vorbereitungsphase kann der Algorithmus gestartet werden und somit die Verteilung der Aktivierung beginnen. Diese erfolgt iterativ durch die Selektion des jeweils am höchsten aktivierten Knotens $v \in Q$. Die Weitergabe der Aktivierung folgt dem von Crestani im Abschnitt 7.6.2 vorgestellten Ablaufschema für Spreading-Activation-Algorithmen. Zunächst werden in der *Vorbereitungsphase* alle mit dem Knoten v in Beziehung stehenden Knoten $u \in V$ durch die Funktion GETPATHSTEPSOFINTEREST(v) erfasst.

Algorithmus 3 : Basisalgorithmus Teil 2 (Pfadselektion)

```
14 Func GetPathStepsOfInterest(v)
15 |   pathSteps ← ∅;
16 |   foreach (u,e) ∈ incoming(v) ⋃ outgoing(v) do
17 |   |   if ((degree(e) ≤ deg_max)||(u ∈ (Q ∪ Y ))) then
18 |   |   |   add (u,e) to pathSteps;
19 |   |   end
20 |   end
21 |   return pathSteps;
22 end
```

Diese Funktion bestimmt die direkt mit dem Knoten v verbundenen Knoten $u \in U$ ($U \subset V$), *d.h.* die Distanz zu jedem der Knoten ist minimal, *d.h.* $d_{v,u} = 1$. Hervorzuheben ist, dass die Richtung der Kanten vernachlässigt wird und daher eine symmetrische Kantenrelation gilt[15] ($E = E^{-1}$) (siehe ebenfalls Abschnitt 7.5). Nach der Selektion der für die Spreadingphase benötigten Zielknoten wird jede Verbindung durch die Funktion ANALYSECONNECTION(v, e, u) individuell analysiert.

Funktion AnalyseConnection(v, e, u) Die Bezeichnerinformationen, die im Ursprungsknoten v und im Zielknoten u vorhanden sind, werden mittels der Funktion ANALYSECONNECTION(v, e, u) untersucht. Im Basisansatz erfolgt ein unidirektionaler Abgleich der Bezeichnerinformationen. Dieser Abgleich impliziert die mögliche *Weitergabe der Aktivierung* innerhalb einer iterativen Untersuchung, bei der zunächst

[15] Im Algorithmus wird dies durch die Selektion der eingehenden $incoming(v)$ und ausgehenden Relationen $outgoing(v)$ ausgedrückt.

Algorithmus 4 : Basisalgorithmus Teil 3 (Kantenanalyse)

```
23 Func ANALYSECONNECTION(v, e, u)
24     foreach identifier l ∈ L do
25         if the activation spreaded from v to u for l is greater than a_{u,l} then
26             update a_{u,l} with this new activation;
27             ACTIVATIONUPDATE(u, l);
28         end
29         if d_{v,l} + 1 ≤ d_{u,l} then
30             if d_{v,l} + 1 < d_{u,l} then
31                 P_{u,l} ← ∅;
32                 d_{u,l} = d_{v,l} + 1;
33                 add v to P_{u,l};
34                 if IS-FULLCONNECTOR(u) then
35                     ADDRESULT(u);
36                 end
37                 COSTUPDATE(u, l);
38             end
39             else
40                 foreach p ∈ P_{u,l} do
41                     if a_p > act then act = a_p;
42                     if a_v > a_p then remove p from P_{u,l};
43                 end
44                 if a_v ≥ act then
45                     add v to P_{u,l};
46                 end
47             end
48         end
49     end
50     add v to P_u;
51 end
```

die mögliche zu übertragende Aktivierung hinsichtlich des aktuell untersuchten Bezeichners zu der im Zielknoten vorhandenen verglichen wird. Falls ein höherer Aktivierungswert erzielt werden kann, wird die Aktivierung weitergegeben (ACTIVATIONUPDATE(u, l), siehe Algorithmus 5). Anschließend wird diese Änderung der Aktivierung in der *Nachbereitungsphase* mittels Rückwärtspropagierung[16]

[16] Die Rückwärtspropagierung erfolgt iterativ über die Verknüpfung der Vorgängerknoten, die in der Menge $P_{u,l}$ abgespeichert sind. Das Prinzip der Nachbereitungsphase wird in Abschnitt 7.6.2 vorgestellt.

jedem Vorgängerknoten mitgeteilt. Dieser besitzt, falls diese in der Zwischenzeit nicht durch eine andere, bei einer vorhergehenden Exploration neu hinzugekommenen Verbindung erhöht wurde, eine geringere Aktivierung hinsichtlich des Bezeichners, als die, die durch den gerade aktualisierten Knoten jetzt weitergegeben werden kann. Nach der Überprüfung des Aktivierungswerts erfolgt die Überprüfung der Distanz zur nächsten Adresse des Bezeichners. Falls die dem Zielknoten aktuell zugewiesen Distanz größer ist als die mittels des Pfades über den Ursprungsknoten hinweg, werden die dem Zielknoten zugeordneten Pfade in $P_{u,l}$ vollständig entfernt (Algorithmus 4, Zeile 31).[17] Ebenfalls erfolgt die Zuweisung eines neuen Distanzwertes. Die Zuweisung von neuen, zuvor unbekannten Informationen bezüglich eines neuen Bezeichners $l \in L$ kann darauf hinweisen, dass der Knoten womöglich über einen Aktivierungswert und somit einen Pfad zu mindestens einem Sem (einer Adresse) je Bezeichner verfügt und somit eine Lösung in Form eines Steinerbaums repräsentiert, dessen Wurzelknoten er ist. Dies wird mit der Funktion IS-FULLCONNECTOR(u) überprüft. Falls dies der Fall ist, wird die mögliche Lösung mittels der Funktion ADDRESULT(u), dem Äquivalent des in Abschnitt 7.6.2 vorgestellten *Path-Evaluator*[18], auf eine mögliche Verwertung hin untersucht. Diese Funktion überprüft zunächst, ob die Gesamtaktivierung des gegebenen Knoten a_v höher ist als die gespeicherte maximale Aktivierung a_{max} aus der Menge der zuvor untersuchten Lösungen. Ist dies der Fall, so wird die Lösung abgespeichert. Ebenfalls enthalten ist die Abbruchbedingung des Algorithmus (*Phase: Abbruchbedingung prüfen*). Ziel des Algorithmus ist eine Optimierung des durch den Knoten repräsentierten Wertes gemäß der Distinct-Root-Semantic, *d.h.* im Rahmen des Algorithmus durch dessen Gesamtaktivierung a_v. Der Algorithmus stoppt, falls insgesamt eine Anzahl von *maxDeviance* Unterschreitungen des a_{max} Wertes auftreten und somit davon auszugehen ist, dass keine weitere Optimierung innerhalb einer angemessenen Laufzeit erreicht werden kann. Der zu diesem Zeitpunkt durch den Knoten o mit der maximalen Aktivierung im Set R, der

[17] Dies trifft nicht für die Algorithmusvariante zu, die eine Fokussierung auf die Aktivierungswerte vornimmt (siehe Abschnitt 8.4.4).

[18] Der Path-Evaluator nimmt eine Gesamtevaluierung hinsichtlich des Knoten vor (siehe Abschnitt 7.6.2). Diese ist von der Überprüfung der den einzelnen Bezeichnern zugeordneten Parametern innerhalb der Funktion ANALYSECONNECTION(v, e, u) zu trennen.

Menge der gespeicherten Resultate, repräsentierte Lösungsbaum beschreibt das Resultat des Algorithmus. Unter der Voraussetzung, dass das Abbruchkriterium nicht erfüllt wurde, erfolgt eine Weitergabe der zuvor ermittelten Änderung des Distanzwertes $d_{u,l}$ (Funktion COSTUPDATE(u, l)). Dies betrifft alle Knoten, die vom aktuellen Knoten u während des Algorithmus zuvor exploriert wurden. Für diese Knoten bedeutet die festgestellte Distanzreduktion gleichzeitig eine mögliche Verbesserung des ihnen bisher zugewiesenen Distanzwertes.

Sofern der Distanzwert nicht niedriger, sondern gleich dem bereits zugewiesenen Wert ist, werden alle u zugeordneten Vorgängerknoten $p \in P_{u,l}$ auf dem Pfad zum Repräsentant des Bezeichners l untersucht. Besitzen diese Vorgänger eine geringere Gesamtaktivierung $a_v > a_p$, so werden diese aus der Menge $P_{u,l}$ entfernt. Dies ist darauf zurückzuführen, dass die Priorisierung innerhalb aller Varianten auf der Aktivierung in Kombination mit dem Distanzwert basiert. Bei geringerem sowie gleichem Distanzwert werden die für den Bezeichner gespeicherten Pfade des Ursprungsknotens durch diesen erweitert (Algorithmus 4, Zeile 33 bzw. 45).

Mit dem Abschluss der Untersuchung aller bezeichnerspezifischen Merkmale der Knoten v und u beginnt die Abwägung über die weitere Verwendung des Knotens u innerhalb des Algorithmus. Der Algorithmus besitzt zum Einen das Set Q, das die explizit zu explorierenden Knoten enthält und zum Anderen das Set Y, in dem die bereits explorierten Knoten vermerkt sind. Knoten der Menge Y können nur implizit innerhalb der Methoden COSTUPDATE(u, l) und ACTIVATIONUPDATE(u, l) aktualisiert werden. Der aktuell untersuchte Knoten u wird daher überprüft, ob er bereits analysiert wurde und die Beschränkungen hinsichtlich minimaler Aktivität a_{min} und maximaler Distanz d_{max} erfüllt. Gegebenenfalls wird er nach dieser Beurteilung der Menge Q hinzugefügt. Zudem erfolgt die Bestimmung der Explorationstiefe des Knoten u ($depth_u = depth_v + 1$).

8.4.3. Zusammenfassung des Algorithmus

Mit dem vorgestellten Algorithmus wird das Ziel verfolgt, die im Kontext des gegebenen Dokuments enthaltenen Bezeichner ihren korrekten Semen innerhalb der ontologischen Wissensstruktur zuzuordnen, *d.h.* den Adressen innerhalb der jeweiligen bezeichnerspezifischen Intension, die welche die korrekte Referenz des Bezeichners darstellt.

Algorithmus 5 : Basisalgorithmus Teil 4 (Zusatzfunktionen)

```
52 Func CostUpdate(u, l)
53 |  propagate change in cost d_{u,l} to all its reached ancestors p ∈ P_u;
54 end
55 Func ActivationUpdate(u, l)
56 |  propagate change in activation a_{u,l} to all its reached ancestors
   |  p ∈ P_u;
57 end
58 Func AddResult(v)
59 |  if a_v ≥ a_max then
60 |  |  add v to R based on the amount on its activation a_v;
61 |  |  a_max = a_v;
62 |  end
63 |  else
64 |  |  w = w + 1;
65 |  end
66 |  if w = maxDeviance then // stop algorithm
67 |  |  return o ∈ R with max(a_o);
68 |  end
69 end
70 Func is-fullConnector(u)
71 |  for l ∈ to L do
72 |  |  if ∄a_{u,l} then return;
73 |  |  false;
74 |  end
75 |  return true;
76 end
77 Func SpreadingActivationValue(v, e, u, l)
78 |  possibleSpreadAct = possible spreading activation from node u
   |  via edge e to node u for identifier l;
79 |  return possibleSpreadAct;
80 end
```

Die zu bestimmenden Seme sind somit die im Rahmen des Dokumentes referenzierten Bedeutungen der Bezeichner. Zunächst werden alle Knoten bestimmt, die Seme repräsentieren, *d.h.* Intensionen der Bezeichner. Sollte ein Knoten in zwei unterschiedlichen Intensionen enthalten sein, *d.h.* verschiedene Bezeichner darstellen, so wird dies bei der Initialisierung bereits berücksichtigt und schließt somit die Erzeugung von Duplikaten aus. Die anschließende Ordnung über die Aktivierungswerte ermöglicht eine priorisierte Exploration und somit eine Fokussierung auf Teilbäume, die Knoten hoher Aktivierung enthalten. Das ist auf die Vorgehensweise des Algorithmus zurückzuführen. Jeder Pfad kann nur bereits explorierte Knoten, *d.h.* Knoten mit einer höheren Gesamtaktivierung als der zu explorierende Knoten, beinhalten. Sollte dennoch eine Optimierung der Aktivität bzw. der Distanz der bereits explorierten Knoten möglich sein, so wird diese über die Funktionen COSTUPDATE(u, l) bzw. ACTIVATIONUPDATE(u, l) unverzüglich weitergeleitet. Bei jeder Änderung dieser Werte hinsichtlich eines Vorgängerknotens wird diese ebenfalls zu dessen bisher explorierten Vorgängern weiter propagiert. Dies ermöglicht ein zu jeder Zeit aktualisiertes Extensionsnetz. Potentielle Wurzelknoten von Steinerbäumen werden innerhalb der stattfindenden Exploration sowie während der Propagierung innerhalb der Update-Funktionen ermittelt. Dort ist es jeweils möglich einen Wurzelknoten u zu erzeugen, der eine Verbindung zu allen Bezeichnern aufweist. Der explorierte Knoten verfügt über alle Informationen seiner Vorgänger und damit alle von diesen ausgehenden Verbindungen zu Bezeichnern. Somit ist dieser Knoten der erste, der innerhalb des von ihm repräsentierten Lösungsbaums über die vollständige Anzahl der benötigten Verbindungen, *d.h.* für alle Bezeichner verfügt. Diese werden durch die Propagierung weitergegeben, so dass jeder Knoten $p \in P_{u,l}, \forall l \in L$ des Lösungsbaums $\mathcal{N}$ einen potentiellen Wurzelknoten darstellt. Durch dieses Vorgehen wird gewährleistet, dass dieser Baum zu jeder Zeit durch den Knoten mit der höchsten Gesamtaktivierung $max(a_n), \forall a \in \mathcal{N}$ als Wurzelknoten repräsentiert wird.

Durch die Vorstellung der algorithmischen Vorgehensweise wird deutlich, dass die asymmetrische Weitergabe der Aktivierungen im Fokus des Algorithmus liegt. Prägnantestes Merkmal hierfür ist die nach Aktivierungswert geordnete Queue Q. Diese ermöglicht bereits eine selektive Auswahl der Knoten, deren Aktivierung bevorzugt weitergegeben werden soll. Weiterhin ermöglichen Gewichtungen von

Knoten und Kanten den Einfluss auf den weiterzugebenden Aktivierungswert (siehe Kapitel 12). Diese Vorgehensweise orientiert sich dadurch am Prinzip der Closed Referential Polysemy (Definition 8.1). Das bedeutet außerdem, dass im Gegensatz zu der in Abschnitt 8.3 erwähnten „breadth-first“ Exploration innerhalb des vorgestellten Algorithmus eine Tiefensuche, durch „best-first“ Exploration basierend auf dem Auswahlkriterium des Aktivierungswerts, vorgenommen wird.

Wie zuvor angesprochen wird eine Änderung der Aktivierung im Knoten u bezüglich eines Bezeichners l umgehend an die Vorgängerknoten in $P_{u,l}$ weitergegeben. Es folgt eine Neubewertung der Relevanz jedes einzelnen Knoten. Diese Bewertung ist abhängig von den Bedeutungen der Knoten, die mit dem untersuchten Knoten innerhalb des Ontologiegraphen in Verbindung stehen und sie wird während des Algorithmus fortwährend aktualisiert. Dieses Vorgehen der Aktualisierung durch den Spreading Activation Algorithmus wird von Kacholia et al. als „PageRank with decay“ [118] bezeichnet, da die in Pulsen, *d.h.* iterativ, erfolgenden Spreading Schritte jeweils zu einer Neugewichtung der Knoten führen können.

8.4.4. Fokussierung auf Aktivitätswerte

Im bisher vorgestellten Algorithmus erfolgt die Bewertung von Knoten durch die Distanzwerte für jeden Bezeichner $l \in L$ (siehe Algorithmus 4 bzw. 6, Funktion ANALYSECONNECTION(v, e, u)). Mit der Berücksichtigung der Distanz geht eine Beschränkung der Anzahl der Explorationsschritte einher, da kürzere Pfade bei einer iterativen Exploration weniger zu untersuchende Knoten aus Q bedeuten. Ist beim Knoten u eine Minimierung der Distanz für den Bezeichner l durch den Knoten v möglich, so werden alle bisher gespeicherten Vorgängerknoten in $P_{u,l}$ durch diesen Knoten ersetzt und es erfolgt die Bekanntmachung dieser Veränderung über die Funktion COSTUPDATE(u, l). Bei der gleichen Distanz wird nur der Knoten mit der höchsten Aktivierung für diese Distanz gespeichert, alle anderen Knoten werden entfernt.

Algorithmus 6 : Distanzfokussierte Exploration

```
1  Func ANALYSECONNECTION(v, e, u)
2      foreach l ∈ L do
3          if the activation spreaded from v to u for l is greater than a_{u,l} then
4              update a_{u,l} with this new activation;
5              ACTIVATIONUPDATE(u, l);
6          end
7          if d_{v,l} + 1 ≤ d_{u,l} then
8              if d_{v,l} + 1 < d_{u,l} then
9                  P_{u,l} ← ∅ ;
10                 d_{u,l} = d_{v,l} + 1;
11                 add v to P_{u,l};
12                 if IS-FULLCONNECTOR(u) then ADDRESULT(u);
13                 COSTUPDATE(u, l);
14             end
15             else
16                 foreach p ∈ P_{u,l} do
17                     if a_v > a_p then remove p from P_{u,l} and
                       insert = tue;;
18                 end
19                 if insert then
20                     add v to P_{u,l};
21                 end
22             end
23         end
24     end
25     add v to P_u;
26 end
```

In Algorithmus 7 ist eine alternative Umsetzung der Funktion ANALYSECONNECTION(v, e, u) dargestellt. Diese alternative Funktion ermöglicht die Knotenbewertung ausschließlich durch die Beurteilung von Aktivierungswerten. Zunächst erfolgt die Bestimmung der möglichen Aktivierung *possibleKeyAct* im Zusammenhang mit dem aktuell untersuchten Bezeichner (Zeile 3 Algorithmus 7, siehe Kapitel 12 für die Umsetzung der Funktion). Die Bewertung der Vorgängerknoten innerhalb der Funktion wird nun anhand der Aktivierungswerte vorgenommen. Ein höherer Aktivierungswert bedeutet die vollständige Ersetzung der vorhandenen Knoten innerhalb von $P_{u,l}$ durch

den Knoten v und eine Bekanntmachung dieser Veränderung über die Funktion ACTIVATIONUPDATE(u, l). Falls die mögliche Aktivierung der vorhandenen Aktivierung entspricht, wird der Knoten hinzugefügt. Danach erfolgt ebenfalls die Überprüfung, ob es sich eventuell um einen Wurzelknoten handelt (Zeile 8).

Algorithmus 7 : Aktivitätsfokussierte Exploration

```
1  Func ANALYSECONNECTION(v, e, u)
2      foreach l ∈ L do
3          possibleKeyAct = SPREADINGACTIVATIONVALUE(v, e, u, l);
4          if possibleKeyAct ≥ a_{u,l} then
5              if possibleKeyAct > a_{u,l} then
6                  P_{u,l} ← ∅ ;
7                  update a_{u,l} with possibleKeyAct;
8                  if IS-FULLCONNECTOR(u) then ADDRESULT(u) ;
9                  ACTIVATIONUPDATE(u, l);
10             end
11             add v to P_{u,l};
12         end
13     end
14 end
```

8.4.5. Nichtzusammenhängende Teilbäume

Durch eine entsprechende Topologie des Instanzgraphen $\mathcal{G}$ besteht die Möglichkeit, dass kein Knoten $v \in V(\mathcal{G})$ aufgefunden werden kann, der über mindestens eine Verbindung zu einem Knoten je Intension verfügt. Somit existiert kein Wurzelknoten eines Steinerbaums, der einen Pfad zu mindestens einem Sem je Bezeichner, *d.h.* einer Adresse je Intension, ausgehend von diesem Knoten enthält. In Abbildung 8.6 ist dies dargestellt.

Der Grund für die fehlende Verbindung liegt in der Tatsache, dass die Instanzen der Ontologie zwar über ihre Konzeptzugehörigkeit eine Verbindung zueinander besitzen können, die sich über die schematischen Zusammenhängen ergibt, diese impliziert jedoch keine Verbindung im Instanzgraphen. Somit existieren nur Extensionen eines beschränkten Umfangs für die gegebenen Knoten der Intensionen $v \in I$.

Diese sind eingeschränkt durch die möglichen Pfade innerhalb des Graphen, *d.h.* lokal beschränkt.

Wird dieser Fall festgestellt, indem kein Wurzelknoten ermittelt werden kann, der für alle Bezeichner Aktivierungswerte aufweist[19], so wird im Rahmen dieser Arbeit ein selbstentwickeltes Verfahren eingebettet, das eine zuvor nicht im Graph vorhandene Verbindung zu den bisher in R vorhandenen Knoten erzeugt. Der Algorithmus wird normal fortgesetzt und es folgt eine Analyse der einzelnen Verbindungen. Die dadurch festgestellten Wurzelknoten werden nach Aktivität geordnet und der höchstaktivierte bezeichnet die Lösung des Verfahrens.

Solche Situationen entstehen, *z.B.* in der Entitätsdisambiguierung, falls in untersuchten Dokumenten verschiedene Themengebiete besprochen werden, die voneinander unabhängig sind. Diese Unabhängigkeit äußert sich oftmals in fehlenden Beziehungen innerhalb des verwendeten Hintergrundwissens.

Beispiel getrennte Teilgraphen In Abbildung 8.6 sind ausserhalb des oberen mittleren Graphen die vom Algorithmus berechneten Resultatgraphen der höchsten Aktivierung für die jeweiligen Bezeichnerkombinationen aufgeführt. Diese sind mit ihren jeweiligen Wurzeln (durch R gekennzeichnet) in der Resultatmenge R enthalten. Hierbei handelt es sich ausschließlich um Teilresultate, *d.h.* es ist kein Resultatbaum enthalten, der für jeden Bezeichner eine Adresse enthält. Daher wird nun ein künstlicher Wurzelknoten erzeugt. Dieser wird mit den zuvor berechneten Wurzelknoten[20] der Teilresultate verbunden (siehe mittlerer oberer Graph in der Graphik). Es erfolgt eine Exploration und zugehörige Spreading Schritte, die dazu führen, dass der Graph nach Wurzelknoten für eine Lösung durchsucht wird, die nun alle Bezeichner berücksichtigt. Dies ist im unteren Teil der Abbildung dargestellt.

[19] Zudem kann eine Abbruchbedingung ausschlaggebend sein. Eine solche basiert *z.B.* auf einer vorgegebenen maximalen Anzahl an Explorationsschritten.

[20] Im Fall, dass mehrere festgestellte Wurzelknoten die höchste Aktivierung teilen (*z.B.* im linken oberen und rechten Teilgraph), wird eine Verbindung zu jedem dieser Wurzelknoten hergestellt.

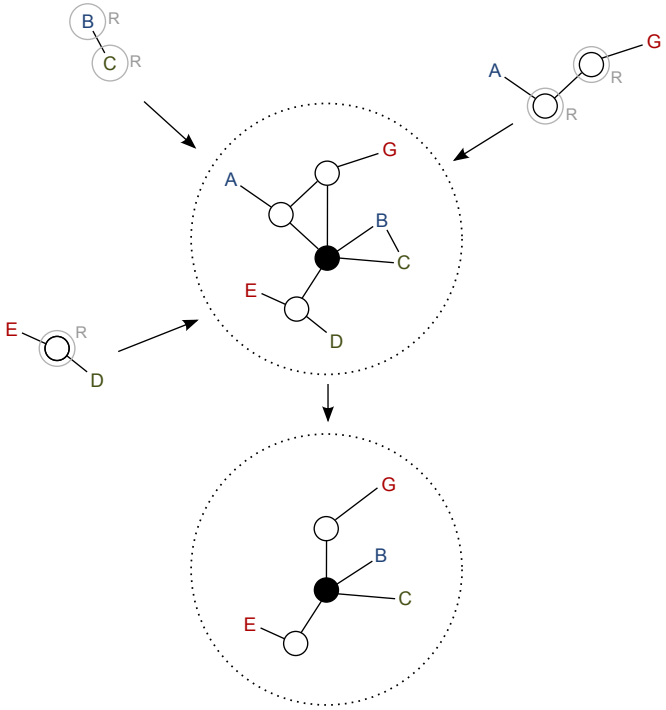


Abbildung 8.6.: Beispiel getrennte Teilgraphen

8.4.6. Anwendungsbeispiel Basisalgorithmus

Im Folgenden wird eine konkrete Anwendung des Algorithmus im Rahmen eines gegebenen Dokuments und einer zugehörigen Domänenontologie vorgestellt. Es handelt sich hierbei um einen Auszug der in Kapitel 13 vorgestellten Evaluationsdaten des Algorithmus. Die gegebene Domäne beschreibt Hintergrundwissen über die Zusammenhänge zwischen geographischen Daten. Ziel, der nun vorgestellten Disambiguierung, ist die Erkennung der im Kontext des Dokumentes verwendeten Entitäten[21], die der Intention des Autors

[21] Es ist darauf hinzuweisen, dass anstatt von Entitäten auch andere Begrifflichkeiten zur Disambiguierung verwendet werden können (*z.B.* Verben, Pronomina). Hierfür ist allerdings eine Anpassung der Knotengenerierung und eine korrekte Auswahl der Hintergrundontologie von Bedeutung.

des untersuchten Dokumentes entsprechen.[22] Der Beispieltext lautet:

Beispiel: *"A wildfire in northern Arizona [...] a fire north of Lake City in Florida. Flames remained about a mile from the community of Christopher Creek. The community is south of See Canyon [...]. Elsewhere New Jersey [...]"*

Im Folgenden wird der in Abschnitt 7.2 vorgestellte Analyseprozess durchlaufen. Zunächst wird der Text analysiert und die darin enthaltenen Entitäten identifiziert. Diese Entitäten *Arizona, Lake City, Florida, Christopher Creek, See Canyon* und *New Jersey* sowie deren Position sind im Text durch Unterstreichen hervorgehoben. Für jeden der sechs Bezeichner wird unter der direkten Verwendung der Wörterbuchfunktion die zugehörige Intension ermittelt. Die initiale Aktivierung ist abhängig vom Grad der Ambiguität, *d.h.* $|f_{Intension}(l)|$ (siehe Kapitel 13 für eine vollständige Darstellung der vorgenommen Gewichtungen). Die Adressen in der Intension des Bezeichners *Florida* werden am geringsten gewichtet, da $|Int_{Florida}| = 249$ und die Adressen des Bezeichners *New Jersey* am höchsten, $|Int_{NewJersey}| = 3$. Dies spiegelt sich in der Reihenfolge der Exploration wieder, da sich die Knoten, der Intensionen des Bezeichners *New Jersey* an der Spitze der Queue Q befinden. Das bedeutet, dass die Exploration von diesen Knoten aus startet und sich zunächst auf deren direkte Umgebung konzentriert. Nach dieser Anfangsphase fokussiert sich die Exploration auf Knoten, in denen die Aktivierungen bezüglich mehrerer Bezeichner zusammentreffen. Das ist in Abbildung 8.7 dargestellt. Zunächst kommen die Aktivierungen mehrerer Bezeichner durch die Explorationen der ersten Knoten beim Knoten zusammen, der das Zentrum des linken Graphen darstellt. Durch dessen weitere Exploration werden Beziehungen zu den noch fehlenden Bezeichnern aufgebaut. Der diskutierte Knoten wird aufgrund der Vollständigkeit der Bezeichnerinformationen als Wurzelknoten identifiziert. Im Kontext der Domäne handelt es sich bei diesem Knoten um den Repräsentant der „USA", die als Staat alle gesuchten Städte und Länder enthält und somit als „übergeordnete Instanz" fungiert.

Der in Abbildung 8.7 links dargestellte Graph zeigt einen Ausschnitt aus dem Instanzgraphen $\mathcal{G}$, welcher die rund um den Wurzelknoten

[22] Es wird vorausgesetzt, dass der Autor des Textes Kenntnisse über die Domäne besitzt und das Dokument in sich keinen Widerspruch darstellt.

der USA lokalisierte Information darstellt. Hierbei wird ebenfalls die Ambiguität durch die Vielzahl an Knoten der gleichen Farbe deutlich.[23] Im rechten Teil ist der vom Algorithmus zurückgelieferte Steinerbaum dargestellt.[24] Die Minimierung der möglichen Seme wurde durch die Aktivierungswerte und Gewichtungen ermöglicht. Im gegebenen Beispiel war es jedoch nicht möglich, die exakte Referenz für *Christopher Creek* zu bestimmen. Jedoch beträgt die Reduktion der Ambiguität für diesen Bezeichner bereits 90%.

[23] Hierbei stehen alle Farben außer rot für unterschiedliche Intensionen.

[24] Basierend auf dem höchstaktivierten Wurzelknoten und somit auf dessen zugeordneten Pfaden zu den Repräsentanten der Bezeichner.

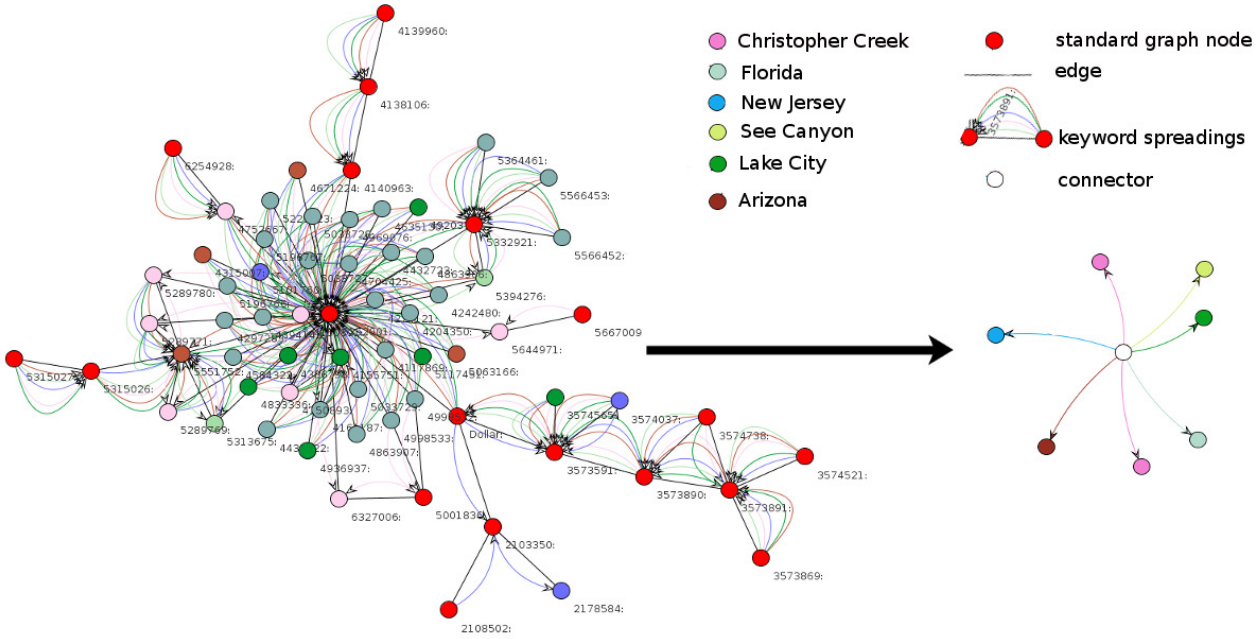


Abbildung 8.7.: Beispielausführung des Basisalgorithmus

9. Bidirektionaler Ansatz

Die im Folgenden vorgestellte Modifikation des in Kapitel 8 eingeführten Basisansatzes befasst sich mit dem Informationsaustausch während der Analyse einer Verbindung zwischen zwei Knoten. Optimierungsmöglichkeiten liegen bei der Menge und Qualität der Informationen über Bezeichnerverbindungen, die ausgetauscht werden können. Diese sind ebenfalls für die Bewertung eines Knotens ausschlaggebend. Die initiale Aktivierung zusammen mit den bezeichnerspezifischen Aktivierungen bestimmt die Gesamtaktivierung des Knotens und somit dessen Explorationszeitpunkt, *d.h.* dessen Selektion von Q. Eine Änderung beeinflusst die weitere Explorationsreihenfolge und somit auch die Wahl des Resultatgraphen.

In Abschnitt 9.1 wird die Vorgehensweise der bidirektionalen Exploration und deren Unterschiede zur unidirektionalen Exploration erläutert und anhand eines Beispiels aufgezeigt. Die Auswirkungen auf die Exploration des Graphen und das Auffinden der Steinerbäume wird in Abschnitt 9.2 behandelt.

9.1. Unterscheidung der uni- und bidirektionalen Exploration

Die Änderung betrifft die Analysephase des Basisansatzes. Innerhalb der Funktion ANALYSECONNECTION(v, e, u) findet eine Analyse anhand des Tripels $< v, e, u >$ statt, *d.h.* anhand der Knoten v und u, die über die Kante $e_{v,u}$ verbunden sind. Innerhalb des Basisansatzes wird kontrolliert, ob der Zielknoten u der Verbindung hinsichtlich der vom Knoten v bereitgestellten Informationen bereits über eigene verfügt und ob diese eine höhere Qualität[1] aufweisen. Nur im Fall, dass

[1] Eine höhere Qualität definiert sich im Basisansatz durch einen höheren Aktivierungswert bzw. einen geringeren Distanzwert.

die vom Knoten v bereitgestellten Werte der Bezeichneraktivierung und/oder der Distanz eine Verbesserung oder Gleichheit gegenüber den aktuell zugeordneten Werten ermöglicht, erfolgt eine Aktualisierung der dem Knoten u zugeordneten Werte. Gegebenenfalls schließt sich bei einer Verbesserung eine Propagierung der Werte an die zuvor explorierten Knoten an. Falls der Knoten u nicht bereits zuvor erkundet wurde, wird er anhand des aktualisierten Aktivierungswerts in die Queue Q eingefügt.

Abbildungen 9.1 und 9.2 veranschaulichen diesen Vorgang des unidirektionalen Abgleichs der Werte an einem Beispiel. Knoten v repräsentiert eine Intension für den Begriff „USA“ ($v \in Int_{USA}$) und besitzt einen diesbezüglichen Aktivierungswert. Knoten p besitzt bereits eine Aktivierung für den Begriff „Kansas“ und Knoten u für den Begriff „Boston“ (*z.B.* durch vorhergehende Spreading-Schritte). Zunächst erfolgt eine Selektion des Knotens v (Entitätsbezeichner „USA“) von Q. Innerhalb des Instanzgraphen besitzt der Knoten v eine Verbindung zum Knoten u ($< v, e_1, u >$) und zum Knoten p ($< v, e_2, p >$). Im darauffolgenden Analyseschritt wird zunächst die zuerst genannte Relation überprüft. Da der Knoten u über keinen zugeordneten Wert für den Bezeichner „USA“ verfügt, überträgt der Knoten v die diesbezüglichen Distanz- und Aktivierungswerte (siehe Schritt 1). Die Untersuchung der Relation führt zu keinen weiteren Aktualisierungen. Für die zweite Relation $< v, e_2, p >$ wird dieselbe Verfahrensweise angewandt. Hier wird der Wert für den Bezeichner „USA“ vom Knoten v zum Knoten p übertragen (siehe Schritt 2).

Abbildung 9.1.: Zustand vor Ausführung des unidirektionalen Abgleichs

In diesem Beispiel ist es offensichtlich, dass die Informationen zu den Bezeichnern „Boston“ und „Kansas“ sich in greifbarer Nähe befinden. Aufgrund der unidirektionalen Exploration können diese Informationen jedoch erst später ausgetauscht werden. Die Verzögerung ergibt sich dadurch, dass die Knoten u bzw. p zuvor exploriert werden müssen und diese erst die jeweilige Bezeichnerinformation an den Knoten

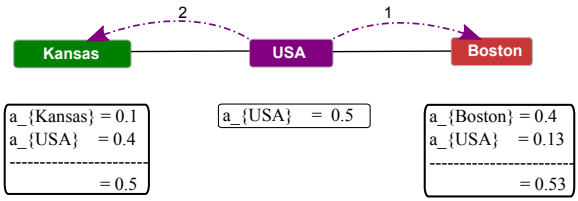


Abbildung 9.2.: Beispielausführung des unidirektionalen Abgleichs

v weitergeben. Eine weitere Möglichkeit stellt die Exploration eines alternativen Knotens der Menge Int_{Boston} bzw. Int_{Kansas} dar. Letztere impliziert die Weitergabe über Propagierung.[2]

Der vorgestellte Ablauf des Informationsaustausches war Motivation für die im Folgenden beschriebene Variante (dargestellt in Abbildung 9.3). Anstatt des in unterschiedlichen Schritten erfolgenden Austauschs liegt dieser die Idee zugrunde, dass alle innerhalb eines Tripels vorhandenen Informationen innerhalb deren Analysephase miteinander ausgetauscht werden. Dies bedeutet ein Zusammentreffen der durch dieses Tripel repräsentierten Information und deren Auswertung für den Ursprungs- sowie für den Zielknoten. Auf das Beispiel und somit zunächst auf das Tripel $< v, e_1, u >$ bezogen bedeutet dies, dass der oben dargestellten Variante entsprechend überprüft wird, ob der Knoten u einen Informationsgewinn durch den Knoten v erzielen kann. Dies ist der Fall, da der Knoten u über keine Information hinsichtlich des Bezeichners „USA“ verfügt (vgl. Schritt 1). Jedoch erfolgt nun im direkten Anschluss eine zusätzliche, inverse Analyse der Verbindung, *d.h.* ANALYSECONNECTION(u, e_1, v). Diese findet unabhängig von der Einordnung des Knotens u in der Queue Q statt. Somit erfolgt auf das Beispiel bezogen die Weitergabe der „Boston“-spezifischen Werte vom Knoten u zum Knoten v (vgl. Schritt 2). Hier findet ebenfalls eine Propagierung der geänderten Werte an die Vorgängerknoten statt. Im Beispiel besitzt der Knoten v zu

[2] Bei einer Restriktion der Erreichbarkeit des Knoten v über die zwei vorgestellten Relationen impliziert diese, dass der Knoten p und/oder u bereits exploriert wurde(n). Beispielsweise über die Verbindung v, e_1, u, e_3, o. Der Knoten $o \in Int_{Kansas}$ besitzt eine höhere Aktivierung als der Knoten p und wird dadurch vor diesem erkundet. Durch das Hinzufügen des Werts für den Bezeichner „Kansas“ zum Knoten u erfolgt ein Propagieren an die Vorgänger und somit auch ein Hinzufügen des Werts zum Knoten v.

diesem Zeitpunkt nur den Knoten u als Vorgängerknoten und in Folge dessen findet hier keine Aktualisierung von Werten statt.[3] In der weiteren Analyse des mit dem Knoten v verbundenen Pfades $< v, e_2, p >$ erfolgt die Weitergabe der Werte für „USA“ vom Knoten v zum Knoten p. Da der Knoten v nun jedoch auch über Informationen zum Bezeichner „Boston“ verfügt, werden diese ebenfalls weitergegeben (vgl. Schritt 3 für die Weitergabe der Werte „USA“ und „Boston“). Anschließend erfolgt auch hier eine Analyse der inversen Relation ANALYSECONNECTION(p, e_2, v), die einen Austausch der Informationen bezüglich des Bezeichners „Kansas“ bewirkt. Zunächst wird der Knoten v aktualisiert (vgl. Schritt 4). Über die Propagierung wird die Information zudem zum Knoten u übertragen (vgl. Schritt 5). Somit verfügen alle drei Knoten v,u und p nach der Analyse des Knoten v über die vollständige Information bezüglich aller drei Bezeichner „Kansas“, „USA“ und „Boston“.

Hervorzuheben ist, dass der auf das Beispiel bezogene Austausch der Informationen bei der unidirektionalen Variante eine Analyse aus Sicht von mindestens[4] drei verschiedenen Knoten, *d.h.* deren jeweilige Selektion aus der Queue Q, erfordert. In der hier vorgestellten Verfahrensweise erfolgt die Analyse aus Sicht eines Knotens, der unabhängig von der Aktivierungspriorisierung die inversen Analysen implizit startet.

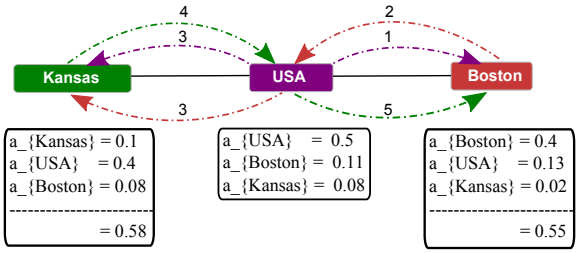


Abbildung 9.3.: Beispielausführung des bidirektionalen Abgleichs

[3] Da der Knoten u den Ursprung der Werte darstellt, die von v propagiert werden, verfügt dieser über vorteilhaftere Werte hinsichtlich des aktuell überprüften Bezeichners.

[4] Je nach Reihenfolge der Knoten in Q können zusätzliche Knoten zwischenzeitlich exploriert werden.

Das als bidirektionale Analyse bezeichnete Verfahren beschränkt sich bei der Analyse der mit dem Knoten v in Beziehung stehenden Knoten jeweils auf die gegebene Relation. Dies ist zu unterscheiden von einer vollständigen Exploration aller mit diesen Knoten in Beziehung stehenden Knoten. Angenommen, die im Rahmen des unidirektionalen Beispiels erwähnte Erweiterung würde existieren, *d.h.* der Knoten p wäre zusätzlich zu der Verbindung mit v auch dem Knoten o verbunden, so würde diese Verbindung frühestens bei der Analyse von o bzw. p nach deren Selektion von Q vorgenommen.

9.2. Auswirkungen der bidirektionalen Exploration

Betrachtet man die Auswirkungen der bidirektionalen Exploration, so können drei wesentliche Punkte identifiziert werden:

1. Jeder Knoten[5] innerhalb der Analysephase besitzt das Maximum an Information, das die direkt mit ihm in Beziehung stehenden Knoten zur Verfügung stellen können.
2. Die Reihenfolge des Informationsaustausches sowie der zu explorierenden Knoten verändert sich.
3. Die Expansion des Graphen tendiert zu einer primär lokal beschränkten Ausdehnung. Die Ausweitung, *d.h.* die Untersuchung längerer Pfade, erfolgt zeitlich mit teils großer Verzögerung.

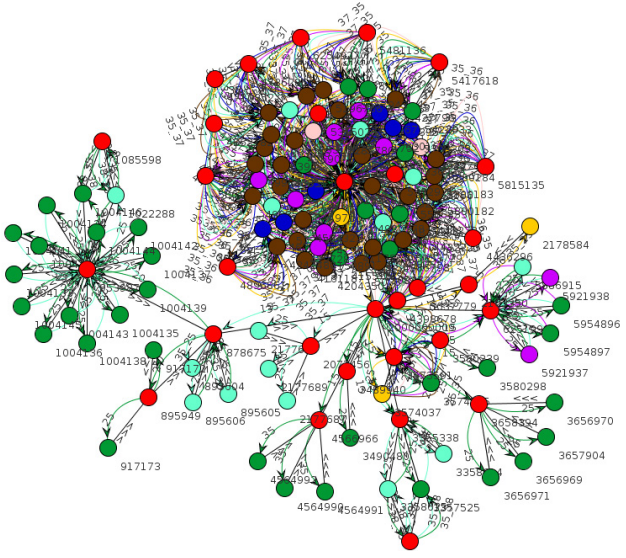
Punkt 1 beschreibt den Umstand, dass alle mit v in Beziehung stehenden Knoten ihre Informationen ebenfalls innerhalb der Analysephase des Knotens v mit v austauschen. Dieser gibt die Informationen bei weiteren Relationsanalysen oder durch propagieren neuer oder verbesserter Werte weiter. Somit besitzen am Ende der Analyse des Knotens v alle von ihm erreichbaren Knoten Informationen über die

[5] Bezogen auf eine Analysephase handelt es sich um den Knoten v und jeden Knoten GETPATHSTEPSOFINTEREST(v). Zudem ist die Analyse auf die Verbindungen vom Knoten v zu jedem einzelnen Knoten u beschränkt. Eine Analyse weiterer Relationen ist erst bei der Wahl eines neuen Knotens von Q möglich.

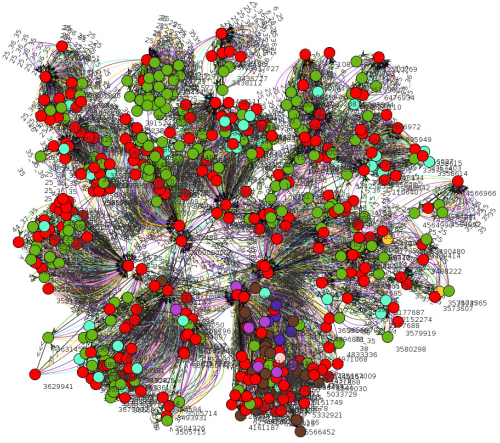
gleiche Anzahl von Bezeichnern.[6] Punkt 2 resultiert aus dem vollständigen Informationsaustausch, der in Punkt 1 erwähnt wird. Dieser geht einher mit der Steigerung der Gesamtaktivierung des Knotens v. Dadurch wird seine Relevanz als Vorgängerknoten erhöht, jedoch – da die Exploration des Knoten nun abgeschlossen wurde – hat dieser Umstand keine Auswirkung auf die Queue Q. Dennoch besitzt nach der Analyse auch jeder Knoten $u \in \text{GETPATHSTEPSOFINTEREST}(v)$ die vollständigen Informationen, mit denen eine mögliche Steigerung seiner Gesamtaktivierung einhergeht und somit gegebenenfalls eine Änderung seines Ranges in Q. Die in Punkt 3 beschriebene Auswirkung des Verfahrens ist ein direktes Resultat der in Punkt 2 beschriebenen Reihenfolgenänderung in Q. Dies wird durch die Darstellung einer Beispiel-Exploration in Abbildung 9.4 deutlich. Die Abbildung zeigt Ausschnitte der unterschiedlichen Explorationen ausgehend von der algorithmischen Analyse des in Abschnitt 8.4.6 gezeigten Textbeispiels. Bei Anwendung des unidirektionalen Abgleichs (siehe Abbildung 9.4(a)) wird der Fokus auf die rasche Exploration von Pfaden mit der Weitergabe des im Lauf des Pfades gesammelten Wissens ersichtlich. Ein intensive Analyse ist im oberen Teil der Abbildung zu sehen, bei dem viele Seme nahe beieinander liegen. Nach außen nimmt die Weitergabe der Informationen aufgrund der Beschränkungen, *z.B.* Minimalaktivierung, ab. Im Fall der bidirektionalen Exploration (siehe Abbildung 9.4(b)) wird die intensive Überprüfung durch die lokal gebündelten Bereiche deutlich. Wie zuvor erläutert ist dies zum einen ein Resultat des vollständigen Informationsaustauschs und der priorisierten lokal zusammenhängenden Exploration. Zum anderen verzögern die gesteigerten Gesamtaktivierungswerte neben der besseren Einordnung in Q das Erreichen des Zeitpunkts, an dem die zusätzlichen Abbruchbedingungen hinsichtlich der Minimalaktivierung *etc.* erfüllt werden.

Der Unterschied der Varianten wird im Rahmen der Evaluation in Kapitel 13 aufgezeigt.

[6] Diese Informationen sind hinsichtlich des Aktivierungs- und Distanzwertes aufgrund der mathematischen Berechnung abweichend von den dem Knoten v zugewiesenen Werte (siehe Kapitel 12).



(a) Unidirektionaler Ansatz



(b) Bidirektionaler Ansatz

Abbildung 9.4.: Unterschied zwischen uni- und bidirektionaler Exploration

10. Ansatz der lokalen Kohärenz

Der Ansatz zur Disambiguierung, der in dieser Arbeit vorgestellt wird, besitzt weitgehende Allgemeingültigkeit für verschiedenste Arten von Eingangsmedien. Voraussetzung ist hierbei die mögliche Extraktion von darin vorkommenden Elementen, denen Instanzen der Ontologie zugewiesen werden können und somit die Übertragung des durch das Medium dargestellten Sachverhalts auf das in der Ontologie gespeicherte Hintergrundwissen.[1] Für die innerhalb dieses Kapitels vorgestellte Erweiterung des in Kapitel 8 eingeführten Basisalgorithmus sind spezifische Eigenschaften von Texten ausschlaggebend. Daher wird in Abschnitt 10.1 zunächst auf die Merkmale von Texten eingegangen und die Begrifflichkeit des Wortes „Kohärenz“ erläutert. Der Zusammenhang zwischen Textkohärenz und dem in Kapitel 8 vorgestellten Verfahren sowie die daraus hervorgehenden Änderungen werden in Abschnitt 10.2 vorgestellt. In Abschnitt 10.3 wird die Verwendung von lokalen Propositionen[2] auf eine algorithmische Vorgehensweise übertragen und in Abschnitt 10.4 anhand eines Beispiels dargestellt.

---

1 Beispielsweise ist die kontextuelle Referenzbestimmung innerhalb Tonaufnahmen, Bilddarstellungen, Videos *etc.* möglich. Je nach Medium bedarf dies einer unterschiedlichen Vorverarbeitung, *z.B.* die Überführung von Tonaufnahmen in Worte, Sätze *etc.*, bei Bildern die Erkennung von Gegenständen und bei Videos eine Kombination von beidem. Die daraus resultierenden Informationsobjekte müssen anschließend dem Hintergrundwissen der Ontologie zugeordnet werden. Darauf aufbauend kann eine Verarbeitung durch die vorgestellte Methode erfolgen.

2 Dieser in der Linguistik häufig verwendete Begriff steht für den ausgedrückten Sachverhalt.

10.1. Textkohärenz

Im Gegensatz zu menschlichen Gesprächen, die in den meisten Fällen einer spontanen und somit unstrukturierten Vorgehensweise folgen, ist es verbreitet und nahezu ein Standard bei schriftlichen Dokumenten sich an eine Form der Strukturierung zu halten.[3] Der Urheber eines Textes, *d.h.* der Autor, steht vor dem Problem, den von ihm anvisierten, zu vermittelnden Inhalt in eine adäquate Struktur zu bringen, um einen inhaltlichen Zusammenhang zu gewährleisten. Während ein Autor jedoch bereits den Inhalt kennt und ihm vorhandenes Wissen über den Inhalt als vorausgesetzt betrachtet werden kann, muss der Leser diesen Schritt des „Verstehens" zuerst vollziehen. Dies bedeutet, dass er das dort Beschriebene miteinander in Zusammenhang bringen muss, *d.h.* die Kohärenz erkennen muss. Hierfür müssen die einzelnen Textaussagen erkannt und deren jeweilige Proposition vom Leser „gespeichert" werden (siehe auch Kintsch und Dijk [123]). Im Folgenden gilt es für den Leser diese einzelnen Propositionen miteinander zu verknüpfen und in eine Wissensstruktur zu überführen, um ein „Verstehen" zu ermöglichen. Hierbei hilft die Tatsache, dass Begriffe oftmals wiederholt werden und somit anhand eines Begriffes Zusammenhänge zugeordnet werden können. Quathamer [183] spricht von einem „Sinnfluss", der sich durch diese Zuordnung ergibt, *d.h.* die Kohärenz des Textes.

Schnotz beschrieb die Kohärenz eines Textes durch: *„Die Kohärenz eines Textes ist das, was ihn von einer bloßen Aneinanderreihung beliebiger Sätze unterscheidet. Im allgemeinsten Sinne bedeutet dieser Begriff, dass die einzelnen Teile eines Textes einen Gesamtzusammenhang bilden, d.h. die ihnen entsprechenden Propositionen bzw. Bedeutungseinheiten durch semantische Relationen zu einem integrierten Ganzen verbunden sind. Ein Text ist demnach kohärent, wenn sein Inhalt durch ein zusammenhängendes Netzwerk darstellbar ist und nicht in einzelne, miteinander unverbundene Teilnetze zerfällt."* [205]

Diese Beschreibung hat zwei Kernaussagen: *(a)* Es gibt Teile eines Textes, die für sich einen Zusammenhang darstellen und *(b)* diese

[3] Ausnahmen bei Gesprächen bilden Vorträge, Interviews *etc.*, die ebenfalls eine zuvor festgelegte Struktur verfolgen.

Teile fügen sich nahezu nahtlos in einen Gesamtzusammenhang [4] ein. Diese Teile verbinden das Wissen, das in ihnen beschrieben wird.

Kintsch et al. [123] beschreiben diese Unterteilung als lokale und globale Kohärenz. Lokale Kohärenz, als Mikrostruktur von den Autoren bezeichnet, baut auf der Analyse einzelner Sätze auf. Zunächst werden die Argumente aus ihnen extrahiert. Eine lokale Kohärenz geht nun aus der Argumentüberlappung benachbarter Sätze hervor. Ein Text lässt sich daher in viele lokale Kohärenzen unterteilen.[5]

Makrostrukturen existieren auf verschiedenen Ebenen. Die unterste beginnt mit der Zusammenfassung zweier lokaler Kohärenzen, die aufgrund Argumentüberlappung zusammengefasst werden. Weitere Ebenen entstehen aus der Zusammenfassung von Argumentüberlappung der nun erzeugten Makrostrukturen. Dies wird fortgesetzt, bis der Gesamtzusammenhang des Textes als letzte Makrostrukturebene dargestellt werden kann (siehe Abbildung 10.1).

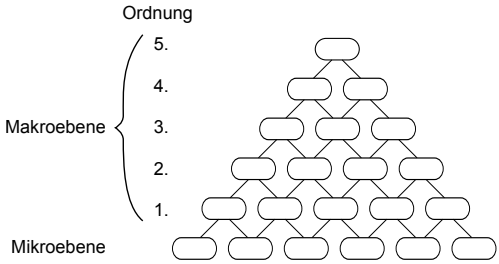


Abbildung 10.1.: Hierarchie der Mikro- und Makrostrukturen

Kintsch et al. [123] erstellten ebenfalls ein Computermodell, das den Vorgang der lokalen und globalen Kohärenzbildung simuliert (siehe auch Darstellung von Anderson in [9]). Die Autoren unterscheiden das Kurz- bzw. Arbeits- und Langzeitgedächtnis. Für unseren Ansatz ist die Tatsache wichtig, dass zuvor unbekannte Informationen mit dem Langzeitgedächtnis, *d.h.* der globalen Wissensbasis abgeglichen werden. Das Arbeitsgedächtnis repräsentiert einen Ausschnitt dieses

[4] Schnotz verwendet den Begiff „Netzwerk“ hierbei metaphorisch, um den Zusammenhang zwischen den Teilen auszudrücken.

[5] *Lokale Kohärenz* bezeichnet somit die Verknüpfung aufeinanderfolgender Zusammenhänge.

Wissens, der dem aktuellen Verständnis entspricht, das der Leser für den Text entwickelt hat.

10.2. Zusammenhang der Textkohärenz mit dem Basisansatz

Die im vorherigen Abschnitt vorgestellte Vorgehensweise des „Verstehens“ von Text beinhaltet viele Parallelen zu der in dieser Arbeit vorgestellten kontextuellen Referenzbestimmung (siehe Abschnitt 8.4) bzw. des Zwei-Ebenen-Modells (siehe Abschnitt 6.2.1). Innerhalb des erwähnten Computermodells findet ein Abgleich zuvor unbekannter Informationen mit dem Langzeitgedächtnis des Lesers statt. Der im Rahmen des in der Arbeit verfolgte Ansatz orientiert sich hieran durch die Verwendung einer Ontologie zur Repräsentation des diesbezüglichen Wissens. Das Arbeitsgedächtnis hingegen hält zu jeder Zeit die wichtigsten Informationen vor, auf die der Leser zur Durchführung der Verknüpfung von Informationen zurückgreift, *d.h.* die im Arbeitsgedächtnis aktiv vorgehaltenen Propositionen sind exakt die, die entlang der gerade vorgenommenen Exploration innerhalb des Graphen angeordnet sind (Kombination von zeitlicher Nähe und Priorität). Der Basisansatz ermöglicht durch die Verwendung von Spreading Activation ebenfalls einen priorisierten Rückgriff auf Informationen. Die wichtigste Information entspricht hierbei der höchstaktivierten. Während Anderson [9] bei der Interpretation des Ansatzes von Kintsch et al. darauf hinweist, dass die Kapazität des Arbeitsgedächtnises beschränkt ist, bestehen beim hier vorgestellten Ansatz keine Kapazitätsbeschränkungen. Die exakte Darstellung einer Proposition wird von Anderson nicht näher beschrieben. Zwar stellt er ein Beispiel hierfür vor, jedoch ist die Form der Darstellung nicht exakt festgelegt. Im Rahmen des vom Autor dieser Arbeit vorgestellten Ansatzes werden Propositionen durch Teilgraphen des Ontologieinstanzgraphen realisiert. Diese repräsentieren mögliche Aussagen, die gleichzeitig durch den Algorithmus gewichtet werden. Das bedeutet, dass eine globale Kohärenz durch eine kombinatorische Auswertung der einzelnen Propositionen bestimmt werden muss und zwar anhand von Informationen, die durch die Teilgraphen gegeben sind, welche die Propositionen repräsentieren.

Die Realisierung durch den Basisansatz weicht in einem wesentlichen Punkt jedoch von der im vorherigen Abschnitt vorgestellten Vorgehensweise ab, da bisher keine gezielte Berücksichtigung lokaler Kohärenz vorgenommen wurde. Alle gegebenen Informationen werden im Basisansatz für die Erstellung eines Gesamtzusammenhangs (globale Kohärenz) verwendet. Lokale Zusammenhänge ergeben sich hierbei im Laufe des Verfahrens durch die lokale Bündelung während der Exploration. Diese lokalen Zusammenhänge müssen jedoch nicht in allen Fällen den lokalen Zusammenhängen im Text entsprechen. Eine lokale Bündelung entspricht einer Exploration, die im engeren Umfeld, *d.h.* bei kleiner Distanz und hohen Aktivierungswerten, eines Knotens vorgenommen wird. Dies ist dann der Fall, falls viele Knoten dieses Umfeldes Seme (Adressen in den Intensionen der Bezeichner) repräsentieren und somit die initial höchsten Aktivierungswerte besitzen. Die Bezeichner dieser Seme müssen im Text nicht zwangsweise nahe beieinander stehen und deshalb kann dies ein Indikator für einen möglichen Fehler durch die Priorisierung suboptimaler Relationen bei der Exploration darstellen. Folglich können Seme im Graphen eng verbunden sein, deren Bezeichner im Dokument weit voneinander entfernt sind. Die Nähe von Semen verschiedener Bezeichner im Graphen hängt nicht zwangsläufig mit der textuellen Nähe (Wortabstand) der Bezeichner im Dokument zusammen. Um einen höheren Grad des in der textuellen Struktur vorhandenen Hintergrundwissen einzubringen, verfolgt der in diesem Kapitel vorgestellte Ansatz die Verbesserung des Verfahrens durch die gezielte Erstellung lokaler Kohärenzen. Hierbei sind folgende Kriterien zu beachten:

1. Lokale Kohärenz baut auf der Proposition einzelner Sätze auf (vgl. Kintsch et al. [123]).
2. Ein Text wird in Abschnitte formatiert, die sich auf Sinneinheiten beziehen (vgl. Kintsch et al. [123]).
3. Wiederholung eines Begriffs bildet einen Sinnfluss (vgl. Quathamer [183]).

Diese Kriterien fügen sich in die in Abschnitt 6.1 vorgestellte Kontextklassifizierung von Schippan ein, *d.h.* in den lexisch-semantischen Kontext und Situationskontext.

10.3. Umsetzung lokaler Kohärenz im Ansatz

Die hier vorgestellte Algorithmusvariante ermöglicht die Berücksichtigung lokaler Kohärenzen. Diese beginnt mit der Identifikation relevanter Textbereiche. Daher werden zunächst (in Übereinstimmung mit der Vorgehensweise des Basisansatzes) die Entitätsbezeichner sowie deren Position innerhalb des Textdokuments identifiziert. Die positionale Einbettung ermöglicht es, textuelle Zusammenhänge zu extrahieren. Diese werden benötigt, um die Lokalität der späteren Kohärenz zu gewährleisten. Daher erfolgt zunächst die Bestimmung des textuellen Kontexts, der den lokalen Zusammenhang darstellt. Hierzu muss der Text in einzelne Fragmente zerlegt werden, die einzeln analysiert werden. In der von Kintsch et al. vorgestellten Vorgehensweise wird auf Satzebene begonnen und anschließend werden die Propositionen iterativ in den Makroebenen ineinander überführt. Der hier vorgestellte Ansatz verwendet anstatt der Satzebene Bereiche aufeinanderfolgender Worte. So wird zunächst ein Wortbereich, der einen Bezeichner l umgibt, in ein Fenster vorgegebener Größe eingebettet, *z.B.* 5 Wörter[6], *d.h.* $w_{k-2}, w_{k-1}, l_k, w_{k+1}, w_{k+2}$ (k bezeichnet die Wortposition). Diese Festlegung entspricht dem lexisch-semantischen Kontext und beinhaltet außer der Bestimmung der innerhalb dieses Fensters vorkommenden Instanzbezeichner (und zugehörige Seme) kein weiteres Hintergrundwissen. Ein Textfenster entspricht daher innerhalb dieses Ansatzes der in Kriterium 1 vorgenommenen Textbereichsfestlegung des Satzes. Die Verwendung von Textfenstern erfolgt nach dem von Firth vorgegebenen Prinzip: „a word is characterized by the company it keeps“ [81]. Auf dieser untersten Ebene ist dies durch direkt aufeinanderfolgende Wörter gegeben. Im nächsten Schritt werden die einzelnen Wortfenster auf Überlappungen überprüft. Dies geschieht anhand der Wortposition innerhalb des Textes. Überlappende Wortfenster werden zu einem Segment miteinander vereinigt. Diese Vorgehensweise ist in Abbildung 10.2 dargestellt. Die Fenster b_1, b_2 und b_3 werden zu einem Segment B_1 zusammengefasst. Die Fenster b_1 und b_4 weiter unten im Text werden in B_2 zusammengefasst. b_5 bildet ein separates Segment B_3. Am Ende dieses Verabeitungsschrittes

[6] Die Größe des Textfensters muss individuell auf den zu untersuchenden Textkorpus angepasst werden. Hierfür sind mehrere Tests bzw. Expertenwissen notwendig.

beschreibt jedes Segment einen lokalen Kontext. Die darin enthaltenen Entitätsbezeichner werden in einer diesen Kontext beschreibenden Menge K_i zusammengefasst.

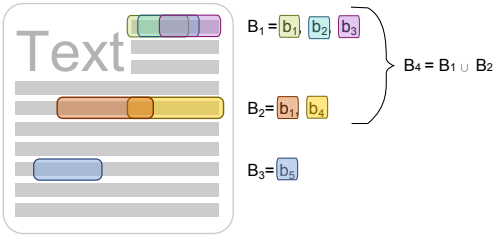


Abbildung 10.2.: Vereinigung überschneidender Textfenster

So wird gewährleistet, dass nur direkt aufeinanderfolgende Bezeichner in einer gemeinsamen Menge zusammengefasst werden. Das bewirkt die in Kriterium 2 genannte Einteilung in Sinnabschnitte. Diese sind jedoch noch nicht konsistent, solange Wiederholungen derselben Bezeichner nicht dem gleichen Sinnabschnitt zugeordnet wurden (siehe Zusammenführen wiederholter Bezeichner, Kriterium 3). Diese Zuordnung erfolgt im nächsten Prozessschritt. Hierzu werden alle Mengen überprüft, ob ein Bezeichner mehreren Mengen zugeordnet ist. Wird ein solcher Bezeichner identifiziert, so erfolgt eine Vereinigung der Mengen, *d.h.* iff $K_i \cap K_j \neq \emptyset$ then $K_i \cup K_j$ (K_i enthält hierbei alle Bezeichner R im Wortfenster i). Im Beispiel in Abbildung 10.2 ist dieser Fall gegeben. Da b_1 in B_1 und B_2 enthalten ist, werden diese zu B_4 zusammengefasst. Es ist in in Abbildung 10.3 dargestellt.

Nach dem Abschluss dieser Vorprozessierung erfolgt die Erstellung der Proposition basierend auf der Verwendung von Hintergrundwissen. Hierzu muss jeder Kontext einzeln prozessiert werden. Zu Beginn werden die einzelnen Intensionen Int_l der im lokalen Kontext vorhandenen Bezeichner $l \in K_i$ bestimmt und letztlich in die den Kontext repräsentierende Intensionsmenge $I_{K_i} = \sum_{l \in K_i} Int_l$ überführt (siehe Alg. 8, Zeile 4). Darauf folgt die kontextuelle Referenzbestimmung. Hierfür ist die Bestimmung des Steinerbaums notwendig, der diesen Kontext, *d.h.* die Kohärenz des lokalen Kontexts, beschreibt (siehe Alg. 8, Zeile 8). Den Unterschied zwischen dem Basisansatz und der hier vorgestellten Variante bildet zunächst die übergebene Menge an Bezeichnern, die je nach Kontext variiert. Nach Durchführung

der Analyse existiert für jeden lokalen Kontext eine Menge der durch den Algorithmus bestimmten Referenzen für die innerhalb des Kontexts vorhandenen Bezeichner, *d.h.* Sem(e) innerhalb des Lösungsbaums. Innerhalb dieses Lösungsbaums ist jedem Sem ein Gesamtaktivierungswert a_v zugeordnet, der die Relevanz dieses Knotens im Rahmen der lokalen Kohärenz beschreibt. Dieser Aktivierungswert ist von entscheidender Bedeutung für die weitere Analyse.

Schritt 1: Termidentifizierung

Schritt 2: Lokaler Kontext durch zugeordneten Textbereich
→ Identifizierung der Textbereiche

Schritt 3: Überprüfung auf Bereichsüberlappungen
→ Mikroebene

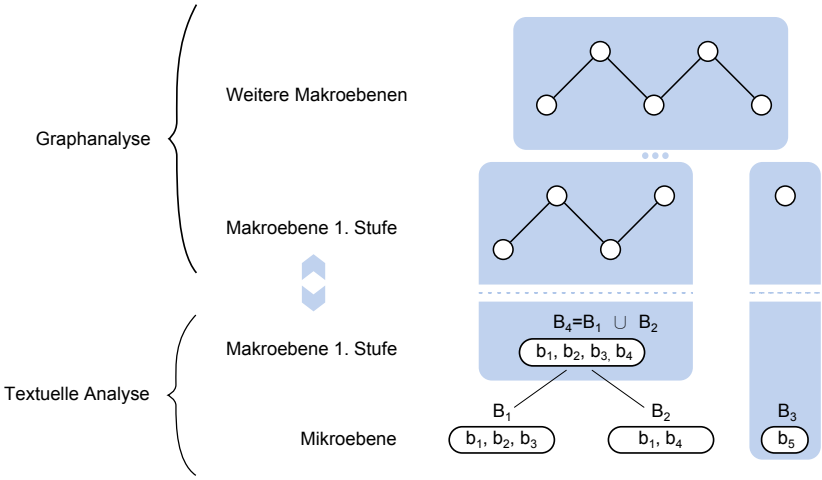


Abbildung 10.3.: Systematisches Vorgehen

In der Vorgehensweise von Kintsch et al. erfolgt ein iteratives Zusammenführen von Propositionen, gemäß der Darstellung in Abbildung 10.1 bzw. 10.3. Wie bereits vorgestellt erfolgt das Zusammenführen der Propositionen innerhalb des Basisansatzes implizit über die Exploration. Nachdem durch die obige Analyse lokaler Kohärenzen ein textueller Bezug geschaffen wurde, erfolgt deren implizite Aggregation zur Erstellung des Gesamtzusammenhangs. Hierzu gibt es zwei mögliche Vorgehensweisen: *(a)* die lokalen Kontextbäume

Funktion	Bedeutung
CALCINITIALACT(Int_l)	Berechnet die initiale Aktivierung für die Knoten, die in der übergegebenen Intension enthalten sind
RUNBASEALGORITHM(I_{K_i})	Ruft die in Kapitel 8 vorgestellte Basisvariante des Algorithmus auf. I_{K_i} gibt hierbei die Knoten vor, die als Seme verwendet werden sollen.

Tabelle 10.1.: Funktionen des Algorithmus

Algorithmus 8 : Algorithmus lokaler Kohärenz (Funktionsbeschreibung in Tabelle 10.1)

```
1  while K is non-empty do
2      foreach K_i ∈ K do
3          foreach l ∈ K_i do
4              estimate Int_l;
5              CALCINITIALACT(Int_l);
6              add Int_l to S_{K_i} and T;
7          end
8          R = RUNBASEALGORITHM(S_{K_i});
9          foreach v ∈ R do
10             get t ∈ T where id(t) = id(v);
11             a_{t,initial} = a(v);
12         end
13     end
14 end
15 RUNBASEALGORITHM(T)
```

werden iterativ erweitert[7], bis sich ein Gesamtzusammenhang ergibt bzw. *(b)* es erfolgt eine erneute Analyse gemäß der Vorgehensweise innerhalb des Basisansatz, jedoch mit dem Priorisieren der Seme gemäß deren Integration in den zuvor berechneten lokalen Kontexten. Die zweite Möglichkeit erlaubt zudem die Pfade von Beginn an neu zu explorieren. Es ist somit möglich den globalen Zusammenhang von Beginn an ebenfalls zu berücksichtigen - jedoch ausgehend von der Bedeutung der Seme, die in den lokalen Kontexten bestimmt wurde. Daher wird die zweite Variante in dieser Arbeit bevorzugt.[8] Um die Berücksichtigung der Bedeutung der einzelnen Kontexte von Beginn an zu gewährleisten, erfolgt bei dieser Verfahrensweise, *d.h.* der Neuprozessierung, die Änderung des initialen Aktivierungswerts der Knoten, die in den Lösungsbäumen der lokalen Kontexte enthalten sind $a_{t,initial}$ (Algorithmus 8, Zeile 10 bestimmt die Äquivalenz). Diesem wird somit der Gesamtaktivierungswert des Sems innerhalb des lokalen Kontexts zugewiesen (Algorithmus 8, Zeile 11). Hierdurch ergibt sich eine Priorisierung des Knotens bei der Selektion von Q.

Schnotz beschrieb, dass die einzelnen Propositionen zu einem Gesamtzusammenhang über semantische Relationen verbunden sind. Der hier vorgestellte Ansatz folgt dieser Vorstellung. Eine Proposition wird hierbei durch eine lokale Kohärenz dargestellt. Diese wird repräsentiert durch den zughörigen Lösungsbaum. Die lokalen Kohärenzen sind untereinander über semantische Relationen, die in der Ontologie definiert sind verbunden und bilden somit einen Gesamtzusammenhang. Der das Dokument und somit den Zusammenhang der lokalen Kohärenzen untereinander darstellende Resultatbaum hängt ab von den auf den lokalen Kohärenzen aufbauenden Gewichtung, die für die Exploration des Gesamtzusammenhangs verwendet wird.

[7] Es erfolgt ein sukzessives Zusammenführen je zweier Kontexte von Ebene zu Ebene.

[8] Erfolgt eine vollständige Überprüfung aller Pfade und Knoten innerhalb des den lokalen Kontext repräsentierenden Teilbaums, so kommt dies ebenfalls einer vollständigen Neuprozessierung gleich.

10.4. Anwendungsbeispiel Ansatz lokaler Kohärenz

Um den Unterschied zum Basisansatz zu verdeutlichen, wird der Ansatz der lokalen Kohärenz anhand des bereits in Abschnitt 8.4.6 eingeführten Beispiels vorgestellt. Innerhalb der Vorprozessierung werden zunächst die Wörter markiert, die den Bezeichnern von Ontologieinstanzen entsprechen. In Verbindung mit der in Abschnitt 13.2.2 vorgestellten geographischen Domänenontologie werden die Wörter *Arizona, Lake City, Florida, Christopher Creek, See Canyon* und *New Jersey* sowie deren Position innerhalb des Textes identifiziert. Jedes dieser Wörter definiert das Zentrum eines Textfensters mit vorgegebener Größe. Im vorgestellten Beispiel beträgt die Größe eines solchen Textfensters 7 Wörter. Innerhalb des nächsten Prozessschritts werden alle Textfenster hinsichtlich Überlappungen geprüft und im Falle von Überlappung werden die Textfenster zu einem gemeinsamen Fenster zusammengefügt. Somit werden alle in diesem Fenster enthaltenen Bezeichner einer diesen Kontext repräsentierenden Menge zugefügt.

Beispiel:

1. *"A wildfire in northern Arizona [...]" (Kontext 1)*
2. *"[...] a fire north of Lake City in Florida. Flames remained about a mile" (Kontext 2)*
3. *"from the community of Christopher Creek. The community is south of See Canyon [...]" (Kontext 3)*
4. *"[...] Elsewhere New Jersey [...]" (Kontext 4)*

In diesem Beispiel wurde in Textbereich 1 nur ein Bezeichner identifiziert, da innerhalb der Analyse kein Zusammenhang zu anderen Bezeichnern festgestellt wurde. Insofern enthält Kontext 1 außer *Arizona* keine weiteren Bezeichner. Dies steht im Gegensatz zu den Textausschnitten 2 und 3. Hier überlappen sich die Textfenster der jeweiligen Bezeichner. In Textausschnitt 2 überschneiden sich das Fenster *[fire, north, of, Lake City, in Florida, Flames]* und das Fenster *[of, Lake City, in, Florida, Flames remained, about]*. Somit wird Kontext 2 durch die Bezeichner *Lake City* und *Florida* definiert. Kontext 3 enthält anhand der

Überlappung beider Textfenster die Bezeichner *Christopher Creek* und *See Canyon* und Kontext 4 enthält *New Jersey*.

Nachdem die lokalen Kontexte identifiziert wurden, erfolgt die Untersuchung der lokalen Kohärenz je Kontext. Für jeden Kontext wird der Basisansatz separat ausgeführt. Hierbei bilden die bestimmten Bezeichner die Voraussetzung. Für diese Art der Bearbeitung muss ein Kontext *mindestens* zwei Bezeichner beinhalten. Abbildung 10.4 zeigt die algorithmische Verarbeitung.

Zunächst wird die lokale Kohärenz für Kontext 2 erstellt, diese ist durch den Graph links oben und den die Lösung repräsentierenden Steinerbaum (Schritt 1) dargestellt. Die lokale Kohärenz für Kontext 3 ist durch den Graph links unten visualisiert. Der zugehörige Steinerbaum ist rechts daneben (Schritt 2) aufgeführt. Die Steinerbäume enthalten jeweils die Resultate der kontextuellen Referenzbestimmung. Die Knoten, welche die jeweiligen Seme darstellen, besitzen hierbei einen Aktivierungswert, der sie gegenüber anderen Semen der gleichen Intension priorisiert. Nach der lokalen Kontextanalyse erfolgt nun eine Analyse des Gesamtkontextes über den Basisalgorithmus. Diese Vorgehensweise impliziert hierbei jedoch die Verwendung der Gesamtaktivierungswerte der bestimmten Referenzen im lokalen Kontext als initialen Aktivierungswert, der diesen entsprechenden Semknoten zu Beginn der Analyse des globalen Kontexts zugewiesen wurde. Der unter diesen Voraussetzungen explorierte Graph und resultierende Steinerbaum ist im rechten Teil der Abbildung dargestellt. Die unterschiedliche Vorgehensweise bei der Exploration wird beim Vergleich dieser Abbildung zur Darstellung des Anwendungsbeispiels des Basisalgorithmus 8.4.6 deutlich. Dort erfolgt die Exploration entlang der höchstaktivierten Knoten aufgrund ihrer Einbettung in den aktuellen Gesamtgraphen, während bei diesem Ansatz die Einbettung in die Lokalgraphen ein wesentlicher Faktor für deren Auswahl von Q und somit deren Priorität bei der Erstellung des Gesamtgraphen darstellt.

Der Unterschied der Varianten wird im Rahmen der Evaluation in Kapitel 13 aufgezeigt.

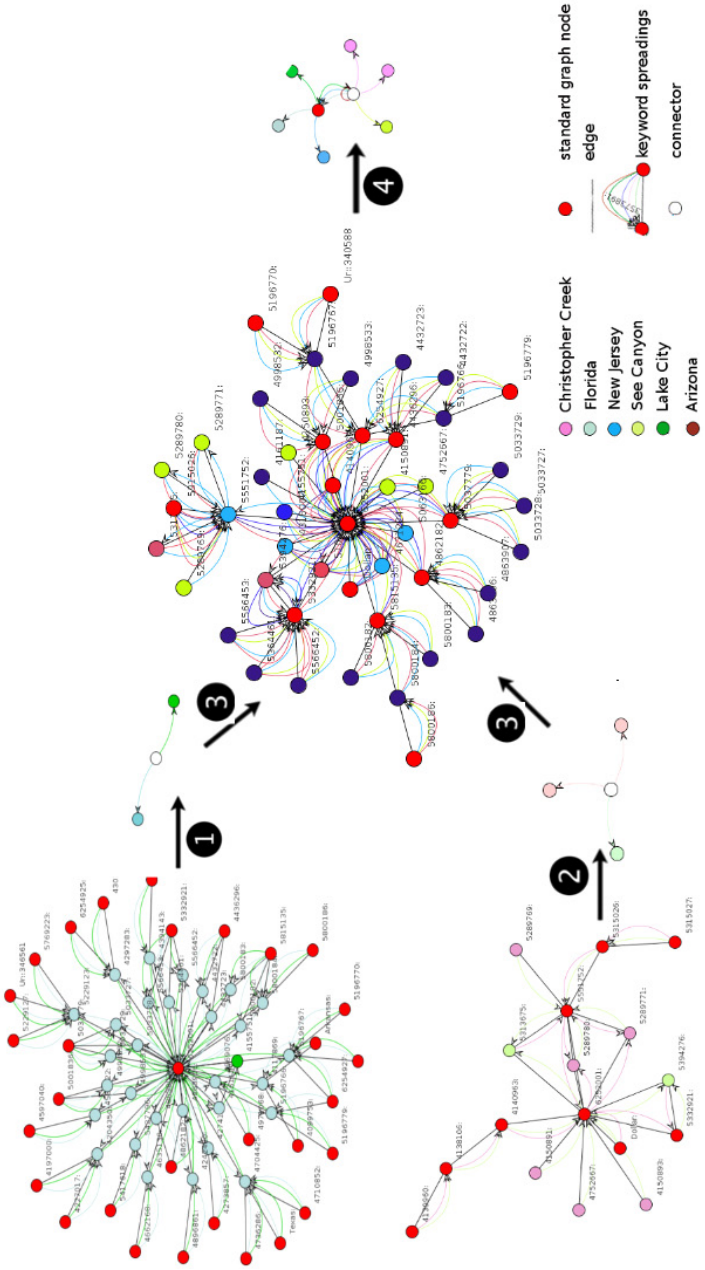


Abbildung 10.4.: Beispielausführung mit lokaler Kohärenz

11. Ansatz mit Bestärkendem Lernen

In der in Kapitel 8 vorgestellten Ausführung des Algorithmus werden die notwendigen Eingabeparameter beim Aufruf des Algorithmus durch die gegebene Ontologie sowie den übergebenen Text gesetzt. Dies gilt auch für die bisher vorgestellten Varianten des Basisalgorithmus. Hierbei werden keine Resultate vorheriger Algorithmusausführungen auf unterschiedlichen Texten herangezogen, um auf die Bearbeitung des aktuell vorliegenden Disambiguierungsproblems einen positiven Einfluss zu nehmen. Die in diesem Kapitel vorgestellte Modifikation des Basisalgorithmus ändert dies, indem hier der Einsatz eines Verfahrens des Bestärkenden Lernens[1] einsetzt. Die vorgestellte Methode ermöglicht somit die Berücksichtigung der selbsterzeugten Resultate früherer Disambiguierungen. Alternativ wird auch die Verwendung einer überwachten Datenmenge als Grundlage für die Lernfunktion, *d.h.* überwachtes Lernen[2], berücksichtigt.

In Abschnitt 11.1 werden zunächst die Grundlagen des Maschinellen Lernens, insbesondere des Bestärkenden Lernens, beschrieben. Anschließend wird in Abschnitt 11.2 das im Rahmen dieser Arbeit erstellte Verfahren vorgestellt und in Bezug zu den grundlegenden Elementen Bestärkenden Lernens gesetzt.

---

[1] *engl.* reinforcement learning; *„[...] Reinforcement learning [...] is concerned with the problem of finding suitable actions to take in a given situation in order to maximize a reward. Here the learning algorithm is not given examples of optimal outputs, in contrast to supervised learning, but must instead discover them by a process of trial and error. Typically there is a sequence of states and actions in which the learning algorithm is interacting with its environment."* [30] (siehe [224])

[2] *engl.* supervised learning; *„Applications in which the training data comprises examples of the input vectors along with their corresponding target vectors are known as supervised learning problems."* [30] (siehe auch *z.B.* [252])

11.1. Grundlagen des Bestärkenden Lernens

Allgemein liegt in diesem Lernverfahren die Idee zugrunde, dass aktuell vorliegende Situationen aufgrund von Erlerntem, *d.h.* zum Zeitpunkt der Analyse der jetzigen Situation (Eingabeparameter) bereits das vorhandene Wissen beurteilt und gemäß einer mit diesem Wissen verbundenen Vorgehensweise erreicht werden können. Die Annahme ist, dass durch die mit Wissen verbundene Vorgehensweise bessere Ergebnisse[3] durch den Algorithmus erreicht wird. Abhängig von der konkreten Methode des maschinellen Lernens variiert die Strategie des Lernens. Beispielsweise verwendet überwachtes Lernen eine gegebene Menge von Daten, die zuvor auf Korrektheit überprüft wurden.

Im Gegensatz zu den meisten Verfahren überwachten Lernens, die eine abgeschlossene Menge zur Erstellung von Hypothesen verwenden, baut bestärkendes Lernen auf einer iterativen Erstellung von Gesetzmäßigkeiten, *d.h.* Regeln, auf. „Iterativ“ weist darauf hin, dass im Laufe der algorithmischen Analyse dieses Modell korrigiert und erweitert wird. Somit ermöglichen die zuvor durchgeführten Analysen eine Beeinflussung der algorithmischen Analyse der aktuell vorliegenden Eingabewerte. Sutton und Barto beschreiben dies durch die Aussage: *„An Agent must be able to learn from its own experience“*. Dies bedeutet: *„a) explore what he already knows“* und *„b) explore to make better action selections“* [224].

Ebenfalls definierten diese Autoren die vier grundlegenden Elemente, über die ein solcher Algorithmus verfügen sollte:

1. eine Strategie
2. eine Funktion zur Beurteilung bzw. zum Honorieren von Reaktionsweisen
3. eine Gewichtungsfunktion
4. ein Modell der Domäne (optional)

[3] Der Term „besseres Ergebnis“ bezieht sich auf eine zur Überprüfung des Algorithmus durchgeführte Evaluation.

Punkt 1 beschreibt eine Strategie, *d.h.* einen Vorgehensplan, die vorgibt wie auf eine gegebene Situation reagiert werden muss. Die Honorierungsfunktion (Punkt 2) beschreibt eine Vorgehensweise zur Ermittlung der Güte, welche die ausgewählte Reaktion (Punkt 1) im Rahmen des Vorgehensmodells zur Folge hat.[4] Dies kann ebenfalls Anlass zu einer Änderung der Strategie geben und enthält gegebenenfalls die Art und Weise der Reaktionsänderung. Punkt 3 beschreibt die Gewichtung von Vorgehensweisen. Diese orientiert sich an deren Auswirkung über mehrere Anwendungen hinweg und somit deren Nutzen auf längere Sicht. Das Modell der Domäne (Punkt 4) bildet die Abläufe innerhalb des gegebenen Arbeitsgebietes ab, *d.h.* die Situationen, die während der Verarbeitung auftreten können und wie darauf reagiert werden kann. Solche Modelle ermöglichen unter anderem die Erstellung und Anwendung von Algorithmen, die in der Lage sind, die Wahrscheinlichkeit hinsichtlich des Auftretens von Situationen und mögliche Konsequenzen, je nach Reaktionsweise, zu bestimmen.

11.2. Bestärkendes Lernen im Basisansatz

Das Verfahren hat die Bestimmung der Referenz des korrekten Ontologieelements ausgehend von einem vorliegenden natürlichsprachlichen Bezeichner unter besonderer Berücksichtigung des Problems der Ambiguität zur Aufgabe. Es setzt einen vorhandenen Textkorpus als auch eine die Domäne beschreibende Ontologie voraus. Die Information, die aus dem Text ausgewertet wird, fokussiert sich auf die Erkennung der dort enthaltenen Entitätsbezeichner. Die Ontologie beschreibt die vorhandene Domäne als Wissensmodell. Dieses ist jedoch zu unterscheiden von einem Verfahrensmodell hinsichtlich einer algorithmischen Vorgehensweise.

[4] *D.h.* ob die durch die Strategie ausgewählte Regel einen positiven oder negativen Effekt auf die weitere Durchführung des algorithmischen Prozesses ausübt.

Im Folgenden sind die wesentlichen *Verfahrensschritte* des vorgestellten Basisalgorithmus angegeben.

1. Extraktion von Entitätsbezeichnern
2. Bestimmung der Seme in der Ontologie, *d.h.* der Adressen innerhalb der Intension des Bezeichners
3. Bestimmung möglicher Steinerbäume
4. Bestimmung der Referenz(en) je gegebenen Entitätsbezeichner aus dem höchstgewichteten Steinerbaum.

Die zunächst genannte Extraktion von Bezeichnern benannter Entitäten (Punkt 1) wird für jedes Dokument des gegebenen Textkorpus durchgeführt, *d.h.* bei einer iterativen Bearbeitung kommt es mit großer Wahrscheinlichkeit zum wiederholten Auftauchen gleicher Bezeichner. Unter der Annahme, dass mit der Wiederholung von Bezeichnern auch eine Tendenz zu der Bestimmung der gleichen Referenzen für diesen Bezeichner einhergeht, führt dies zu der Konsequenz, dass innerhalb der Intension eines Bezeichners bestimmte Seme wahrscheinlicher sind als andere (Punkt 2). Zugleich kann dies in die Bestimmung möglicher Steinerbäume im Spreading-Activation-Verfahren einfließen. Wesentliches Kriterium stellt in diesem Verfahren die Aktivierung eines Knotens dar. Daraus folgt, dass solchen Semen ein höherer bezeichnerspezifischer Aktivierungswert zugeordnet ist. Dieser muss jedoch den Grad der Wahrscheinlichkeit wiederspiegeln (Punkt 3). Dies nimmt implizit Einfluss auf die Bestimmung der Lösungsreferenz, da diese anhand des Aktivierungswerts bestimmt wird (Schritt 4).

Diese Annahmen basieren darauf, dass eine Domäne ein abgeschlossenes Fachgebiet bezeichnet. Es stellt eine Spezialisierung (bzw. ein Teilgebiet) gegenüber einer allgemeinen Wissensbasis dar. Die Definition einer Ontologie (vgl. Abschnitt 3.2) beinhaltet die Zuordnung der Ontologie zu einer Domäne, *d.h.* das Wissen, das durch die Ontologie

dargestellt wird, betrifft ein abgeschlossenes Fachgebiet.[5] Dieser Sachverhalt ist von besonderer Bedeutung, da er Anlass dazu gibt, dass eine Textbasis, die sich auf dieses Wissen bezieht, mit hoher Wahrscheinlichkeit mehrere Texte beinhaltet, die gleiche oder ähnliche Inhalte beschreiben. Dokumente, die teils wiederholende Bezeichner zu Referenzzuordnungen enthalten, sind wahrscheinlicher innerhalb von Texten des gleichen Fachgebietes[6], als in Texten, die sich auf allgemeines Wissen beziehen. Ein typisches Beispiel sind Texte, die das gleiche Ereignis beschreiben. Betrachtet man solche Dokumente, so enthalten zeitlich später erscheinende Artikel oftmals zusätzliche Informationen, die zuvor nicht bekannt waren. Das deutet mit hoher Wahrscheinlichkeit auf dieselben und somit wiederkehrende Referenzen für dieselben Bezeichner hin, die in den aufeinander folgenden Artikeln genannt werden. Dieser Sachverhalt ist unabhängig vom Grad der Ambiguität innerhalb der Ontologie.

Aus dem vorgestellten Sachverhalt ergibt sich die im Folgenden vorgestellte Umsetzung des Ansatzes mit bestärkendem Lernen. Das wird anhand der zuvor aufgeführten vier Grundelemente eines Ansatzes zum bestärkenden Lernen (Abschnitt 11.1) vorgestellt:

Strategie Die Strategie analysiert zunächst die vorliegende Menge an Entitätsbezeichnern. Für jedes Sem n_i innerhalb der Intension Int_t eines Bezeichners t erfolgt eine Überprüfung des Fundus des bereits erlernten Wissens, der durch zuvor durch geführte Monosemierungsprozesse erzeugt wurde. Diese Überprüfung untersucht ob und in welchem Zusammenhang das untersuchte Sem bereits als Referenz für t bestimmt wurde und ermittelt dadurch die Wahrscheinlichkeit für dieses Sem, dass es gleichzeitig die im Kontext des vorliegenden Dokumentes gesuchte Referenz für diesen Bezeichner darstellt.

---

5 Es existieren Ontologien, die diese Spezialisierung nicht aufweisen. Beispielsweise beschreibt die DBPedia Ontologie (Ontologie zugreifbar unter `http://downloads.dbpedia.org/3.7/dbpedia_3.7.owl.bz2` [letzter Zugriff am 18.11.2011], Erläuterung unter `http://wiki.dbpedia.org/Datasets` [letzter Zugriff am 18.11.2011]) das in der freien Enzyklopädie Wikipedia (`http://www.wikipedia.org` [letzter Zugriff am 12.09.2011]) enthaltene Wissen. Diese hat den Anspruch ein allgemeines und nicht ein auf eine Domäne beschränktes Nachschlagewerk zu sein. Für weitere Informationen zu DBpedia wird auf den initialen Artikel von Bizer et al. [32] verwiesen.

6 Im Zusammenhang mit einer Ontologie, die allgemeines Wissen beschreibt, kann dies ebenfalls durch eine zuvor erfolgte Kategorisierung erreicht werden, *z.B.* eine thematische Ordnung (Clustering) nach Themengebieten (*z.B.* Politik, Sport *etc.*).

Die detaillierte Vorgehensweise ist in Abbildung 11.1 dargestellt. Der linke Bereich zeigt die Vorgehensmethode, die ohne Bestärkendes Lernen vom Algortihmus verfolgt wird (Initialisierungsphase des Basisalgorithmus 8.4.2). Für jeden Bezeichner wird die Intension bestimmt. Jedem Sem wird eine initiale Aktivierung für den repräsentierten Bezeichner (*z.B.* für Knoten n_1 als Sem des Bezeichners t die Aktivierung $a_{n_1,t}$) gemäß der initialen Aktivierungsfunktion (Abschnitt 12.1) zugeordnet. Im Rahmen der Modifikation durch den Einsatz von Bestärkendem Lernen (rechter Bereich der Abbildung 11.1) erfolgt eine Neugewichtung anhand der bisher erfahrenen Informationen in den zuvor durchgeführten Disambiguierungen. Die Kriterien hierfür sind im unteren Abschnitt *Gewichtung* dargestellt. Die Gewichtung ist innerhalb des Algorithmus zum einen verantwortlich für die Auswahl des nächsten zu prozessierenden Knotens von Q und zum anderen für die Auswahl des Knotens als Repräsentant der Referenz.

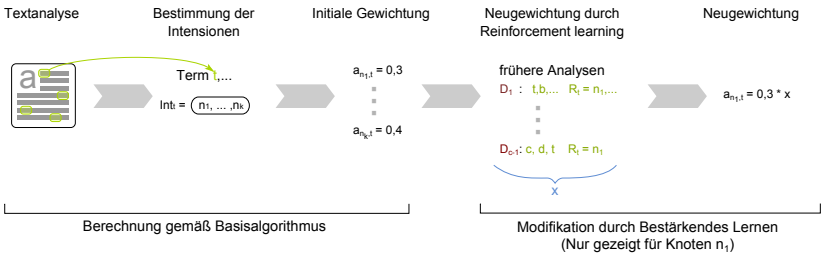


Abbildung 11.1.: Strategie des Ansatzes bestärkendes Lernen

Honorierung Eine Honorierung, *d.h.* die Beurteilung des Erfolgs bzw. Misserfolgs einer angewandten Strategie, lässt sich dadurch bestimmen, ob die durch den Ansatz des bestärkenden Lernens, am wahrscheinlichsten die bestimmte Referenz auch als endgültige Referenz für den Entitätsbezeichner im Resultat enthalten ist. Der hier vorgestellte Algorithmus verwendet als Honorierung diese Information, da ein zusätzliches Resultat direkt zur Bestimmung der Gewichtung des Sems bei einer erneuten Textprozessierung verwendet wird. Somit wird im Falle, dass das Ergebnis dieser Referenzbestimmung mit diesem Sem übereinstimmt, diesem ein höheres Gewicht zugeordnet und die Auswahl dieses Sems bei einer später erfolgenden Analyse eines neuen Dokumentes begünstigt. Andernfalls wird das Gewicht abgeschwächt und somit die Wahrscheinlichkeit für dessen Auswahl reduziert.

Gewichtung Die Gewichtung eines Sems hängt von der Anzahl der durchgeführten Analysen ab, bei denen das Sem als Referenz für den gegebenen Entitätsbezeichner bestimmt wurde. Ohne weitere Faktoren steht die alleinige Anzahl für die Präsenz des Sems als Referenz des Bezeichners im Zusammenhang mit dem Textkorpus. Insofern werden von allen zuvor analysierten Dokumenten *Rt* in denen der Bezeichner *t* disambiguiert wurde, diejenigen Dokumente $R_{n_{1,t}}$ ausfindig gemacht, in denen der Knoten *n1* als Referenz für *t* bestimmt wurde. Jedoch ist dieses Merkmal allein ungenügend, da es der Situation, die durch die Kohärenz aller Bezeichner eines Dokumentes ausgedrückt wird, nicht gerecht wird. Um diese nachzubilden, wird eine individuelle Gewichtung jedes zuvor berechneten Resultats durchgeführt. Die Höhe der bestimmten Aktivierung für eine Referenz ist hierbei abhängig von (vgl. Gleichungen (12.9)):

1. Wie viele Entitätsbezeichner des aktuell zu prozessierenden Dokumentes waren in diesem Dokument enthalten?
2. Wie viele zusätzliche Entitätsbezeichner sind enthalten?

Die Anzahl der übereinstimmenden Entitätsbezeichner gibt Hinweis darauf, ob ähnliche Zusammenhänge im zu untersuchenden Dokument r und im bereits zuvor prozessierten Dokument d beschrieben werden, die auf die gleichen Entitäten schließen lassen. L_r bezeichnet die Menge der Bezeichner innerhalb des Dokuments r und L_d die Menge der Bezeichner innerhalb des Dokuments d. Die Schnittmenge $L_r \cap L_d$ gibt somit die Menge der übereinstimmenden, *d.h.* der sowohl in Dokument r als auch in Dokument d vorkommenden, Bezeichner an. Das bedeutet, dass die Kohärenz beider Dokumente durch die Schnittmenge beschrieben wird. Insofern bestätigt dies, dass die Wahrscheinlichkeit desselben Sems für einen gegeben Bezeichner in beiden Dokument zunimmt, je mehr Bezeichner innerhalb der Dokumente übereinstimmen. Beispielsweise weisen zwei Texte in denen der Bezeichner „Madita“ vorkommt noch keinen Hinweis auf Übereinstimmung der möglichen Referenz auf. Sind jedoch in beiden Texten die Bezeichner „Madita, Offenburg, Joachim“ enthalten, so lässt dies bereits weniger Raum für unterschiedliche Referenzen, da die Informationsdichte durch die Anzahl gleicher Bezeichner zunimmt. Während Punkt 1 die Anzahl der übereinstimmenden Entitätsbezeichner hervorhebt, kann dennoch davon ausgegangen werden, dass

zusätzliche Bezeichner im zu untersuchenden Dokument mehr Möglichkeiten für unterschiedliche Referenzen offen lassen. Das ist in Punkt 2 beschrieben, da diese gegebenenfalls unterschiedliche Zusammenhänge beschreiben und somit keine Kohärenz zwischen beiden Texten besteht, *d.h.* die Menge der übereinstimmenden Bezeichner wird in das Verhältnis zu allen Wörtern im zu untersuchenden Dokument gesetzt, *d.h.* $\frac{L_r \cap L_d}{L_r}$.

Es gibt Varianten des bestärkenden Lernens, die auf Grundlagen des überwachten Lernens zurückgreifen. In dem hier vorgestellten Verfahren gibt es ebenfalls eine solche Alternative. Diese greift iterativ auf die zur Evaluation benutzten, *d.h.* die zuvor durch den Benutzer auf Korrektheit überprüften, Ergebnisse zurück, anstatt auf die Ergebnisse, die durch den Algorithmus selbst erstellt wurden.

Modell In Abschnitt 11.1 wird die Verwendung eines Modells als *optional* bezeichnet. Für das gegebene Verfahren wird in Einklang damit kein Vorgehensmodell verwendet. Als alleiniges Modell kommt das durch die Ontologie vorgegebeneWissensmodell zum Einsatz.

Zusammenfassend liegt der Hauptunterschied dieser Variante zum in Kapitel 8 vorgestellten Basisansatz in der veränderten initialen Gewichtung der Knoten. Durch Bestärkendes Lernen wird es ermöglicht, anhand bereits zuvor durchgeführter Disambiguierungsverfahren bestimmte Seme der Intension anderen gegenüber vorzuziehen. Die detaillierte Beschreibung des mathematischen Verfahrens zur Gewichtungsberechnung basierend auf Bestärkendem Lernen ist im Kontext der anderen Maßfunktionen in Abschnitt 12.3 beschrieben. Der Erfolg der Methode wird im Rahmen der Evaluation in Kapitel 13 besprochen.

12. Maße, Parameter und Heuristiken

In den Kapiteln zuvor wird das grundlegende Vorgehen sowie die algorithmischen Varianten beschrieben, die darauf aufbauen. Jedoch wird der Detailgrad hinsichtlich der verwendeten Maße hierbei reduziert, um die Erläuterung der jeweiligen Techniken[1] in den Vordergrund zu stellen. Dieses Kapitel enthält die Maße und Parameter, die zur Berechnung der Aktivierungswerte innerhalb des Ansatzes verwendet werden. Die Qualität des Ansatzes steigt und sinkt basierend auf den Aspekten, die bei dieser Berechnung herangezogen werden.[2]

Abschnitt 12.1 beschreibt die Maße, die bei Berechnung der Bedeutung einer Instanz Anwendung finden, *d.h.* deren Gewichtung und die Weitergabe der Aktivierungswerte beim Spreading. In Abschnitt 12.2 wird die Bestimmung entitätsspezifischer Zusammenhänge ausgehend von den Entitätsbezeichnern beschrieben, während in Abschnitt 12.3 die Maße für den Ansatz des Bestärkenden Lernens vorgestellt werden. Abschnitt 12.4 führt die Gewichtung von Kanten (Object-Properties) ein und in Abschnitt 12.5 wird auf die Abbruchkriterien, *d.h.* das Stoppen der weiteren Exploration, innerhalb des Algorithmus eingegangen.

[1] Dies bezieht sich auf die Erläuterungen in den Kapiteln 8,9,10 und 11.

[2] Eine genaue Darstellung der Effektivität der einzelnen Aspekte befindet sich in Kapitel 13, das eine Evaluation des Ansatzes enthält.

12.1. Bestimmung der Instanzspezifischen Aktivierungswerte

Die Gewichtung einer Instanz, *d.h.* eines Knotens innerhalb des Instanzgraphen[3], ist von entscheidender Bedeutung für deren Berücksichtigung innerhalb des Algorithmus. Je Instanz gibt es drei verschiedene Typen von Aktivierungswerten: *1) ein Aktvierungswert, der die Wertigkeit*[4] *der Instanz unabhängig von Bezeichnern angibt*, 2) *der Aktivierungswert der Instanz je Bezeichner* und *3) der Gesamtaktivierungswert*, der aus den ersten beiden zusammengesetzt wird.

Die Zuweisung eines initialen instanzspezifischen Aktivierungswerts $nodePrestige(v)$ ermöglicht die Festlegung der Wertigkeit einer Instanz *unabhängig* von deren Stellung zu den zu berücksichtigenden Bezeichnern (siehe Tabelle 12.1 für eine Übersicht der verwendeten Variablen). Ein solches Maß kommt ebenfalls in den Ansätzen von Bhalotia [24, 1] und Kacholia [118] zum Einsatz. Typische Maße sind *indegree* und *outdegree* des Knoten v. Eine weitere Möglichkeit ist die Gewichtung nach Konzeptzugehörigkeit durch dieses Maß. Beispielsweise können im Fall einer Geoontologie Instanzen von Städten gegenüber denen von Dörfern bevorzugt werden.[5] Eine Übersicht ist in [229] zu finden. Dieser Wert $nodePrestige(v)$ wird in die Berechnung der Gesamtaktivierung a_v (Gleichung (12.3)) miteinbezogen. Sie bestimmt die Position des Knotens in der Queue Q und somit dessen Platz in der Explorationsreihenfolge. Ein von den Bezeichnern unabhängiges Kriterium zur Knotenbewertung stellt beispielsweise die allgemeine Einbettung der Instanz in die Domäne dar und kann durch die Anzahl, die Art der Properties oder über die Zuordnung zu einem vorgegebenen Konzept bestimmt werden.

---

[3] Innerhalb diesem Abschnitt werden die Begriffe „Knoten“ und „Instanz“ alternativ verwendet, da innerhalb des Instanzgraphen alle Knoten für Instanzen stehen.

[4] Definiert die Bedeutung des Knotens für das Verfahren unabhängig vom Bezeichner. Beispielsweise können Instanzen bestimmter Konzepte dadurch eine höher Relevanz zugewiesen werden.

[5] Hierbei ist der Zusammenhang zur Informationsdarstellung innerhalb des Korpus entscheidend.

$$a_{v,l} = \frac{proximity(v,l,i)}{|Int_l|} \tag{12.1}$$

Gleichung 12.1: Initiale Sem Aktivierung (Einsatz: Initialisierungsphase)

Entscheidender für das Verfahren ist die Zuweisung der bezeichnerspezifischen Aktivierungswerte. Ein solcher Aktivierungswert wird einer Instanz vor Ausführung des Algorithmus initial zugewiesen (siehe Gleichung (12.1)), falls es sich bei ihr um ein Sem in der bezeichnerspezifischen Intension Int_l handelt. Ausschlaggebend für einen Knoten (Sem) $v \in Int_l$ ist zunächst dessen individueller Bezug zum gegebenen Bezeichner, der durch die Funktion $proximity(v,l,i)$ bestimmt wird.[6] Umsetzungsmöglichkeiten dieser Funktion sind in Abschnitt 12.2 dargestellt. Zur Normierung erfolgt eine Division durch die Größe der Intension $|Int_l|$ für den Bezeichner l. Hintergrund dieser Division ist die Tatsache, dass je weniger Seme in dieser Menge enthalten sind, desto größer ist die Wahrscheinlichkeit, dass es sich bei einem darin enthaltenen Sem um die gesuchte Referenz handelt.

$$a_{u,l} = a_{v,l} \times \lambda \times P(e_{v,u}) \tag{12.2}$$

Gleichung 12.2: Berechnung der Aktivierungsweitergabe (Einsatz: Spreading der bezeichnerspezifischen Aktivierung)

Neben der initialen Zuweisung von bezeichnerspezifischen Aktivierungswerten erfolgt die Weitergabe von Aktivierungen während des Spreading-Activation-Verfahrens (siehe Gleichung (12.2)[7]).Bei der Exploration einer Kantenverbindung zwischen zwei Knoten v und u

[6] Je nach Art und Weise der Textanalyse kann hierbei auch der Vergleich mit dem ursprünglich im Text vorhandenen Wort vorgenommen werden, das zum Nachschlagen im Lexikon verwendet wurde.

[7] Diese Formel kommt in Algorithmus 4 in den Zeilen 25 bzw. 26 zur Anwendung.

wird die Aktivierung für l von v nach u via der Kante e weitergegeben.[8] Zum einen erfährt der Aktivierungswert hierbei eine Abschwächung durch den Faktor λ[9], der die Aktivierung im Verhältnis zu der weitergegebenen Distanz zwischen Sem und Knoten reduziert (*z.B.* $\lambda = 0.8$). Zum anderen erfolgt eine Bewertung der Verbindung zwischen beiden Knoten basierend auf der vorliegenden Kante mittels der Funktion $P(e_{v,u})$. Mögliche Umsetzungen dieser Funktion werden in Abschnitt 12.4 vorgestellt.

$$a_v = \sum_{l \in L} a_{v,l} + nodePrestige(v) \tag{12.3}$$

Gleichung 12.3: Gesamtaktivierung (Einsatz: Bestimmung der Position innerhalb der Queue Q)

Die in Gleichung (12.3)[10] dargestellte Berechnung der Gesamtaktivierung setzt sich zusammen aus der Summe des je Bezeichner individuell zugewiesenen Aktivierungswerts, als auch dem initial zum Knoten zugeordneten $nodePrestige(v)$. Der Gesamtaktivierungswert a_v bestimmt die Position innerhalb der Queue Q. Eine Verbindung zu mehreren Semen kann sich hierbei als vorteilhaft erweisen. Der Rang innerhalb dieser Queue ist insbesondere von Bedeutung, falls der Algorithmus über Abbruchbedingungen gestoppt wird, bevor Q vollständig exploriert wurde. Zudem beeinflusst es das Ranking der Ergebnisse, da die Knoten, die die Resultate (*d.h.* die Wurzeln der Steinerbäume) repräsentieren, ebenfalls anhand der Gesamtaktivierung beurteilt werden.

[8] Dies geschieht im Basisverfahren unter der Voraussetzung, dass der Knoten v von Q selektiert wurde und nun exploriert wird. Beim bidirektionalen Verfahren kann es innerhalb dieses Explorationsschritts, der ursprünglich von v ausging, auch um den Knoten u handeln.

[9] Der Wertebereich der Variable λ beträgt $0 < \lambda < 1$. Je nach gesetztem Wert nimmt der zu übertragende Aktivierungswert von Spreadingvorgang zu Spreadingvorgang ab.

[10] Diese Formel kommt in Algorithmus 2 in der Zeile 8 zur Anwendung.

Variable	Bedeutung
$a_{v,l}$	Aktivierung für Entitätsbezeichner l bei Knoten v
$proximity(v,l,i)$	Zusammenhang zwischen Label i und Entitätsbezeichner l bei Knoten v
Int_l	Intension für Entitätsbezeichner l
a_v	Gesamtaktivierung von Knoten v
$nodePrestige(v)$	Gewichtung Knoten v unabhängig von Entitätsbezeichnern
Q	Queue für Exploration (geordnet nach Gesamtaktivierungen)
λ	Reduktionsvariable für eine Verringerung des Wertes bei Weitergabe des Aktivierungswerts
$P_{e_{v,u}}$	Funktion für Kantengewichtung
i	Label eines Knotens
R_l	Zuvor analysierte Dokumente die den Entitätsbezeichner l enthalten
d	Dokument
$t_{d,R_{l,v}}$	Zuvor analysierte Dokumente bei denen der Knoten v als Referenz des Entitätsbezeichners l bestimmt wurde
$\gamma_{d,r}$	Schnittmenge der Bezeichner in Dokument r und Dokument d

Tabelle 12.1.: Verwendete Variablen

12.2. Aktivierung anhand Entitäts- und Knotenbezeichner

Neben der $nodePrestige(v)$-Funktion bietet die Funktion $proximity(v,l,i)$ die Möglichkeit der Zuweisung eines Aktivierungswerts, der explizit unter Berücksichtigung der individuellen Merkmale eines Knotens berechnet wird. Die Funktion $proximity(v,l,i)$ bestimmt den Zusammenhang des der Instanz zugeordneten Labels zum gegebenen Entitätsbezeichner. Mögliche Wege dies umzusetzen sind die Verwendung textspezifischer Maße, welche die

Schreibweisen miteinander vergleichen, und/oder Heuristiken, die diesen Zusammenhang bewerten.

Im Rahmen dieser Arbeit wird auf die Levenshtein Distanz [137] zurückgegriffen. Dieser Algorithmus erfuhr weite Verbreitung und wird verwendet, um die Ähnlichkeit zweier Textfragmente zu beurteilen. Navarrro bezeichnet ihn als *„the most important measure of similarity“* [159]. Levenshtein erlaubt die Berücksichtigung der Möglichkeit von Buchstabenersetzungen, deren Löschen sowie Neueinfügungen. Dies unterscheidet es von anderen ebenfalls verwendbaren Maßen, wie *z.B.* der Hamming-Distanz [198] (Ersetzen) und der Episode Distanz [58] (Löschen). Für eine Übersicht der verschiedenen Maße wird auf die Artikel von Navarro [159], Manivannan et al. [148] und das Buch von Manning et al. [149] verwiesen.

Neben dieser Überprüfung via textspezifischen Maßes, die gegebenenfalls zu einer Abwertung aufgrund von Rechtschreibfehlern führt, wird kontrolliert, ob der Entitätsbezeichner mit dem Wert der einer Instanz zugewiesenen `rdf:alterLabel` Data Property übereinstimmt. Diese Übereinstimmung führt zu einer Abwertung, da angenommen wird, dass solche Instanzen nicht primär für den Entitätsbezeichner stehen.[11]

Eine weitere Möglichkeit ist die Berücksichtigung von Heuristiken. Beispielsweise können Konzeptbezeichner (*z.B.* „GmbH“ für Firmen) in der direkten Umgebung eines Instanzbezeichners lokalisiert werden und dementsprechend Adressen dieses Konzepttyps in der Intension des Entitätsbezeichners höher gewichtet werden.

12.3. Bestärkendes Lernen

Der Ansatz des Bestärkenden Lernens wurde in Kapitel 11 eingeführt. Im Folgenden sind die dem Ansatz zugrundeliegenden mathematischen Berechnungen dargestellt. Der Ansatz beruht darauf, dass je gegebenem Bezeichner $l \in L$ innerhalb des gegebenen Dokuments d überprüft wird, ob dieser bereits in zuvor durchgeführten Analysen berücksichtigt wurde. Im Fall, dass in vorherigen Analysen eine

[11] Dies ist nicht zutreffend, falls der `rdf:label`-Wert ebenfalls dem Bezeichner entspricht.

Referenz (bzw. Referenzen) für diesen bestimmt wurden, wird dies bei der aktuellen Analyse berücksichtigt. Dies geschieht dadurch, dass die Initialgewichtung für diesen Entitätsbezeichner für einen Knoten $v \in Int_l$, der zuvor als Referenz für l bestimmt wurde, neu berechnet wird. Diese Neuberechnung hat eine Änderung der initialen Gewichtung für diesen Knoten zur Folge.

Die Berechnung ist in Gleichung (12.4) dargestellt. Die (neue) Gewichtung des Knotens v für den Bezeichner l setzt sich dabei zusammen aus einem Maß basierend auf den Dokumenten, in denen dieser Knoten zuvor als Referenz bestimmt wurde $(t_{d,R_{l,v}})$ und dem zuvor vorgestellten Textvergleich ($proximity(v,l,i)$). Somit wird die in Gleichung (12.1) vorgestellte Initialaktivierung vollständig ersetzt durch die im Folgenden vorgestellte Gleichung (12.4). Beides wird in das entsprechende Verhältnis gesetzt, *d.h.* $proximity(v,l,i)$ zu der Größe der aktuellen Intension $|Int_l|$ und $(t_{d,R_{l,v}})$ zur Menge der zuvor analysierten Dokumente, die diesen Entitätsbezeichner enthalten $|R_l|$.

$$a_{v,l} = \frac{t_{d,R_{l,v}}}{|R_l|} + \frac{proximity(v,l,i)}{|Int_l|} \tag{12.4}$$

Gleichung 12.4: Initiale Sem Aktivierung (Einsatz: Initialisierungsphase bei Reinforcement Learning)

Der Parameter $\gamma_{d,r}$ (12.5a) beschreibt das spezifische Resultat für den Knoten v im Zusammenhang mit dem Dokument r des Resultatsets $R_{l,v}$. Bei $R_{l,v}$ handelt es sich um die Teilmenge von R_l ($R_{l,v} \subseteq R_l$), welches die Dokumente enthält, bei denen v als Referenz für l bestimmt wurde. Der Wert von $\gamma_{d,r}$ wird gebildet durch die Berücksichtigung der Menge der im aktuell zu analysierenden Dokument d enthaltenen Entitätsbezeichner L_d, die gleichzeitig auch in den Entitätsbezeichnern L_r des zu überprüfenden Dokument r des Resultatsets $R_{l,v}$ enthalten sind. Beispielsweise liegt ein solche Situation vor, dass wenn im aktuell zu überprüfenden Dokument 5 Entitätsbezeichner enthalten sind und im zu analysierenden Dokument r, welches bereits zuvor dem Prozess der Monosemierung unterlag, sind 3 dieser 5 Bezeichner ebenfalls vorhanden. Daraus ergibt sich der Schluss, dass diese Dokumente eine gegenseitige Ähnlichkeit von über 50% aufweisen ($\frac{3}{5} = 0,6$).

$$\gamma_{d,r} = \frac{|L_d \cap L_r|}{L_r|} \tag{12.5a}$$

$$t_{d,R_{l,v}} = \sum_{r=0}^{R_{l,v}} \gamma_{d,r} \tag{12.5b}$$

Gleichung 12.5: Bezeichnerspezifische Gewichtung, die auf vorherigen Resultaten basiert (Einsatz: Initiale Bezeichnerpezifische Aktivierung des Sem bei Reinforcement Learning)

Das knotenspezifische Resultat, das durch die Analyse der zuvor überprüften Dokumente errechnet wird, ist in Gleichung (12.5b) beschrieben. Hier werden alle vorherigen Resultate, die je Dokument $r \in R_{l,v}$ in Gleichung (12.5a) bestimmt wurden, aufsummiert. Somit beschreibt $t_{d,R_{l,v}}$ die Anzahl der Dokumente, in denen der Knoten v als Referenz für den Entitätsbezeichner l bestimmt wurde. Je exakter die gegebenen Entitätsbzeichner L_d mit den enthaltenen Entitätsbezeichnern L_r übereinstimmen, desto eher ist davon auszugehen, dass die zuvor bestimmten Ergebnisse für das Dokument r auch für das aktuelle Dokument d gelten. Daher werden Dokumente innerhalb der Menge der Resultate bevorzugt, die möglichst die gleichen Bezeichner wie das aktuell zu analysierende Dokument enthalten. Dies bedeutet im Zusammenhang mit dem zuvor erwähnten Beispiel, dass das aktuell analysierte Dokument aussagekräftiger für den Knoten v ist, als ein Dokument in dem die Übereinstimmung der Bezeichner nur 30% beträgt.

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^{-2}} dt \tag{12.6a}$$

$$a_{v,l} = erf(a_{v,l}) \tag{12.6b}$$

Gleichung 12.6: Normalisierung mittels Gaußscher Fehlerfunktion (Einsatz: Initale Berechnung der Bezeichneraktivierung des Sem bei Reinforcement Learning)

Da der neuberechnete Wert $a_{v,l}$ gegebenenfalls 1 übersteigen kann, ist eine Normalisierung notwendig. Diese betrifft alle initialen Aktivierungen, *d.h.* auch der Knoten, die in keinem zuvor analysierten Dokument enthalten waren. Hierzu wurde die Gaußsche Fehlerfunktion, die eine Sigmoid-Kurve durch den 0-Punkt beschreibt, gewählt (siehe 12.6). Diese gewährleistet die Einhaltung des Wertebereichs zwischen 0 und 1 sowie die unterschiedlichen Größenverhältnisse der Aktivierungen.

12.4. Kantenmaße

Ontologien bieten Data- und Object-Properties, um die Eigenschaften einer Instanz zu beschreiben. Data-Properties sind nur bedingt geeignet, um einen Bezug zwischen Instanzen auszudrücken.[12] Object-Properties hingegen werden dazu verwendet, direkte Beziehungen zwischen Ontologieelementen aufzuzeigen und bilden die Kanten innerhalb des Instanzgraphen. Bei jeder Exploration erlaubt die Berücksichtigung der Object-Property daher eine individuelle Gewichtung des Knotenzusammenhangs. Dies erfolgt über die zuvor bereits vorgestellte Funktion $P(e_{v,u})$.

Im Folgenden wird die Berechnung eines heuristischen (Abschnitt 12.4.1) sowie semantischen (Abschnitt 12.4.2) Maßes für einen gegebenen Zusammenhang vorgestellt.

12.4.1. Heuristisches Maß

Das hier vorgestellte Maß setzt die individuelle Object-Property zwischen zwei Instanzen in Bezug zu anderen Object-Properties, die dem aktuell untersuchten Knoten zugeordnet sind. Dies geschieht über die Betrachtung des Kantentyps. Der Knoten v ist hierbei der aktuell explorierte Knoten. Die Anzahl der Kanten zwischen den

[12] Data-Properties sind nur bedingt geeignet, da Zuammenhänge hierbei abhängig von Datenwerten sind. Die Auswertung einer solchen Übereinstimmung ist im Allgemeinen nicht durch die Ontologie selbst gegeben, sondern das Resultat eines darauf angewandten Verfahrens (*z.B.* die Gruppierung von Instanzen, die den gleichen Wert für ein bzw. mehrere Properties aufweisen (Clustering)). Der Einsatz muss daher im Einzelfall sehr genau abgewogen werden.

Knoten v und u, die dem Typ der gegebenen Kante e entsprechen ($|E_{v,u}|$), werden in das Verhältnis aller Kanten dieses Typs gesetzt, die v zugeordnet sind ($|E_v|$). Die Güte der individuellen Zuordnung zwischen v und u wird somit durch Gleichung (12.7) berechnet.

$$P(e_{v,u}) = \frac{|E_{v,u}|}{|E_v|} \tag{12.7}$$

Gleichung 12.7: Heuristisches Kantenmaß (Einsatz: Kantenmaß bei Berechnung des Spreading Activation Wertes)

Mittels dieser Heuristik wird eine Annahme getroffen, von welcher Relevanz diese Beziehung in Anbetracht der Menge von weiteren Relationen des gleichen Typs ist. Ein Beispiel, bezogen auf die zuvor bereits erwähnte Ontologie `geonames.org` [letzter Zugriff am 12.09.2011], ist das Verhältnis zwischen der Geoinstanz, welche die Vereinigten Staaten von Amerika bezeichnet v und derjenigen, die den Staat Florida bezeichnet u. Die Kante e ist durch `geo:inCountry` gegeben. Wird v exploriert, so wird $P(e_{v,u})$ durch $\frac{1}{1886307}$ berechnet. Grund hierfür ist, dass sehr viele Instanzen in Amerika liegen. Wird u hingegen exploriert, so wird $P(e_{u,v})$ durch $\frac{1}{1}$ berechnet. Durch die beiden Werte wird der Unterschied in der Wertigkeit der Zuordnung deutlich, der bei dieser Heuristik durch die Anzahl der Kanten des vorliegenden Kantentyps normiert wird. Zudem wird aufgezeigt, dass die Explorationsrichtung hierbei entscheidend ist.

12.4.2. Semantisches Maß

Die Beurteilung der Güte einer Object-Property wird bei einem „Semantischen Maß“ durch die Berücksichtigung sowohl konzeptueller Zusammenhänge (T-Box)[13] als auch der Zusammenhänge auf der Instanzebene (A-Box) erzielt. Mögliche Vorgehensweisen finden sich im Forschungsbereich *Ranking von Ergebnissen semantischer Suche, z.B.* [5, 11]. Ranking beschäftigt sich mit der Ordnung des Suchergebnisses, *d.h.* mit dem Priorisieren von Resultaten innerhalb der

[13] Eine Übersicht von Maßen zur Bestimmung von Konzeptähnlichkeiten ist in Artikel [133] beschrieben.

Ergebnisliste. Semantische Suche bedarf somit eines semantischen Maßes, um die einzelnen Suchergebnisse zu bewerten.

Der Unterschied zum vorliegenden Verfahren ist, dass beim Ranking eine Bewertung nach Abschluss der Suche vorgenommen wird. Für das im Rahmen dieser Arbeit entwickelte Verfahren ist es jedoch erforderlich, dass die Bewertung im Laufe der Exploration vorgenommen werden kann. Bei der Analyse verschiedener Verfahren wird deutlich, dass der von Anyanwu, Maduko und Sheth entwickelte Ansatz „SemRank" [10] hierfür geeignete Methoden bietet. Es wird hierbei nur auf Teile des Verfahrens zurückgegriffen, da auf die Analyse vollständiger Pfade verzichtet wird.[14] Der Ansatz verwendet eine Kombination informationstheoretischer Methoden und Heuristiken. Die verwendeten Gewichtungen werden im Folgenden dargestellt. Für eine vollständige Darstellung wird auf den Artikel [10] verwiesen.

Die Methode ermittelt den Informationsgewinn, der durch eine Object-Property erzielt werden kann. Anayanwu et al. orientieren sich hierbei an den Prinzipien der Informationstheorie, welche die Information eines Ereignisses durch den negativen Logarithmus der Wahrscheinlichkeit dessen Eintritts bestimmt. Der Ansatz baut auf der Ermittlung der Spezifität einer Object-Property auf, die in Gleichung (12.8) beschrieben ist.

$$I_s(p) = -log(Pr(\chi = p)) = -log(\frac{|[[p]]^\wedge|}{|[[P]]^\wedge|}),\ p \in P \tag{12.8}$$

Gleichung 12.8: Bestimmung der Spezifizität einer Object-Property hinsichtlich der gesamten Wissensbasis

Hierbei sind die Parameter definiert durch:

a) $[[c]]^\wedge = \{r | r \text{ ist } \texttt{rdf:type} \text{ of } c\}$, *d.h.* nur direkte Instanzen von c

b) $[[p]]^\wedge = [r_1, r_2] | r_1 \in \{[[c]]^\wedge | c \in p.domain\}, r_2 \in \{[[c]]^\wedge | c \in p.range\}$, *d.h.* nur direkte Instanzen von p

[14] Das geht auf die Tatsache zurück, dass die Methode Speading Activation bereits implizit eine Bewertung des Pfades vergleichbar zu PageRank vornimmt. Siehe auch die Anmerkungen hierzu in Abschnitt 8.4.3.

Die Spezifität einer Property wird ermittelt, indem zunächst die Anzahl ihrer Verwendung in der Wissensbasis bestimmt wird, *d.h.* wie häufig diese zur Verbindung von Instanzen innerhalb der gesamten Wissensbasis zum Einsatz kommt. Diese wird in das Verhältnis zur Häufigkeit des Einsatzes aller Object-Properties in der Wissensbasis gesetzt.

Anyanwu et al. nehmen eine Konkretisierung (siehe Gleichung 12.9b) vor. Diese fokussiert auf die Situation zwischen zwei Konzepten r_1 und r_2. Bei Bewertung der Spezifität einer Object-Property p zwischen diesen zwei Konzepten hängt diese zunächst von der Anzahl $|[[p]]^\wedge|$ der Instanzen der beiden Konzepte ab, zwischen denen die Object-Property p definiert wurde. Des Weiteren werden Sub-Properties transitiv ihrer jeweils allgemeinsten Super-Property, *d.h.* in der hierarchischen Gliederung zuerst stehende Property, zugeordnet und die Menge aller Instanzen, die den Properties zwischen beiden Konzepten, die diesem Vererbungsbaum angehören, wird durch θ beschrieben (siehe Beispiel unten). Hierbei gilt $p \in \theta$. Die Anzahl der Verbindungen aller Object-Properties $|[[\theta]]^\wedge|$ innerhalb von θ bildet den Wert zur Normierung.

$$Pr(\chi = p | \chi \in \theta) = \frac{Pr(\chi = p, \chi \in \theta)}{Pr(\chi \in \theta)} = \frac{|[[p]]^\wedge|}{|[[\theta]]^\wedge|} \tag{12.9a}$$

$$I_{\theta - s}(p) = I(\chi = p | \chi \in \theta) = -logPr(\chi = p | \chi \in \theta) \tag{12.9b}$$

Gleichung 12.9: Semantisches Kantenmaß (Einsatz: Kantenmaß bei Berechnung des Spreading Activation Werts)

Diese Berechnung ist in Gleichung (12.9a) dargestellt, *d.h.* die Häufigkeit der Object-Property p im Verhältnis zu der Häufigkeit der Properties in θ. Dieses Maß erlaubt somit die Bewertung einer Object-Property im Rahmen der konzeptuellen Zuordnung von r_1 und r_2, *d.h.* zur Menge der weiteren Object-Properties zwischen diesen Konzepten.

Angenommen, r_1 ist dem Konzept `Bürgermeister` zugeordnet, während r_2 dem Konzept `Stadt` zugehörig ist. Es existieren folgende Super-Object-Properties in $\theta = \{$`repräsentiert`, `wohntIn`$\}$. Es gibt 15 Instanzen von `repräsentiert` und 20 von `wohntIn`. Insgesamt gibt

es 100 Object-Property-Instanzen. Daraus folgt für `repräsentiert` die Wahrscheinlichkeit von 0.15 im Verhältnis zu allen Properties und $\frac{15}{15+20} = 0.428571429$ im Verhältnis zu θ.

Für das Maß $P(e_{u,v})$ wird die Gleichung (12.9b) verwendet ($e = p$).

12.5. Abbruchkriterien für den Algorithmus

Ohne die Verwendung von Kriterien zur vorzeitigen Beendigung der algorithmischen Ausführung stoppt der Algorithmus erst, nachdem alle möglichen Resultate berechnet wurden. Mögliche heuristische Abbruchbedingungen sind das Setzen eines *a) minimalen Aktivierungswerts* (ω)[15] und/oder einer *b) maximalen Explorationstiefe* (η)[16]. Eine maximale Explorationstiefe beschränkt die Größe des Steinerbaums, der ermittelt werden kann, *d.h.* der längste Pfad darin kann maximal die Länge 2η besitzen. Dieses Maß ist unabhängig von der Güte des Baumes, die durch die Aktivierungswerte der jeweiligen Entitätsbezeichner dargestellt wird, die dem den Baum repräsentierenden Knoten zugeordnet sind (Gleichung (12.3)). Aufgrund der unterschiedlichen Aktivierungsverteilung können auch größere Bäume, *d.h.* die einen längeren maximalen Pfad enthalten, ein teilweise besseres Resultat erzielen, als kleinere Bäume. Die Verwendung eines minimalen Aktivierungswerts ω kommt dieser Tatsache nach. Ist die berechnete weiterzugebende Aktivierung hinsichtlich eines Entitätsbezeichners vom Knoten v zum Knoten u kleiner als die minimale Aktivierung, *d.h.* $a_{u,l} < \omega$, so wird diese nicht weitergegeben.[17] Dadurch erfolgt ein Stopp der Ausbreitung der Aktivierung für diesen Entitätsbezeichner. Eine zusätzliche Einschränkung des Explorationsprozess kann durch weitere Heuristiken vorgenommen werden. Der Algorithmus 2 (Zeile 20) sieht das Maß deg_{max} vor, um Knoten von der Analyse auszuschließen, die mehr ein- und ausgehenden Kanten besitzen als durch diese Beschränkung vorgegeben.

Unabhängig von den vorgestellten heuristischen Abbruchbedingungen kann ein Abbruch auch aufgrund einer Beurteilung der in der

[15] Im Algorithmus 2 mit $\omega = a_{min}$ in Zeile 8 verwendet.
[16] Im Algorithmus 2 mit $\eta = depth_{max}$ in Zeile 8 verwendet.
[17] Siehe Gleichung (12.2) zur Weitergabe des Aktivierungswerts.

Zwischenzeit erstellten Resultate vorgenommen werden. Ein verbreiteter Ansatz ist der Abbruch der Analyse, nachdem ein Resultat durch x darauffolgend errechnete Resultate nicht verbessert werden konnte. Der Algorithmus bietet diese Möglichkeit durch das Setzen des *maximumDeviance*-Parameters (Algorithmus 5, Zeile 68). Jedoch gilt es die Verwendung abzuwägen, da gegebenenfalls doch noch ein besseres Ergebnis hätte berechnet werden können. Im Fall, dass die Aktivierung allein vom Distanzmaß abhängig ist, ist dies eine valide Vorgehensmethode. Der Ansatz des Spreading Activation deutet darauf hin, dass das beste Ergebnis innerhalb einer Distanzbeschränkung auch zuerst aufgefunden wird. Dies liegt daran, dass die Exploration immer den besten Aktivierungswerten folgt und diese höher sind je kürzer die Distanz zu den Semen ist. Sind jedoch weitere Parameter involviert ($nodePrestige(v)$, $P(e_{u,v})$ *etc.*), so hängt es von diesen ab, ob der bestbewertete Resultatbaum ebenfalls durch die kürzesten Pfaden zu den Semen charakterisiert ist.

Teil III.

Evaluation und verwandte Arbeiten

13. Evaluation

In diesem Kapitel wird die Evaluation des in Kapitel 8 vorgestellten Algorithmus sowie aller vorgestellten Varianten erörtert. Die Evaluation bringt die Güte der einzelnen Verfahren zum Ausdruck, indem sie die Resultate anhand eines mathematischen Maßes bewertet und ein Kriterium der Vergleichbarkeit schafft. Voraussetzung für eine Evaluation im Zusammenhang mit dem beschriebenen Verfahren ist die Verwendung einer Hintergrundontologie, welche die semantischen Zusammenhänge beschreibt und eines Textkorpus, dessen Entitäten in dieser Ontologie beschrieben sind.

Zunächst werden die theoretischen Grundlagen des Evaluationsprozesses in Abschnitt 13.1 eingeführt. Danach werden in Abschnitt 13.2 die Grundlagen der beiden durchgeführten Fallstudien vorgestellt. Die erste verwendet eine Ontologie, die eine allgemeine Wissensbasis darstellt. Die zweite eine Ontologie, die auf geographische Lokationen fokussiert. Beide Fallstudien werden zunächst hinsichtlich der verwendeten Ontologie und Textbasis vorgestellt. Im Anschluss daran werden in Abschnitt 13.3 die spezifischen Testergebnisse je Studie vorgestellt und analysiert. Abschnitt 13.4 beschreibt die daraus resultierenden Schlussfolgerungen.

13.1. Evaluationsprozess

Der Evaluationsprozess hat die Aufgabe, den in dieser Arbeit vorgestellten Ansatz mit vorhandenen Systemen zu vergleichen. Gleichzeitig soll er jedoch auch über die Praxistauglichkeit hinsichtlich der gegebenen Komplexität Auskunft geben. Die in diesem Kapitel vorgestellte Evaluation stellt mit der ersten Fallstudie eine Vergleichbarkeit zum Disambiguationsansatz

von Nguyen [163] her. Der Ansatz von Nguyen wurde aufgrund seiner gleichen Zielsetzung – der Disambiguierung mehrdeutiger Entitätsbezeichner – ausgewählt. Der Ansatz selbst wird in Kapitel 14 „Verwandte Arbeiten“ näher beschrieben. Die verwendete Ontologie und die zugehörigen Daten werden in Abschnitt 13.2 vorgestellt. Die Voraussetzung für vergleichbare Systeme ist, dass deren Verfahren ebenfalls eine Ontologie zur Disambiguierung von Daten verwenden. Dies bedeutet, dass die Ambiguität innerhalb eines Dokuments auf die Ontologie zurückgeführt werden kann und dort aufgelöst werden muss. Der Ansatz von Nguyen ist den Recherchen des Autors dieser Arbeit zur Folge ein hierfür geeigneter Ansatz, da dieser mittels eines Vektorenvergleichs zwischen in Texten enthaltenen ambiguen Entitätsbezeichnern und deren möglichen Äquivalenzen in der Ontologie eine Disambiguierung vornimmt. Diese grundlegende Art und Weise des mathematischen Prozessierens textueller Inhalte beruht auf deren Repräsentation in Form von Vektoren (vgl. [149]). Daher ermöglicht ein Vergleich des Verfahrens von Nguyen und dem des Autors dieser Arbeit zugleich eine Gegenüberstellung der verwendeten mathematischen Modelle.

Explizit wird davon abgesehen das vorgestellte Verfahren mit Ansätzen zur Disambiguierung zu vergleichen, die auf Datensätzen ohne Ontologie basieren. Für die Anwendung des im Rahmen dieser Arbeit vorgestellten Verfahrens ist die Zuordnung einer Ontologie zum zugrundeliegenden Datensatz Voraussetzung. Ist jedoch durch das Vergleichsverfahren keine Ontologie vorgegeben, muss die zu verwendende Ontologie selbst gewählt werden. Durch die Zuordnung einer selbst gewählten Ontologie für die Evaluation ergibt sich allerdings eine Fehlerquelle, die eine Vergleichbarkeit der Resultate nicht mehr gewährleistet.[1] Hassel et al. [106] beschreiben dies durch die Aussage *„the up-to-date status of an ontology can have an impact on the quality of the disambiguation results“*, *d.h.* nicht nur die Ontologie, sondern auch die vorliegende Version von dieser können das Resultat beeinflussen. Insbesondere Ansätze zur Disambiguierung, die auf Wikipedia

[1] Die Entitäten des Textes müssen dieser Ontologie zunächst zugeordnet werden. Während die Disambiguierung vieler verwandter Verfahren allein auf Textfeatures basiert, steigt und sinkt die Qualität des hier vorgestellten Verfahrens mit der verwendeten Ontologie. Dies bedeutet, dass das Verfahren nicht allein den kritischen Punkt darstellt, sondern insbesondere die Ontologie. Insofern ist ein valider wissenschaftlicher Vergleich, der auf den gleichen Voraussetzungen der Verfahren basiert, nicht möglich.

basieren (*z.B.* [55, 154, 131, 139, 79]) fallen in die Kategorie von Verfahren ohne Ontologie. Zwar gibt es die Möglichkeit der Verwendung von DBpedia, der Ontologie basierend auf Wikipediadaten, diese enthält nur eine Klasse von Properties. Sie basiert auf den PageLinks in Wikipedia-Artikeln. PageLinks stellen Beziehungen zwischen der in Wikipedia erklärten Entität und allen auf der zugehörigen Website referenzierten Entitäten dar. Diese Art der Assoziation lässt jedoch keine getypten Schlüsse zu, *z.B.* <*HelmutSchmidt pagelink Hamburg*> oder <*HelmutSchmidt pagelink Vegesack*>. Es lässt sich aufgrund dieser Art der Assoziation *z.B.* nicht herausfinden, dass er in Hamburg geboren wurde und in Vegesack seinen Wehrdienst geleistet hat. Dieser Umstand erlaubt keine Anwendung der in Kapitel 12 vorgestellten Maße (*z.B.* heuristische und semantische Property-Bewertung) und gewährleistet zudem - wie zuvor dargestellt - keine Vergleichbarkeit mit Ansätzen, die nicht diese Ontologie zur Disambiguierung verwenden.[2]

Zusätzlich zu der Vergleichbarkeit zu anderen Ansätzen hat die Evaluation auch die Aufgabe einen Eindruck der Leistungsfähigkeit des Ansatzes in Anbetracht einer großen und äußerst ambiguen Datenbasis zu geben. Dieser Nachweis ist durch die Untersuchung der zweiten Fallstudie gegeben, die auf geographische Daten ausgerichtet ist und wird in Abschnitt 13.2 vorgestellt.

Die innerhalb der Fallstudien verwendeten Daten stellen eine Assoziation zwischen Entitäten der Texte und den diese repräsentierenden Instanzen in der jeweiligen Ontologie zur Verfügung. Die Evaluation erfolgt anhand von Resultaten für einen gegebenen Textkorpus, *d.h.* jeder Text wird auf Entitäten untersucht. Diese werden mithilfe des vorgestellten Verfahrens disambiguiert und das Ergebnis für jedes Dokument anschließend gespeichert. Basierend auf dem Vergleich dieser Ergebnisse zu manuell überprüften Referenzergebnissen wird die Qualität des Ansatzes evaluiert.

13.1.1. Evaluationsmaße

Zur Evaluation der erzielten Ergebnisse werden die Standardmaße im Forschungsgebiet „Information Retrieval“, *Precision* (13.1a) und

[2] Es konnten leider keine auf DBPedia basierten Disambiguierungverfahren festgestellt werden.

Recall (13.1b) verwendet. Voraussetzung zur Anwendung dieser Maße ist ein vorhandenes Referenzergebnis, mit dem das durch das Programm erzielte Ergebnis verglichen werden kann. Diese Referenzergebnisse werden im Normalfall durch Menschen erstellt. Beim vorliegenden Evaluationsszenario ist ein Referenzergebnis durch die Instanzen gegeben, welche die relevanten Referenzen (*relevant entities*) für die vom Algorithmus aufgefundenen Entitätsbezeichner (*retrieved entities*) vorgeben.

Precision P (13.1a) beschreibt das Verhältnis der zuvor vorgegeben, als relevant bezeichneten Referenzen zur Menge der vom Verfahren aufgefundenen Referenzen, *d.h.* das Maß beschreibt die Korrektheit des aufgefundenen Resultats. Recall R (13.1b) hingegen beschreibt das Verhältnis der korrekt identifizierten Referenzen durch den Algorithmus zu allen relevanten Referenzen, die das Referenzergebnis vorgibt. Somit beschreibt das Maß die Anzahl der gesuchten Referenzen, die aufgefunden werden, während *Precision* zusätzlich noch auf die Anzahl der Referenzen je Entität Bezug nimmt.

$$P := \frac{|relevant\ entities \cap retrieved\ entities|}{|retrieved\ entities|} \tag{13.1a}$$

$$R := \frac{|relevant\ entities \cap retrieved\ entities|}{|relevant\ entities|} \tag{13.1b}$$

Zusätzlich zu diesen individuellen Maßen existieren verschiedene kombinierte Maße. Eines der am weitesten verbreiteten Maße ist die sog. F-Measure (13.2). Dieses beschreibt das *harmonische Mittel* zwischen den beiden oben genannten Maßen, *d.h.* diese sind hierbei gleich gewichtet. Informationen zu diesen Maßen finden sich in [234, 145].

$$f_{\text{measure}} := \frac{2 * R * P}{R + P} \tag{13.2}$$

Tritt derselbe Bezeichner mehrfach im selben Text auf, so erfordert dieser Fall eine konsistente Vorgehensweise. Im Rahmen dieser Arbeit wird davon ausgegangen, dass es sich bei einer solchen Mehrfachnennung durchweg um die gleiche Entität handelt. Diese weitverbreitete

Vorgehensweise wurde in der Arbeit „One Sense per Discourse“ [87] von Gale et al. näher erörtert.

13.2. Grundlagen der Fallstudien

Dieser Abschnitt stellt beide Fallstudien vor, die für die Evaluation verwendet wurden. Jede von ihnen besteht aus einer Ontologie, welche das Wissensmodell der Domäne repräsentiert, und einem Textkorpus, welcher die zu verwendenden Dokumente beinhaltet. Die in den Dokumenten enthaltenen Entitäten müssen ebenfalls in der Ontologie enthalten sein.[3]

13.2.1. Fallstudie 1

Die KIM (**K**nowledge and **I**nformation **M**anagement)[4] - Plattform wurde entwickelt, um die innerhalb von Texten erwähnten Informationen mit denen in Ontologien zusammenzubringen. Eine in die Plattform integrierte Applikation erlaubt die automatische Annotation von Wörtern im Text mit Ontologieelementen. Ebenfalls erlaubt sie manuelles Hinzufügen neuer Ontologieelemente, eine Suche und eine graphische Ontologienavigation. KIM wurde von der Firma Ontotext[5] entwickelt. Nguyen verwendete die Ontologie der Plattform sowie die Named Entity Recognition Komponente der integrierten Applikation, um die Entitäten innerhalb der Texte seines Testkorpus zu erkennen.

Ontologie

KIM verwendet die PROTON-Ontologie[6] (**PROT**o **ON**tology) [227], die speziell zur Beschreibung Benannter Entitäten im Bereich von allgemeinen Nachrichtentexten entwickelt wurde. Sie repräsentiert

[3] Bei der Evaluation werden nur Entitätsbezeichner berücksichtigt, die dem Label von mindestens einem Ontologieelement zugeordnet werden können, *d.h.* durch `rdf:label` oder `rdf:alterLabel` bezeichnet wurden.

[4] `http://www.ontotext.com/kim` [letzter Zugriff am 12.09.2011]

[5] `http://www.ontotext.com/` [letzter Zugriff am 12.09.2011]

[6] `http://proton.semanticweb.org/` [letzter Zugriff am 12.09.2011]

keine spezifische Domäne, sondern Allgemeinwissen aus den Bereichen Sport, Politik und Finanzen. Insgesamt enthält sie zirka 300 Konzepte und 100 Properties. Hauptsächlich werden die Entitäten in untersuchten Textkorpora den Konzepten *„Person, Location, Organisation, Money (Amount), Date etc.“* [227] zugewiesen.

Die Instanzbasis der Ontologie enthält zirka 77.500 Entitäten, denen zirka 110.000 Namen zugewiesen sind. Weiterhin sind in ihr geographische Orte abgelegt, sowie eine Vielzahl von Organisationen. Die genauen Zahlen sind im Anhang B.1 aufgeführt.

Die von Nguyen verwendete Ontologie konnte auf Anfrage nicht mehr zur Verfügung gestellt werden. Der Autor dieser Arbeit verwendet daher den aktuellen Stand der Ontologie, der unter der Adresse `http://www.ontotext.com/kim/ontologies` [letzter Zugriff am 12.09.2011] bezogen wurde. Die Ontologie besitzt keine Informationen über die drei Entitäten mit dem Namen „John McCarthy“. Dies war bereits bei Nguyen der Fall. Dieser hat die erforderlichen Informationen selbst zur Ontologie hinzugefügt. Um welche Informationen es sich hierbei handelt wurde von ihm im Artikel [163] nicht vorgestellt und konnte auch auf Nachfrage nicht rekonstruiert werden. Daher wurden vom Autor dieser Arbeit notwendige Informationen zur Ontologie hinzugefügt. Die exakte Auflistung dieser Tripel ist in Anhang B.2 gegeben.[7]

Textkorpus

Der Textkorpus wurde von Nguyen für den in seinem Artikel [163] durchgeführten Test zusammengestellt und dem Autor dieser Arbeit auf Anfrage zur Verfügung gestellt.[8] Es handelt sich hierbei um drei verschiedene Datensets, die von verschiedenen online Nachrichtenservices, *z.B.* BBC, CNN *etc.*, stammen. Insgesamt sind 140 Dokumente enthalten, die nach drei Entitätsbezeichnern aufgeteilt sind. Die genaue Verteilung der Entitäten auf die Dokumente im Korpus ist in Tabelle 13.1 dargestellt.

[7] Diese Informationen betreffen die drei Entitäten mit dem Bezeichner „John McCarthy“. Aufgrund fehlender Ontologieinformationen wurden ebenfalls zwei Tripel für eine Entität mit dem Namen „Georgia“ hinzugefügt.

[8] Der Autor dieser Arbeit möchte sich an dieser Stelle nochmals ausdrücklich hierfür bedanken.

Entität	Anzahl Dokumente im jeweiligen Teilkorpus	Vorkommen in den Dokumenten
John McCarthy	28	73
Georgia	72	407
Columbia	40	109

Tabelle 13.1.: Dokumente je Entitätsbezeichner

Entität	Anzahl der Dokumente
John McCarthy (Compter Scientist)	17
John McCarthy (Linguist)	5
John McCarthy (UFC Referee)	3
Georgia (USA)	30
Georgia (Country next to Russia)	21
Georgia (Country in North America)	30
Columbia (Canada)	20
Columbia (Sportswear Company)	6
Columbia (University (USA))	5
Columbia (City (USA))	3
Columbia (Columbia District (Washington, USA))	6

Tabelle 13.2.: Entitätsübersicht

Bei den Entitätsbezeichnern handelt es sich um *„John McCarthy“*, *„Georgia“* und *„Columbia“*. Im Rahmen der Ontologie weist der Namen *„John McCarthy“* auf 3 Entitäten (Personen) hin. Es sind 3 Entitäten mit dem Namen *„Georgia“* (Lokationen) und 7 mit dem Namen *„Columbia“* (4 Organisationen und 3 Lokationen) in der Ontologie enthalten. Deren Verteilung im Korpus, *d.h.* die in den Texten verwendeten Entitäten, ist in Tabelle 13.2 angegeben. Zusätzlich zu den Dokumenten, die nur eine Entität für einen der gegebenen Bezeichner beschreiben, sind Dokumente enthalten, die mehrere Entitäten für einen Bezeichner referenzieren, *z.B. „Columbia“* als Sportswear und als Distrikt.

13.2.2. Fallstudie 2

Geonames `http://www.geonames.org` [letzter Zugriff am 12.09.2011] wurde ursprünglich von Marc Wick entwickelt und ist mittlerweile eine bekannte Plattform für geographische Informationen. Die zur Verfügung gestellte Datenbasis vereint viele bekannte Datenquellen, *z.B.* die Daten der National Geospatial-Intelligence Agency (Geolokationen außerhalb der USA), der U.S. Geological Survey Geographic Names Information System (Auswahl geographischer Daten innerhalb der USA) sowie 44 weiterer Quellen. Die Webseite erlaubt die Suche nach Orten, Städten *etc.* sowie das Hinzufügen oder Ändern von Informationen. Täglich werden zirka 20 Millionen Anfragen[9] an die Seite gerichtet.

Ontologie

Die Ontologie für Geonames[10] wurde von Bernard Vatant entworfen. Sie enthält 645 Subkategorien, die 9 Oberklassen zugeordnet sind. Diese sind *Administrative Region, Hydrography, Location, Populated Place, Road, Spot Place, Hypsography, Undersea* und *Vegetation*. Die Ontologie ist auf die Domäne Geographie beschränkt und in der Konsequenz sind ausschließlich geographische Entitäten in ihr enthalten.

Die Ontologie enthält über 10 Millionen Namen. Es wurden 5,5 Millionen alternative Namen für Instanzen verwendet, *d.h.* die mittels `rdfs:alterLabel` zugewiesen wurden.[11] Der Ontologie sind 7,5 Millionen Instanzen zugewiesen, von denen 2,8 Millionen Populated Places (Städte, Orte, *etc.*) beschreiben. Betrachtet man die Anzahl der Namen und die umfangreiche Verwendung der Möglichkeit alternative Namen zuzuweisen, so wird deutlich, dass diese Ontologie eine Vielzahl an ambiguen Entitätsbezeichnern enthält. Die exakte Verteilung ist in Abbildung 13.1 aufgezeigt. Zirka 6 Millionen Namen sind eindeutig vergeben. Teilweise sind Namen enthalten, denen über 255 Entitäten und mehr zugewiesen sind, *z.B.* 255 Entitäten für „Florida“, 2085 Entitäten für „First Baptist Church“. Die Ontologie ist im RDF-Format verfügbar und besteht aus *„60 million triples“* [31]. Zudem ist

9 `http://www.geonames.org/about.html` [letzter Zugriff am 12.09.2011]

10 `http://www.geonames.org/ontology` [letzter Zugriff am 12.09.2011]

11 Diese Namen wurden zum Teil ebenfalls als primäre Bezeichner verwendet.

sie Teil der Linked Open Data Cloud [32, 236]. Dieses Projekt hat die Vernetzung von öffentlich verfügbaren Daten zur Aufgabe.[12]

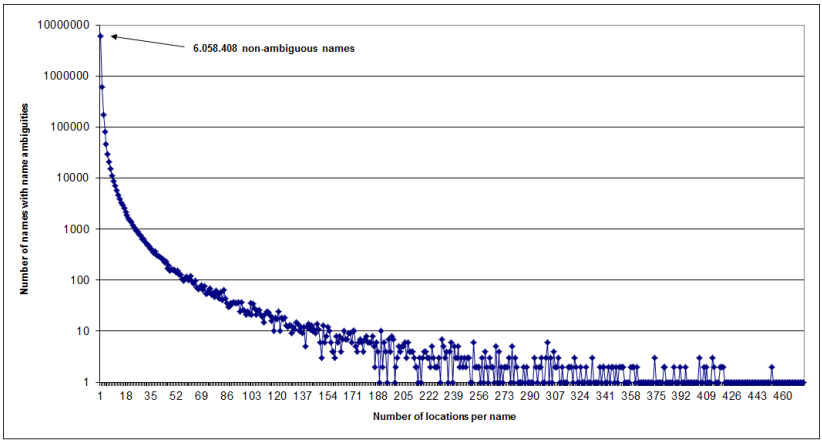


Abbildung 13.1.: Ambiguität innerhalb der Geonames-Ontologie

Textdaten

AllDocs	AllEntities	AvgEntitiesProDoc	AllDiffEntities
1002	12776	12,75	2217

Tabelle 13.3.: Korpus Statistik

Auch in dieser Domäne wurde der Textkorpus mithilfe von Webdokumenten zusammengestellt. Hierfür wurde das System des European Media Monitor (EMM)[13] verwendet. Das System wurde von Steinberger et al. [221] am Europäischen Joint Research Center entwickelt und bietet eine automatische Medienüberwachung. Es werden Webseiten in 43 Sprachen, die sich Nachrichten, Diskussionen *etc.* widmen, auf aktuelle Informationen überprüft und ins Archiv aufgenommen. Ursprüngliche Aufgabe des Systems war

[12] Eine von Richard Cyganiak erstellte Übersicht über die LOD Datensätze ist auf `http://richard.cyganiak.de/2007/10/lod/` [letzter Zugriff am 12.09.2011] gegeben.

[13] `http://emm.newsbrief.eu/` [letzter Zugriff am 12.09.2011]

es, die manuelle Medienrecherche an Europäischen Institutionen zu ersetzen. EMM bietet hierfür zwei wesentliche Anwendungen *1)* Newsbrief und *2)* Newsexplorer. Das erste System nimmt eine Sammlung der Daten vor, während das zweite umfangreiche Möglichkeiten bietet, um strukturiert auf diese zuzugreifen.

SumDiffEntityAmbi	AvgAmbiProEntity	AvgAmbiProDoc
36092	16,28	36,02

Tabelle 13.4.: Ambiguität innerhalb des EMM-Korpus

Der verwendete Textkorpus enthält Artikel aus den Kategorien *„Natural Desaster“* und *„Terrorist Attack“*. Beide Kategorien enthalten Artikel, die eine Vielzahl geographischer Entitäten beinhalten und somit geeignet sind den Zusammenhang mit der Geonames-Ontologie herzustellen. Die Daten wurden in Intervallen im Zeitraum zwischen 2008-2011 importiert. Insgesamt handelt es sich um 1002 Dokumente, von denen im Durchschnitt jedes nahezu 13 Entitätsbezeichner enthält (vgl. Tabelle 13.3). Durchschnittlich besitzt ein Entitätsbezeichner, der in diesen Texten enthalten ist, $16,28$ mögliche Bedeutungen. Die genaue Verteilung der Namensambiguität bezogen auf den verwendeten Korpus ist in Abbildung 13.2 dargestellt. Hervorzuheben ist, dass 1.504 von 2.217 Entitätsbezeichner, *d.h.* 67,84 %, innerhalb dieses Korpuses mehrdeutig sind. Die Mengenverhältnisse bezüglich der Zuweisung der im Text genannten Entitäten zu den Ontologiekonzepten sind in Anhang C angegeben.

13.3. Evaluationsergebnisse

In den folgenden Abschnitten sind die Evaluationsergebnisse des vorgestellten Verfahrens und dessen Varianten hinsichtlich der zuvor beschriebenen Fallstudien aufgeführt. Hierbei widmet sich Abschnitt 13.3.1 dem Vergleich der Verfahren zu den Ergebnissen des Ansatzes von Nguyen. Im Abschnitt 13.3.2 hingegen sind die Ergebnisse der Analyse der Geonames-Ontologie in Verbindung mit dem EMM-Datensatz aufgeführt. Dort liegt der Schwerpunkt im Vergleich der einzelnen Varianten, da der Umfang dieses Datensatzes eine Vielzahl

von Dokumenten umfasst, die es erlaubt diesbezüglich Schlussfolgerungen zu ziehen.

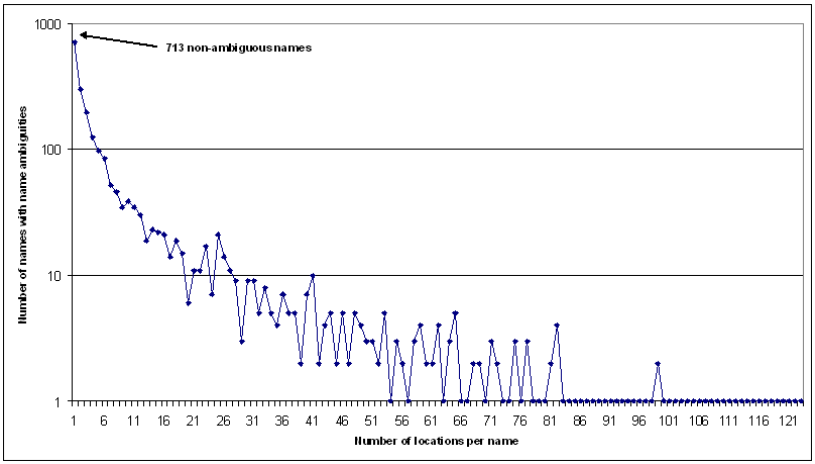


Abbildung 13.2.: Ambiguität innerhalb des Geonames Korpus

13.3.1. Fallstudie 1

Die Ergebnisse von Nguyen sind in Tabelle 13.5 angegeben.[14] Der von ihm entwickelte Ansatz basiert auf einem Vektorvergleichsverfahren, *d.h.* die Entitäten im Text sowie in der Ontologie werden jeweils als Vektor repräsentiert.[15] Im Basisansatz verwendet Nguyen alle im Text erwähnten Entitäten und Features, die innerhalb des Textes vorkommen sowie den ambiguen Bezeichner, der die Entität des Textes repräsentiert (globalIE). Diese bilden die Dimensionen des ersten Vektors. Dieser aus der textuellen Information erzeugte Vektor

[14] Diese geben die von ihm ermittelten Ergebnisse des Basisverfahrens für diesen Datensatz an, die im Artikel [163] vorgestellt werden. Die Resultate basierend auf der Verwendung von Wikipedia-Material werden nicht zu einem Vergleich herangezogen, da es sich hierbei um eine Erweiterung durch eine zweite Wissenbasis handelt.

[15] Eine Erklärung hinsichtlich der Verwendung des Vektormodells im Bereich Natural Language Processing ist im Buch von Manning und Schütze beschrieben [149]. Des weiteren wird das Verfahren bei der Analyse des Ansatzes in Kapitel 14 erörtert.

wird mit einem Vektor basierend auf der Ontologieinstanz der Entität verglichen. Der zuletzt genannte Vektor enthält als Dimensionen alle der Instanz direkt zugeordneten Properties. Die Ergebnisse sind im oberen Abschnitt der Tabelle 13.5 angegeben. Die BaseNP-Variante berücksichtigt die Noun Phrases, die innerhalb jeden Satzes, in dem ein ambiguer Bezeichner genannt wird, und innerhalb der Überschrift des untersuchten Dokuments vorkommen. Dies ändert teilweise die Dimensionswerte im Vektor des ambiguen Entitätsbezeichners. Nguyen gibt hierfür folgendes Textbeispiel an: *„The New York Philharmonic is looking at film music this week, with the composer John Williams on hand to lead works of his own and selections from Bernard Herrmann scores."*. In der globalIE-Variante werden folgende Begriffe als Benannte Entitäten verwendet: *„New York Philharmonic"*, *„John Williams"*, *„Philharmonic"* und *„Bernard Herrmann"*. BaseNP sind folgende vorhanden: *„week"*, *„film"*, *„music"*, *„works"*, *„composer John Williams"* und *„Bernard Herrmann scores"*. Insbesondere werden die direkten Zusammenhänge durch *„composer John Williams"* und *„Bernard Herrmann scores"* herausgestellt. Für weitere Details wird auf [163] verwiesen. Nähere Informationen zum Thema „Noun Phrases" werden ebenfalls in der Arbeit von Ramshaw und Mitchell [188] aufgeführt.

Der zu Grunde liegende Datensatz wurde von den programmatischen Umsetzungen der in den Kapiteln 8, 9, 10 und 11 vorgestellten Verfahren prozessiert. In Tabelle 13.5 sind die Ergebnisse für den Verfahrensvergleich beschrieben. Zunächst sind in den Zeilen eins und zwei die Ergebnisse des Ansatzes von Nguyen dargestellt. In Zeile drei werden die Ergebnisse, die durch die KIM-Plattform erzielt wurden beschrieben[17]. In Zeile 4 werden die besten Ergebnisse aus den ermittelten

[16] Die nachimplementierte Version des Nguyen-Algorithmus liefert von diesen Angaben abweichende Resultate. Die exakte Auflistung ist im Anhang D.1 gegeben. Das F-Measure Resultat ist für „John McCarthy" 6,249%, für „Georgia" 54,205% und für „Columbia" 73,115%.

[17] Die exakten Regeln, die diese Plattform zur Disambiguierung einsetzt, sind nicht frei zugänglich. Nach den Informationen in [179] ist zu schließen, dass jede Entität für sich analysiert wird. Es werden zunächst die möglichen Konzepten von Instanzen mit dem identifizierten Namen bestimmt. Anschließend wird versucht die möglichen Relationswerte mit Informationen innerhalb des Textes abzugleichen. Hierzu werden spezielle Muster verwendet, die innerhalb von Gate, welches zur Textprozessierung verwendet wird, hinterlegt sind. Hierbei ist anzunehmen, dass die Properties innerhalb des Textes (die aufgrund der Muster bestimmt werden) nicht eindeutig identifiziert werden können und somit die Bestimmung der zugehörigen Werte nicht möglich ist.

Zeile	Ansatz	Algorithmusvariante	John	Georgia	Columbia	Durchschnitt
1	Nguyen:	globalIE	82,14%	15,27%	92,50%	50,71%
2	Nguyen:	zuzüglich BaseNP	82,14%	16,67%	92,50%	51,42%
3	Kim:		28,89%	54,07%	67,88%	50,28%
4	Kleb:	Beste Ergebnisse	87,821%	86,854%	59,459%	78.044%
5	Kleb:	1011 bzw. 1010	87,821%	77,114%	59,459%	74.798%

Tabelle 13.5.: Ergebnisse der verschiedenen Ansätze: Nguyen [163][16], KIM und der in dieser Arbeit vorgestellten Methode.

Resultaten aller vom Autor dieser Arbeit vorgestellten Algorithmusvarianten aufgezeigt.[18] In Zeile 5 hingegen wird das Resultat der am besten evaluierten Algorithmusvarianten vorgestellt.[19] Hierbei stammen die Ergebnisse für „John McCarthy“ und „Columbia„ in Zeile vier ebenfalls von diesen beiden Algorithmusvarianten. Der Vergleich der Ergebnisse, die durch alle Varianten des Algorithmus erzielt wurden, sind vollständig im Anhang D.2 dieser Arbeit enthalten.

Werden die aufgeführten Resultate beider Ansätze verglichen, so stellt sich heraus, dass der Ansatz von Nguyen ein besseres Ergebnis für den Testfall „Columbia“ erzielt. Dem hingegen erreicht das im Rahmen dieser Arbeit entwickelte Verfahren bessere Resultate für die Testfälle „John McCarthy“ und „Georgia“. Das in dieser Arbeit vorgestellte Verfahren erreicht ebenfalls ein besseres Durchschnittsergebnis.

Aufgrund der Ähnlichkeit der Ergebnisse beider Verfahren für „John McCarthy“ erfolgt eine Fokussierung auf „Georgia“ und „Columbia“, um den Unterschied zwischen beiden Ansätzen aufzuzeigen. Im Fall von „Georgia“ ist der direkte semantische Zusammenhang zwischen den in den Texten erwähnten Entitäten nicht gegeben, *d.h.* die Auswertung der direkt zugeordneten Property-Information stellt sich als ungenügend heraus. Das im Rahmen dieser Arbeit entwickelte Vorgehensmodell überwindet das, indem Verbindungen über mehrere Instanzen hinweg zur Auflösung der Ambiguität berücksichtigt werden. Im Fall von „Columbia“ hingegen birgt die Restriktion auf die direkt in Beziehung stehenden semantischen Informationen Vorteile. Die Beschränkung auf den direkten Entitätsvergleich, *d.h.* dass kein Zusammenhang zwischen allen Entitäten erreicht werden muss, erzielt hier bessere Ergebnisse.[20] Im Verfahren des Autors dieser Arbeit hingegen resultiert die Verwendung von Relationen über mehrere Entitäten hinweg in den meisten Fällen in der Bevorzugung der Entität, die den „District of Columbia“ beschreibt. Dies geschieht aufgrund der Verbindungen innerhalb der Ontologie. Somit wird diese in den meisten Fällen bei der Disambiguierung als Referenz bevorzugt. Dies führt

[18] Eine Beschreibung der in der Tabelle verwendeten Zahlenkodierung findet sich in Anhang D.2.

[19] Für das Ergebnis trat kein Unterschied dabei auf, ob die Exploration nur über Ontologieelemente fortgeführt wurde, die nur eine ausgehende Kante besitzen oder über die Ontologieelemente mit der geringsten Anzahl ausggehender Kanten.

[20] Der Zusammenhang wird implizit betrachtet, falls die Bezeichner der weiteren Entitäten im Vektor enthalten sind.

somit zu falschen Ergebnissen. Dieses Problem kann mit hoher Wahrscheinlichkeit in einer zukünftigen Implementierung durch eine höhere initiale Gewichtung derjenigen Entitäten, deren Data-Properties im Text gefunden werden, behoben werden.

Im Anhang D.1 werden die Ergebnisse vorgestellt, die durch eine Nachimplementierung des von Nguyen entwickelten Ansatzes erzielt werden konnten. Diese werden dort näher analysiert und mit dem vom Autor dieser Arbeit entwickelten Ansatz verglichen. Beim Vergleich der Entitätsbezeichner zeigt sich, dass bei einer Berücksichtigung des vorgegebenen ambiguen Bezeichner, *z.B.* „Columbia“ *etc.*, gegenüber den im Text exakt genannten Bezeichnern, *z.B.* „District of Columbia“ eine Verschlechterung der Resultate des von Nguyen vorgeschlagenen Ansatzes eintritt. Nähere Informationen hierzu sind im Anhang gegeben.

Im Rahmen des Vergleichs der verschiedenen Algorithmusvarianten, die in dieser Arbeit vorgestellt werden, stellte sich im Rahmen dieser Evaluation der bidirektionale Ansatz unter Verwendung von Bestärkendem Lernen und dem semantischen Kantenmaß als Variante mit den besten Ergebnissen heraus. Im Vergleich der verschiedenen Testvarianten zueinander zeigt sich, dass für die Analyse die Verwendung des semantischen Kantenmaßes durchschnittlich zu besseren Ergebnissen führt, als die Verwendung des heuristischen Maßes oder dem Verzicht auf die Gewichtung der Kanten. Eine Verwendung des Lernverfahrens in Kombination mit der Berücksichtigung eines Kantenmaßes führt in den meisten Fällen zu einer Verbesserung. Hierbei stellt sich heraus, dass zum einen die Entitätskonstellationen in mehreren Texten Ähnlichkeiten aufweisen, *d.h.* Teile der in Texten aufgeführten Entitäten in anderen Texten ebenfalls vorhanden sind. Zum anderen ist das korrekte Referenzieren von Entitätsbezeichnern in zuvor analysierten Dokumenten entscheidend für die Qualität, die durch Bestärkendes Lernen erreicht werden kann. Dies bedeutet, dass die umfangreiche Bestimmung falscher Referenzen in vorherigen Dokumenten negativ auf die weitere Referenzbestimmung auswirkt. Letzteres ist bei der Verwendung der vorgestellten Kantenheuristik sowie dem Verzicht auf eine Kantengewichtung gegeben. Die Fokussierung auf Aktivierungswerte anstatt auf Distanzwerte hat bei der Evaluation keinen Unterschied zur Folge. Dies resultiert bei diesem Testfall aus der gegebenen Ontologiestruktur. Hierbei stimmen die

Resultatgraphen aufgrund ihres geringen Umfangs für beide Varianten in allen Fällen überein.

13.3.2. Fallstudie 2

Im Folgenden werden die Resultate der Evaluation des Geonames-Datensatzes analysiert. In diesem Abschnitt erfolgt eine Konzentration auf die Ergebnisse der Ansätze, die mittels Fokussierung auf die Aktivitätswerte (siehe auch Algorithmus 7) vorgenommen wurden. Die Resultate, die mittels einer Fokussierung auf die Distanz (siehe auch Algorithmus 6) berechnet wurden, werden im Anhang E erörtert.

Die durch die verschiedenen Verfahrensvarianten erzielten Ergebnisse sind in Tabelle 13.6 bzw. 13.7 aufgelistet. Die besten Resultate wurden von der Algorithmusvariante erzielt, die auf der bidirektionalen Explorationsmethode mit heuristischer Kantengewichtung und lokaler Kohärenz basiert. Die Tabelle ist wie folgt strukturiert:

- Explorationsmethode: 0 - Unidirektional (siehe Kapitel 8); 1 - Bidirektional (siehe Kapitel 9)
- Bestärkendes Lernen: 0 - Nein; 1 - Ja
  Verwendung des Bestärkenden Lernens (siehe Kapitel 11)
- : 0 - Nein; 1 - Ja
  Verwendung des Ansatzes der lokalen Kohärenz (siehe Kapitel 10)

Bei einer näheren Auseinandersetzung mit den erzielten Werten fällt zunächst auf, dass in allen Evaluationsergebnissen der Recall einen höheren Wert als die Precision besitzt. Ohne die Verwendung eines Kantenmaßes liegen die Werte am weitesten voneinander entfernt. Des Weiteren erfolgt eine Steigerung der Ergebnisse durch die verschiedenen Verfahrensvarianten. Durch die bidirektionale Explorationsmethodik (siehe Kapitel 9) wird ein Maximum an Informationen in jeder gegebenen Explorationssituation ausgetauscht. Die dadurch erzielte Änderung der Prozessierungsfolge der zu berücksichtigenden Knoten ermöglicht eine primär lokal fokussierte Exploration, die sich im Rahmen der verwendeten geographischen Ontologie als

Explorationsmethodik	Bestärkendes Lernen	Kontextgröße 10	Recall	Precision	F-Measure
Fokussiert auf Aktivierungswerte - ohne Ranking					
0	0	0	79,779	58,644	67,598
1	0	0	79,403	59,235	67,852
0	0	1	80,045	58,749	67,763
1	0	1	79,779	59,750	68,327
0	1	0	77,522	62,770	69,370
1	1	0	77,339	64,496	70,336
0	1	1	78,392	59,913	67,918
1	1	1	77,643	61,172	68,430
Fokussiert auf Aktivierungswerte - semantisches Maß					
0	0	0	70,921	67,677	69,261
1	0	0	70,693	68,168	69,408
0	0	1	71,369	67,858	69,569
1	0	1	71,685	68,905	70,268
0	1	0	71,377	69,178	70,260
1	1	0	71,980	70,523	71,244
0	1	1	71,326	68,579	69,926
1	1	1	71,977	69,990	70,970
Fokussiert auf Aktivierungswerte - Heuristisches Maß					
0	0	0	74,226	67,860	70,900
1	0	0	73,709	69,679	71,637
0	0	1	74,675	68,279	71,331
1	0	1	74,132	69,995	72,004
0	1	0	73,373	68,879	71,055
1	1	0	73,024	70,836	71,913
0	1	1	73,343	68,140	70,646
1	1	1	73,133	70,030	71,548

Tabelle 13.6.: Resultate Geonames Datensatz (Fokussiert auf Aktivierungswerte)

Explorationsmethode	Induktives Lernen	Kontextgröße 10	Resultatgröße	Korrekte Resultate	Recall	Precision	Fmeasure
Activation Focused - ohne Ranking							
0	0	0	14,113	5,631	79,779	58,644	64,595
1	0	0	13,666	5,598	79,403	59,235	64,937
0	0	1	14,284	5,640	80,045	58,749	64,773
1	0	1	13,526	5,628	79,779	59,750	65,446
0	1	0	11,553	5,524	77,522	62,770	67,521
1	1	0	10,721	5,459	77,339	64,496	68,633
0	1	1	12,815	5,567	78,392	59,913	65,411
1	1	1	12,099	5,510	77,643	61,172	66,097
Activation Focused - semantisches Maß							
0	0	0	7,925	5,072	70,921	67,677	68,877
1	0	0	7,673	5,058	70,693	68,168	69,090
0	0	1	7,975	5,112	71,369	67,858	69,162
1	0	1	7,732	5,133	71,685	68,905	69,928
0	1	0	7,698	5,096	71,377	69,178	69,979
1	1	0	7,510	5,144	71,980	70,523	71,053
0	1	1	7,749	5,084	71,326	68,579	69,576
1	1	1	7,602	5,164	71,977	69,990	70,705
Activation Focused - Heuristisches Maß							
0	0	0	8,488	5,242	74,226	67,860	70,026
1	0	0	7,861	5,234	73,709	69,679	71,163
0	0	1	8,482	5,274	74,675	68,279	70,440
1	0	1	7,897	5,266	74,132	69,995	71,504
0	1	0	8,116	5,210	73,373	68,879	70,507
1	1	0	7,611	5,178	73,024	70,836	71,638
0	1	1	8,212	5,199	73,343	68,140	69,889
1	1	1	7,732	5,196	73,133	70,030	71,128

Tabelle 13.7.: Resultate Geonames Datensatz (Fokussiert auf Aktivierungswerte

vorteilhaft erweist. Insofern wird eine Steigerung der Evaluationsergebnissen gegenüber dem unidirektionalen Ansatz unabhängig vom verwendeten Kantenmaß durchweg erzielt. Betrachtet man den diesbezüglich Recall so wird deutlich, dass dieser nur geringfügig von der unidirektionalen Explorationsweise abweicht. Die Precision hingegen erfährt in allen Fällen eine Verbesserung.

Die Resultate des unidirektionalen sowie bidirektionalen Ansatzes erfahren eine weitere Steigerung durch die Berücksichtigung einer lokalen Kohärenz, die durch die Kontextgröße 10 vorgegeben wurde. Dies bedeutet, dass jeweils die Umgebung von fünf Wörtern vor und fünf Wörtern nach der Entität als Kontext berücksichtigt wurden (siehe Kapitel 10). Die Analyse der dadurch erzielten lokalen Segmente ermöglicht eine Konzentration auf direkt miteinander in Beziehung stehende Entitäten und dadurch eine bereits gewinnbringende Disambiguierung im ersten Prozessschritt. Diese kann erfolgreich in den nachfolgenden Schritten eingebunden werden. Hierbei ist zu berücksichtigen, dass die bidirektionale Exploration gegenüber der unidirektionalen bessere Resultate erzielt. Ebenfalls zeigt sich eine Steigerung der Ergebnisgüte unabhängig vom verwendeten Kantenmaß.

Eine weitere Verbesserung der Resultatwerte kann durch die Anwendung bestärkenden Lernens (siehe Kapitel 11) erzielt werden. Bestärkendes Lernen ermöglicht die Berücksichtigung zuvor erzielter Ergebnisse der Analyse einzelner Entitäten bei deren erneuten Nennung in Texten, die während einer späteren Analyse untersucht werden. Die durch das bestärkende Lernen ermöglichte Steigerung geht insbesondere auf die hohe Anzahl an Texten zurück, die mit der Nennung gleicher Entitätsbezeichner in mehreren Texten einhergeht. Kernmerkmal der vorgestellten Variante des Bestärkenden Lernens ist eine Steigerung der Relevanz einer Entität in einem aktuell untersuchten Text, falls diese in diesem Text mit weiteren, bereits in einem zuvor analysierten Text enthaltenen Entitäten, auftaucht. Das ist verbunden mit der Häufigkeit der zuvor analysierten Texte mit dieser Eigenschaft. Die begründet zugleich die Abnahme der Ergebnisqualität, falls Bestärkendes Lernen mit dem Ansatz der lokalen Kohärenz kombiniert wird. Die Steigerung der Relevanz findet nur im ersten Schritt statt. Da bei der lokalen Kohärenz nur eine geringe Anzahl von Entitäten im Textausschnitt enthalten ist, tritt der Effekt der gegenseitigen Steigerung der Relevanz durch gemeinsames Auftauchen von mehreren Entitäten im untersuchten Text kaum auf. Frei von diesem Phänomen

tritt eine Steigerung unabhängig vom Kantenmaß und von bidirektionaler gegenüber unidirektionaler Exploration auf. Die Verwendung von bestärkendem Lernen bewirkte ebenfalls bei nahezu ähnlichem Recall eine Steigerung der erzielten Precision.

	Geo (ohne NounChunk / ohne Stemming)	Geo (mit NounChunk / ohne Stemming)	Geo (mit NounChunk / mit Stemming)
Precision	61,06	58,89	60,59
Recall	53,05	51,17	42,26
F-Measure	56,77,82	54,76	49,79

Tabelle 13.8.: Resultat der Nachimplementierung des Nguyen-Ansatzes

Bei einer Analyse der verwendeten Kantenmaße stellt sich heraus, dass die Verwendung eines Maßes eine Verbesserung der Resultatwerte zur Folge hat. Die besseren Werte des heuristischen Maßes gegenüber dem semantischen Maß lassen sich auf die Ontologiestruktur zurückführen. Bei der Geonames-Ontologie handelt es sich um eine Wissensbasis, die geographische Information beinhaltet. Generell kann davon ausgegangen werden, dass je größer die Anzahl der aus- und eingehenden Verbindungen, desto „größer“[21] und bekannter ist die geographische Entität. Es kann davon ausgegangen werden, dass Entitätsbezeichner sich in den meisten Texten auf die bekanntesten Instanzen beziehen, so dass sich diese Effekte gegenseitig begünstigen.

Ergebnisse der Nguyen-Nachimplementierung Um einen Vergleich zu ermöglichen wurde der Geonames-Datensatz ebenfalls von einer vorgenommenen Nachimplementierung des Nguyen-Ansatzes analysiert. Die erzielten Ergebnisse werden in Tabelle 13.8 wiedergegeben.

[21] „Größer“ bezieht sich hierbei auf die Einteilung in Dorf, Stadt, Großstadt *etc.*

Die geringeren Ergebniswerte gegenüber den zuvor vorgestellten Resultaten der innerhalb dieser Arbeit entwickelten Verfahren sind darauf zurückzuführen, dass die vektorbasierte Implementierung nur den direkten Kontext einer Entität berücksichtigt. Somit können längere Wege, *d.h.* Zusammenhänge, die sich über mehrere Entitäten hinweg ergeben, durch dieses Verfahren nicht gewinnbringend untersucht werden. Ohne „NounChunk“ erfolgt eine Berücksichtigung der konkreten Terme ohne Zusammenfassung von komplexeren Nomensausdrücken. Dies zusammen mit der Berücksichtigung der gegebenen Wortform, *z.B.* ohne Stemming, erzielt die besten Ergebnisse, da auf die exakten Terme zurückgegriffen werden kann. Die Berücksichtigung beider Modifikationen erhöht die Ambiguität und führt zu schlechteren Ergebnissen.

13.4. Schlussfolgerungen

Die vorgenommenen Evaluationen ermöglichen verschiedene Schlussfolgerungen, die zum einen eine Einschätzung der in dieser Arbeit vorgestellten Verfahrensweisen zulassen und zum anderen auf mögliche Verbesserungen hinweisen, die in einer künftigen Weiterentwicklung vorgenommen werden können.

Durch die Analyse des KIM-Datensatzes kann nachgewiesen werden, dass die Analyse anhand eines Ontologiegraphen Vorteile bei der Berücksichtigung des Zusammenhangs zwischen den Entitäten bietet. Hierbei ist hervorzuheben, dass die Länge der durch die Object-Properties gegebenen Pfade nicht beschränkt ist. Jedoch ist darauf hinzuweisen, dass die Berücksichtigung von Data-Properties zusätzliche Informationsquellen birgt, die momentan nicht in dem vorgestellten Verfahren dieser Arbeit berücksichtigt wird (siehe auch Kapitel 15.2). Außerdem ergeben sich durch die verwendeten Entitätsbezeichner Unterschiede in den Resultaten (siehe hierzu Anhang D).

Bei der Analyse des Geonames-Datensatzes zeigt sich, dass in nahezu allen Fällen über $\frac{2}{3}$ der Entitätsbezeichner den richtigen Entitäten innerhalb der Ontologie zugeordnet und durchschnittlich 75% von ihnen aufgefunden werden können. Es wird auch nachgewiesen, dass die verschiedenen Modifikationen des Basisalgorithmus, *d.h.* bidirektionale Exploration, lokale Kohärenz und die der Einsatz von

Bestärkendem Lernen, in einer Steigerung der Ergebnisgüte resultieren. Zudem wird der Nachweis erbracht, dass die Verwendung eines Kantenmaßes eine genauere Beurteilung der ontologischen Zusammenhänge ermöglicht und dadurch eine Verbesserung der vorgenommenen Disambiguierung erreicht werden kann.

14. Thematisch verwandte Arbeiten

In diesem Kapitel werden Arbeiten mit ähnlicher Zielsetzung bzw. mit verwandter algorithmischer Methodik vorgestellt. Es handelt sich hierbei um Ansätze zur Disambiguierung ambiguer Begriffe, *d.h.* Entitäten, allg. Wörter *etc.*, Arbeiten zur Suche bzw. zur Erkennung von Entitäten in Texten/Ontologien sowie zur Definition von Semantischen Maßen. Ebenfalls betrachtet werden sowohl die theoretischen Hintergründe zur Disambiguierung von Entitäten, als auch der Methodik von Spreading Activation.

Zunächst werden in Abschnitt 14.1 Arbeiten von Autoren vorgestellt, die sich der gleichen Problemstellung, *d.h.* Disambiguierung mittels einer gegebenen Ontologie, widmen.[1] Weitere verwandte Arbeiten, die - ähnlich dem in dieser Arbeit vorgestellten Verfahren - Algorithmen basierend auf einem Graphmodell verwenden, werden in Abschnitt 14.2 dargestellt. Die Abschnitte sind jeweils zweigeteilt. Zunächst werden Arbeiten, die algorithmisch eine Ähnlichkeit zum vorgestellten Ansatz des Autor dieser Arbeit aufweisen, explizit diesem gegenübergestellt und nachfolgend erfolgt eine etwas knappere Darstellung weiterer thematisch relevanter Arbeiten.

14.1. Arbeiten mit vergleichbarer Problemstellung

Die im Folgenden vorgestellten Arbeiten zeichnen sich durch ein direkt vergleichbares Vorgehensmodell aus. Dies impliziert, neben

[1] Die Verwendung einer Ontologie steht hierbei im Vordergrund, *d.h.* die Ansätze müssen nicht zwangsläufig auf einem Graphmodell basieren.

dem vorausgesetzten Problem ambiguer Entitäten und der sich daraus ergebenden Problemstellung der Disambiguierung, die Verwendung einer **Ontologie** als Wissensmodell sowie die optionale Verwendung der **Spreading-Activation** Technik zur Referenzbestimmung. Zunächst erfolgt in Abschnitt 14.1.1 die Vorstellung der Arbeiten von Nguyen und Cao, deren Artikel [163] in Kapitel 13 als Vergleichsarbeit verwendet wird. Darauffolgend werden in Abschnitt 14.1.2 die Arbeiten weiterer Autoren vorgestellt.

14.1.1. Arbeiten von Nguyen

Im Folgenden werden die Arbeiten von Nguyen und Cao vorgestellt. Diese Autoren zeichnen sich dadurch aus, dass der Schwerpunkt ihrer Forschung sich dem Problem der Entitätsdisambiguierung widmet. Zur Lösung dieses Problems greifen die Autoren in ihrer Arbeit zum einen auf eine bereits existierende Hintergrundontologie (KIM) zurück. Zum anderen erstellen sie eine eigene Wissensbasis bzw. erweitern eine existierende Wissensbasis anhand von Wikipedia-Daten.

Nguyen und Cao: ***Artikel zur Disambiguierung von Entitäten [161, 163, 162, 164]***

Die Arbeiten von Nguyen et al. sind durchweg fokussiert auf den Forschungsbereich der Entitätsdisambiguierung. Hervorzuheben ist die Tatsache, dass sich alle Arbeiten auf die Disambiguierung textueller Entitätsbezeichner anhand von gegebenen Entitäten in bereits vorhandenen oder selbsterstellten Wissensbasen beziehen. Im zuerst erschienenen Artikel [161] beschäftigen sich die Autoren mit der Disambiguierung anhand von direkten[2] Object-Properties zwischen den Instanzen. Die mehrdeutigen Entitätsbezeichner im Text werden als jeweiliges Zentrum eines umgebenden Textfensters verwendet. Die Disambiguierung eines ambiguen Bezeichners findet anhand weiterer, ebenfalls in diesem Fenster enthaltenen Entitätsbezeichner statt, *d.h.* es wird überprüft, ob eine für den Entitätsbezeichner in Frage kommende Entität zu einer oder zu mehreren in diesem Textfenster enthaltenen Entitäten, die durch die weiteren Entitätsbezeichner beschrieben werden, in Beziehung stehen.

[2] Entspricht einer Pfadlänge mit Abstand 1.

Hierzu werden nur monoseme Entitäten im Kontextfenster verwendet. Für eine Entität werden alle Koreferenz-Informationen[3] berücksichtigt, *d.h.* alle in den verschiedenen Textfenstern der Koreferenzen enthaltenen eindeutigen (*d.h.* nicht ambiguen) Entitäten werden zur Disambiguierung des Bezeichners verwendet. Zudem führen die Autoren eine Klassenheuristik ein, die eine höhere Wahrscheinlichkeit von Instanzen eines bestimmten Konzepts gegenüber den Instanzen anderer Konzepte und somit die Bevorzugung einer bestimmten Konzeptzugehörigkeit ermöglicht. Beispielsweise ermöglicht dies die Bevorzugung von Personen, *d.h.* Instanzen des Konzepts „Person", gegenüber Organisationen, *d.h.* Instanzen des Konzepts „Organisation".[4] Der Ansatz ist auf die Entitäten innerhalb der KIM-Ontologie (siehe auch Abschnitt 13.2) ausgerichtet.

Im darauffolgenden Artikel [163], der ebenfalls die Grundlage der Vergleichsevaluation in Kapitel 13 bildet, wird dieser Ansatz durch ein Vektormodell konkretisiert. Hierzu wird sowohl ein Vektor für den im Text enthaltenen Entitätsbezeichner als auch für jede in der Ontologie beschriebene Entität erstellt. Für Letztere wird die Ontologieinformation zu einer Entität, *d.h.* deren Klassenhierarchie, Object- und Data-Properties *etc.*, zunächst zu einem String zusammengesetzt und anschließend in einen Termvektor umgewandelt. Die Autoren versuchen eine Erweiterung dieses zunächst rein auf der Ontologie basierenden Vektors durch Informationen, die sie in Wikipedia vorfinden. Hierzu erstellen sie ein auf Wikipedia basierendes Entitätslexikon, *d.h.* je Entität in Wikipedia wird ebenfalls ein Termvektor erzeugt, welcher die Informationen des Titels, der Titel der Weiterleitungsseiten, Kategorie(n) *etc.* enthält. Zur Erweiterung der Information zu einer Entität der Ontologie erfolgt ein Vergleich zwischen den erstellten Entitätsvektoren und den Vektoren der Entitäten aus Wikipedia. Dieser Vektorvergleich erfolgt anhand des Maßes Termfrequency - Inverted Document Frequency (tf-idf) (siehe auch [48]). Das Ergebnis ist Grundlage zur

[3] Eine Koreferenz beschreibt eine erneute Nennung der Entität gegebenenfalls auch unter alternativem Namen, *z.B.* zunächst „[...] die USA sind eine reiche Industrienation [...]" und später „US Präsident Truman [...]". Hier beziehen sich „USA" und „US" auf dieselbe Entität.

[4] Dies bedarf der Voraussetzung, dass sowohl das Konzept „Person" als auch das Konzept „Organisation" existieren.

Vereinigung von Vektorinformation, *d.h.* Erweiterung der ontologischen Entitätsinformation. Zur Disambiguierung textueller Entitätsbezeichner wird zunächst für jeden im Text erwähnten Entitätsbezeichner ebenfalls ein Vektor erstellt. Dieser basiert auf den im Text verwendeten Nominalphrasen in der Überschrift und im Textfenster, in dem der Entitätsbezeichner den Mittelpunkt bildet (vgl. [161]). Die eigentliche Disambiguierung erfolgt anhand eines Vergleichs jedes dieser Vektoren mit den erweiterten Vektoren der Ontologieentitäten. Die aus dem Vergleich hervorgegangene, am besten bewertete Entität in der Ontologie wird als Referenz für den Bezeichner verwendet. Dieser Ansatz baut auf dem vorherigen auf, da er ebenfalls den textuellen Bezeichner einer Instanz der Ontologie zuordnet. Die Verwendung der Wikipedia-Informationen kann als Erweiterung eines Basisverfahrens bezeichnet werden. Dieses Basisverfahren beschreibt eine Referenzbestimmung anhand eines Vektorvergleichs zwischen einem Vektor des Entitätsbezeichners im Text und den Vektoren der Ontologieentitäten. Dieses Verfahren wird daher ohne die Erweiterung durch Informationen aus Wikipedia als Referenzverfahren für die Evaluation in Kapitel 13 gewählt.

Im anschließend erschienenen Artikel [162] greifen die Autoren auf keine explizite Ontologie, sondern ausschließlich auf Informationen aus Wikipediadaten zurück.[5] Die Disambiguierung basiert bei diesem Ansatz auf den Entitätsvektoren aus Wikipedia, die dem in Artikel [163] beschriebenen Vektoraufbau entsprechen. Der Algorithmus zur Disambiguierung folgt einem iterativen Modell. Zunächst werden alle eindeutigen (*d.h.* nicht ambiguen) Entitäten bestimmt und deren bisherige Bezeichner im Text durch die Titel der diese Entitäten beschreibenden Wikipediaseiten ersetzt. Diese beinhalten oft zusätzliche Informationen, *z.B.* der Staat für Lokationen (Miami, Florida). Die Änderung im Text bewirkt implizit eine Veränderung der Vektoren der Entitätsbezeichner im Text. Anschließend erfolgt für jeden ambiguen Entitätsbezeichner die Bestimmung aller Entitätsbezeichner, die zu diesem einen Abstand von maximal 10 Wörtern aufweisen. Diese werden überprüft, ob sie sich in der Wikipedia-Beschreibung einer in Frage kommenden Entität wiederfinden.

[5] Die Vektoren werden aufgrund einer Interpretation aus Wikipedia-Artikeln erstellt. Diese basieren daher auf keiner gegebenen Ontologie, wie *z.B.* DBpedia.

Je nach Übereinstimmung ist eine Disambiguierung möglich. Falls diese durchgeführt werden konnte, werden die entsprechenden Entitätsbezeichner im Text ebenfalls ersetzt. Dieses Vorgehen wird iterativ durchgeführt. Zuletzt erfolgt die tf-idf basierte Bestimmung der Referenzen für die Entitätsbezeichner, die bis zu diesem Zeitpunkt nicht zugeordnet werden konnten.

Im zuletzt erschienenen Artikel [164] wird der Ansatz von [163] erneut aufgefriffen. Das Entitätslexikon basiert, wie im Ansatz [162], ausschließlich auf Wikipedia-Daten. Insofern wird die dort beschriebene Termvektorerstellung für die Entitätsbezeichner im Text und die Entitäten des Thesaurus verwendet. Im Unterschied zu den vorherigen Artikeln wird eine sehr detaillierte Evaluation durchgeführt, *d.h.* es werden verschiedene Kombinationen der zu berücksichtigten Charakteristika für die Erstellung des Vektors des Entitätsbezeichners im Text und der Entitäten im Entitätslexikon untersucht. Diese werden in einer Analyse zueinander abgewogen und evaluiert.

Verfahrensvergleich
Die vorgestellten Ansätze basieren sowohl auf der Repräsentation von Ontologieinstanzen als auch von Entitäten innerhalb des Textes durch Termvektoren. Im Bereich des Informationen Retrieval ist der Vergleich durch Vektordarstellung weit verbreitet und demzufolge basieren viele Verfahren für Textclusterung, Suche, Klassifizierung *etc.* auf der Auswertung von Termvektoren.

Für das Verfahren von Nguyen und Cao sind hierbei zwei Faktoren wesentlich. Zum einen die verwendeten Dimensionen, *d.h.* die Auswahl der Informationen basierend auf der Ontologie bzw. dem Text. Zum anderen das verwendete Maß für den Vektorvergleich. Für Letzteres greifen die Autoren auf das Maß tf-idf zurück. Dieses setzt jedoch einen vorhandenen Korpus mit mehreren Dokumenten voraus, damit der Parameter der Inverted Document Frequency Aussagekraft erhält. Hervorzuheben ist insbesondere die Einschränkung auf einen reinen Termvergleich, *d.h.* jede Art von Wissen, das berücksichtigt werden soll, muss als Term kodiert werden, um im Vektormodell dargestellt werden zu können. Das tf-idf Maß selbst beschreibt nur die Häufigkeit der Nennungen eines Terms und unterscheidet

sich somit grundlegend von der Verwendung semantischer Maße. Die vorgeschlagenen Vorgehensweisen zur Disambiguierung berücksichtigten insofern nur die direkten ontologischen Beziehungen zwischen Entitäten. Die Beschreibung des ontologischen Zusammenhangs ist somit stark eingeschränkt. Im Gegensatz zu dem vom Autor dieser Arbeit vorgestellten Verfahren ergibt sich zudem kein Bild des Gesamtzusammenhangs zwischen den enthaltenen Entitäten.[6] Dies ist darin begründet, dass jeweils nur ein kleiner Ausschnitt der im Text enthaltenen Entitäten für den Vergleich einer Entität herangezogen wird, *d.h.* nur die Entitäten, die im Textfenster der untersuchten Entität vorkommen. Weiterhin ist das Verfahren äußerst abhängig von den Informationen in Wikipedia und ist in späteren Artikeln ausschließlich darauf beschränkt. Problematisch ist hierbei, dass die Heuristiken zur Erstellung eines Entitätsvektors aus Wikipedia, *d.h.* Titel, Links *etc.*, nicht direkt auf weitere Ontologien übertragen werden können und somit der Vergleich in Kapitel 13 auf die in Artikel [163] vorgestellte Methode zur Erstellung von Entitätsvektoren mittels Ontologien beschränkt ist.

14.1.2. Übersicht weiterer verwandter Arbeiten

Das Problem der Entitätsdisambiguierung wird mit Interesse verfolgt und viele Autoren widmen sich diesem, indem sie für diese Problemstellung Ansätze zur Lösung entwerfen. Im Folgenden werden diese Verfahren jeweils kurz beschrieben und dem Ansatz des Autors dieser Arbeit vergleichend gegenübergestellt. Hervorzuheben ist die Tatsache, dass diese Ansätze ebenfalls Ontologie(n) als zugrundeliegende Wissensbasis für die aufzufindenden Referenzen verwenden.

Hassell, Aleman-Meza, Arpinar: **Ontology-Driven Automatic Entity Disambiguation in Unstructured Text** [106]

Hassell et al. stellen ein Verfahren zur Disambiguierung von Autorennamen anhand der DBLP-Ontologie[7] vor. Das Verfahren ist auf diese Ontologie abgestimmt und beschreibt den Vorgang Entitätsbezeichner in Texten, *d.h.* die Namen von

6 Dieses ergibt sich durch den Steinerbaum.

7 `http://sw.deri.org/~aharth/2004/07/dblp` [letzter Zugriff am 12.09.2011]

Autoren der DBLP-Ontologie, den im Kontext korrekten Instanzen dieser Ontologie zuzuordnen. Das Verfahren basiert zum einen auf Informationen textueller Art, *z.B.* der Erkennung von Autorennamen, Institutionsbezeichnern, Forschungsgebieten und den diesbezüglich erstellten textuellen Maßen, *z.B.* textuelle Distanz, *d.h.* die Anzahl der Terme, zwischen diesen Bezeichnern. Zum anderen werden Informationen innerhalb der Ontologie verwendet, *z.B.* der Grad der Ambiguität eines Namens (*d.h.* wieviele Instanzen werden durch diesen Namen bezeichnet) und die semantischen Beziehungen zwischen diesen Instanzen. Anhand dieser Informationen werden die in Frage kommenden Instanzen gewichtet und anhand dessen die Wahrscheinlichkeit bestimmt, zu der sie als Referenz für die Entität im Text in Betracht kommen. Die Gewichtung basiert auf den textuellen Maßen sowie den semantischen Beziehungen zu den Co-Autoren. Für diese Autoren wird ein iteratives Verfahren zur Anpassung der Wahrscheinlichkeit verwendet. Dieses basiert auf der Auswertung semantischer Relationen, indem Instanzen hoher Wahrscheinlichkeit die mit ihnen verbundenen Instanzen höher gewichten und diese wiederum diese Gewichtung an die mit ihnen über die Co-Autor-Object Property verbundenen Instanzen weitergeben.

Verfahrensvergleich
Das Verfahren ist auf die DBLP-Ontologie abgestimmt. Indikator hierfür ist der Umstand, dass Institutionen nicht als Entitäten in der Ontologie dargestellt werden, sondern als Data-Properties. Je nach Data-Property, *d.h.* Instutionzugehörigkeit oder Interessengebiet, wird ein unterschiedliches Textmaß verwendet. In der Ontologie existiert nur eine Objekt-Property. Diese bezeichnet die Autor-zu-Co-Autor Relation. Das führt dazu, dass im Gegensatz zum Verfahren des Autor dieser Arbeit, hier nur in Ausnahmefällen ein Gesamtzusammenhang des Textes, *d.h.* zwischen allen genannten Autoren, dargestellt werden kann. Dieser ist auf den Fall beschränkt, dass die erwähnten Personen über Co-Autor Beziehungen miteinander verbunden sind.

Garcia, Blazquez del Toro, Sanchez: **IdentityRank: Named Entity Disambiguation in the Context of the NEWS Project** [88]

Der Ansatz von Garcia et al. ermöglicht die Disambiguierung von Entitäten in Nachrichtenartikeln. Das benötigte Hintergrundwissen ist durch zwei Ontologien gegeben, die semantische Informationen über diese Entitäten im Zusammenhang mit analysierten Artikeln der Nachrichtendomäne enthielt. Die erste Ontologie ist die News-Ontologie [78], die Informationen über die Nachrichtendomäne birgt. Die zweite Ontologie beschreibt die zuvor analysiertem Artikel und die darin enthaltenen Entitäten. Die Vorgehensweise zur Disambiguierung basiert auf einer angepassten Version des PageRank-Algorithmus [170]. Im Gegensatz zu der von Page et al. entwickelten Variante ist es beim diskutierten Ansatz möglich, den Kanten zwischen den Entitäten Gewichte hinzuzufügen. Das Vorgehensprinzip von PageRank, *d.h.* die Relevanz einer Entität hängt von der Relevanz der diese umgebenden Entitäten ab, bleibt hierbei bestehen. Zusätzlich kommen Heuristiken zum Einsatz, die auf die Nachrichtendomäne abgestimmt werden. Insofern wird berücksichtigt, wie häufig die Entität in den vergangenen Tagen in den Nachrichten erwähnt wurde und welcher (Nachrichten-)Kategorie sie zugeordnet ist. Ersteres basiert auf der Annahme, dass Ereignisse über mehrere aufeinander folgende Tage in den Nachrichten vorkommen, während Letzteres über die Zuordnung zu Sport, Politik *etc.* die Anzahl an möglichen Entitäten für einen gegebenen Bezeichner einschränkt. Die mit dem Programm disambiguierten Texte werden mit Menschen nachbearbeitet, so dass die erzeugte Lernphase eine Aktualisierung der Ontologie ist.

Garcia et al. entwickelten eine Erweiterung dieses Ansatzes [89]. In dieser Arbeit erstellen die Autoren eine Verknüpfung je Entität, zu der diese repräsentierenden Wikipedia-Seite und verwenden die Verknüpfungen zwischen den Wikipedia-Seiten als weitere Quelle der Information.

Verfahrensvergleich
Die Informationen innerhalb der News-Ontologie sind im Rahmen dieses Artikels beschränkt auf die gemeinsame

Erwähnung von Entitäten innerhalb von Artikeln, auf deren Kategoriezugehörigkeit sowie auf einer Klassenhierarchie für Nachrichtenthemen. Zusammenhänge zur Disambiguierung von Entitäten innerhalb eines Textes basieren insofern auf den gemeinsam genannten Entitäten in zuvor erschienenen Artikeln und deren ontologischen Beziehungen zueinander. Das Verfahren berücksichtigt jedoch keine zwischen den Entitäten liegenden Entitäten, *z.B.* A kennt B kennt C. Falls B nicht erwähnt ist, wird auch keine Relation zwischen A und C gefunden. Für die Erweiterung, die im Artikel [89] vorgestellt wird, trifft dieses Problem ebenfalls zu. Die Erweiterung nimmt ausschließlich Bezug zur Hintergrundontologie. Der Algorithmus des Verfahrens hingegen bleibt unverändert.

Banek, Vrdodljak, Tjoa: **Word Sense Disambiguation as the Primary Step of Ontology Integration** [17]

Banek et al. weisen in ihrem Artikel darauf hin, dass viele existierende Ansätze im Bereich „Ontology Integration“[8] auf WordNet[9] zur Ermittlung zusätzlicher Informationen zurückgreifen. Um jedoch an diese Informationen zu gelangen, muss zuvor ein Namensvergleich zwischen Konzeptbezeichnern und WordNet-Bezeichnern durchgeführt werden. Hierbei ist insbesondere die möglicherweise vorliegende Ambiguität zu beachten. Die Autoren schlagen hierfür ein Verfahren vor, das Ontologiekonzepte mittels der in WordNet dargestellten Zusammenhänge disambiguiert. Es erfolgt somit eine Überprüfung, welcher WordNet-Eintrag zu welchem Konzept passt. Hierfür kommen der A-Box-Graph der Ontologie sowie ein selbsterstellter Graph, der auf WordNet-Informationen (Homonymen, Hypernymen, Meronymen und Holonymen) basiert, zum Einsatz. Optional kann hierbei auch die Glosse eines WordNet-Eintrags[10] und/oder sein Antonym verwendet werden. Die Disambiguierung basiert auf dem Vergleich des

8 Euzenat und Shvaiko definieren Ontology Integration als *„the inclusion in one ontology of another and assertions expressing the glue between these ontologies, usually as bridge axioms. The integrated ontology is supposed to contain the knowledge of both initial ontologies. Contrary to merging, the first ontology is unaltered while the second one is modified“* [71].

9 `http://wordnet.princeton.edu/` [letzter Zugriff am 12.09.2011]

10 Eine Glosse beschreibt eine textuelle Erklärung zum vorliegenden Begriff.

Zusammenhangs zwischen einem Konzept und den mit diesem Konzept direkt in Beziehung stehenden Konzepten zu deren Abbild in WordNet, *d.h.* jeder individuelle Zusammenhang zwischen zwei Konzepten wird versucht in WordNet wiederzufinden. Je mehr Zusammenhänge aufgefunden werden und je geringer die Distanz zwischen diesen ist, desto höher ist die Wahrscheinlichkeit, dass es sich um das richtige WordNet-Äquivalent des fraglichen Konzepts handelt. Die Autoren schlagen zur Berechnung dieses Verhältnisses zwei unterschiedliche Maße vor: zum einen die Pfadlänge des Pfades zwischen den möglichen Referenzen in WordNet und zum anderen ein Vergleich zwischen der in den Glossen enthaltenen Informationen der WordNet-Einträge über den Pfad hinweg.

Verfahrensvergleich
Das Verfahren interpretiert Disambiguierung als einen Strukturvergleich zwischen zwei Ontologien.[11] Dies ist kein allgemein gültiger Ansatz, da beide Ontologien je nach Domäne keine Ähnlichkeiten zueinander aufweisen müssen. Bei diesem Vergleich profitiert der Ansatz von der Einschränkung der Disambiguierung auf Ontologiekonzepte. Die T-Box einer Ontologie basiert aufgrund ihrer höheren Abstraktion eher auf allgemeinen Begriffen, die ebenfalls in Thesauri wiederzufinden sind. Die Disambiguierung von Instanzbezeichnern ist jedoch mit dem Problem konfrontiert, dass diese – im Gegensatz zu Konzeptbezeichnern – äußerst selten in Thesauri, *z.B.* WordNet[12], enthalten sind. Daher liefert dieses Verfahren keine Lösung für die Disambiguierung von Instanzbezeichnern. Der Ansatz unterscheidet sich somit grundlegend vom Ansatz des Autors dieser Arbeit .

[11] WordNet kann als Ontologie interpretiert werden, da aus lexikalischer Sicht Homonym- und Hypernymrelationen als Ober- und Unterklassen interpretiert werden können. Ebenfalls können Meronym- und Holonyminformationen semantisch interpretiert werden.

[12] WordNet verfügt über einen Teil von Instanzbezeichnern, *z.B.* die Künstlerin Madonna. Dies kann jedoch nicht allgemein auf alle Thesauri übertragen werden und ist in WordNet in der Möglichkeit begründet, dass Benutzer selbständig neue Einträge diesem Thesaurus hinzufügen können. Selbst dies führt nur zu einem kleinen Ausschnitt an möglichen Instanzen und lässt sich nicht als vollständiges Bild für mögliche Domänenbeschreibungen interpretieren.

Zusätzlich zu den hier vorgestellten Arbeiten sind auch die Ansätze [259, 257] des Autors dieser Arbeit zu nennen, die bereits im Abschnitt 7.3 näher vorgestellt werden.

Ebenfalls anzumerken ist die Tatsache, dass alle vorgestellten Verfahren, auch die Verfahren von Nguyen et al. untereinander, unterschiedliche Testdaten verwenden. Dies ermöglicht insofern keinen statistischen Vergleich der Verfahren auf Grundlage der diese beschreibenden Artikel.

14.2. Arbeiten basierend auf Graphenmodellen

Im Folgenden sind Ansätze basierend auf Graphmodellen aufgeführt. Zu jedem der aufgelisteten Verfahren wird ein Vergleich zu dem in dieser Arbeit entwickelten Verfahren vorgenommen und somit deren Gemeinsamkeiten und Unterschiede hervorgehoben.

Veronis und Ide: **Word Sense Disambiguation with Very Large Neural Networks. Extracted from Machine Readable Dictionaries** [235]

Veronis und Ide erstellen einen Ansatz zur Disambiguierung ambiguer Begriffe in einem gegebenen Textausschnitt. Der Ansatz basiert auf einem Wissensmodell, das durch den Ansatz selbst anhand eines vorliegenden Thesaurus erstellt wird sowie einer Spreading Activation Technik zur Bestimmung der wahrscheinlichsten Referenz je gegebenen ambiguen Begriff.

Der gegebene Thesaurus besitzt die in Abbildung 14.1 aufgezeigte Struktur. Der Algorithmus baut anhand diesem ein „very large neural network (VLNN)“ auf, das alle darin enthaltenen Begriffe, deren Sinne und wiederum deren Begriffe im Erklärungstext[13] miteinander verbindet. Hierdurch entsteht eine Verbindung von Bedeutung zu Bedeutung. Beim Vorgang der Disambiguierung werden die durch den Text gegebenen ambi-

[13] Der jeweilige Informationstext, den die Glosse enthält, wird nach Begriffen durchsucht, die sich ebenfalls als exklusive Einträge im Thesaurus wiederfinden. Diese werden dann für die Knoten im Netzwerk verwendet.

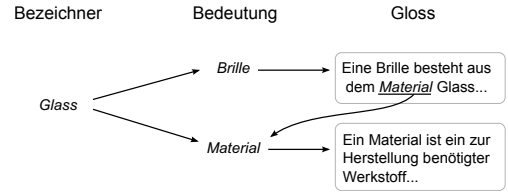


Abbildung 14.1.: Beispiel der Struktur des verwendeten Thesauri (vgl. Ansatz Veronis und Ide [235])

guen Begriffe begriffsspezifisch im Netzwerk aktiviert und via Spreading Activation die entsprechenden Aktivierungswerte weitergegeben. Der verwendete Algorithmus veranlasst eine Breitensuche durch das gesamte Netzwerk und die Wortbedeutung des ambiguen Bezeichners wird durch dessen Bedeutung mit der höchsten Aktivierung, *d.h.* diese wird als Referenz bestimmt.

Verfahrensvergleich
Die Autoren bauen auf einem Graphenmodell ausgehend von einem Thesaurus auf. Die darin enthaltenen Erklärungstexte, die jeder Bedeutung zugeordnet sind (Glossen), sind die Grundlage für Verknüpfungen. Diese selbst erstellten Verknüpfungen unterscheiden sich von ontologischen Instanzbeziehungen, da sie nicht Konzepten zugeordnet sind. Daher können sie beispielsweise nicht für semantische Maße verwendet werden. Hervorzuheben ist der Unterschied in der Spreading Technik. Diese verwendet die unmodifizierte Breitensuche und somit muss der gesamte Graph durchsucht werden, um Resultate zu bestimmen. Als Ergebnis werden ebenfalls die Bedeutungen bestimmt, die durch die Knoten der höchsten Aktivierung repräsentiert werden. Das berechnete Resultat stellt jedoch kein graphbasiertes Zusammenhangmuster dar, wie es durch die Berechnung eines Steinerbaumes gegeben ist.

Bhalotia, Hulgeri, Nakhe, Chakrabarti: **BANKS: Browsing and Keyword Searching in Relational Databases** [1]

Dieser entwickelte Ansatz ist fokussiert auf die Suche anhand von Schlüsselworten in Datenbanken und verwendet

ein Graphmodell der Datenbank. Die Suche ist vergleichbar mit dem in Abschnitt 7.5 vorgestellten Suchverfahren, *d.h.* es werden ebenfalls Steinerbäume erzeugt, die jeweils durch einen Knoten, welche die diesbezüglichen Informationen birgt, repräsentiert werden. Ursprünglich wurde dieses Suchmodell von Palmon et al. [56, 57, 171] entwickelt. Der Ansatz von Bahlotia et al. verwendet jedoch ein eigenes Gewichtungsmodell. Beim Suchalgorithmus zur Erstellung der Steinerbäume handelt es sich um eine Variante des Djikstra-Algorithmus [62]. Der Algorithmus basiert auf einer Rückwärtssuche. Um dies zu ermöglichen, wird eine Verfahrensweise zur Erstellung von zusätzlichen Kanten vorgestellt, die, falls die Kante falsch gerichtet ist, eine künstliche[14], spiegelverkehrte Kante anlegt. Hierfür wird ebenfalls eine Gewichtung eingeführt. Zusätzlich zum Algorithmus selbst wird eine Visualisierungskomponente, die das Durchsuchen bzw. Ordnen von Daten erlaubt, vorgestellt.

Verfahrensvergleich
Das theoretische Vorgehensmodell, *d.h.* die Erstellung von Resultaten anhand der Generierung eines Steinerbaumes, der durch einen Knoten dargestellt wird, entspricht dem Vorgehensmodell des vom Autor dieser Arbeit vorgestellten Ansatzes (vgl. 7.5). Die Grundlagen dieses Modells wurden bereits zuvor entwickelt (siehe [56, 57, 171]) und von Bhalotia et al. ebenfalls aufgegriffen. Die Publikation auf der VLDB-Konferenz weist jedoch nachdrücklich darauf hin, dass die eigentliche Schwierigkeit in der Adaptierung dieses Modells auf der jeweiligen Domäne liegt, *d.h.* im Ansatz auf der Suche in Datenbanken. Im direkten Verfahrensvergleich unterscheiden sich die Ansätze durch das verwendete Hintergrundmodell, die Maße, die algorithmischen Vorgehensweise (*z.B.* Erstellung/Auffinden des Steinerbaumes) und insbesondere durch die Suche von Schlüsselbegriffen zur Disambiguierung von Texten. Bhatolia et al. stellen keine Varianten des Algorithmus vor, *z.B.* die Verwendung maschinellen Lernens *etc.*

[14] Diese Kante ist im ursprünglichen Modell nicht vorhanden.

Kacholia, Pandit, Chakrabarti, Sudarshan, Desai, Karambelkar:
Bidirectional Expansion for Keyword Search on Graph Databases [118]

Bei diesem Verfahren handelt es sich um eine Erweiterung des BANKS-Ansatzes. Diese umfasst zum einen die Verwendung von Spreading Activation zur algorithmischen Umsetzung der Suche und zum anderen ein Bidirektionales Suchverfahren. Die Verwendung von Spreading Activation basiert auf der Vorgehensweise, wie sie unter anderem in den Artikeln [180, 196] und [53] zu finden ist. Im Ansatz von Kacholia et al. werden Anpassungen hinsichtlich der Struktur von BANKS vorgenommen. Bidirektionalität bedeutet hier, dass zunächst alle eingehenden Kanten eines Knotens und später (basierend auf einer weiteren Prozessierungsliste) alle ausgehenden Kanten untersucht werden. Somit wird der Graph nicht ausschließlich nach dem aus der Kantenrichtung folgenden Prinzip der Rückwärtsexploration aufgebaut.

Verfahrensvergleich
Der hier vorgestellte Ansatz ist technisch verwandt mit dem Ansatz des Autors dieser Arbeit . Das liegt an der hohen Übereinstimmung der Vorgehensweise an dem Suchprinzip via Spreading Activation (vgl. [53, 47]). Wesentliche Unterschiede sind die Gewichtung von Kanten (*z.B.* Spreadingfunktion) und Knoten (*z.B.* Gesamtaktivierung) sowie der Prozess (*z.B.* keine Kantenrichtungen) *etc.* Die Bidirektionalität des Ansatzes von Kacholia et al. unterscheidet sich von der in Kapitel 9 vorgestellten Bidirektionalität. Für das in diesem Kapitel vorgestellte Verfahren ist nicht die Kantenrichtung entscheidend, sondern der vollständige, beidseitige Informationsaustausch zwischen Ursprungs- und Zielknoten der aktuellen Exploration, während im Ansatz von Kacholia der Austausch einseitig bleibt.[15]

[15] Kacholia et al. beschreiben, dass die Bidirektionale Suche *"improves [...] backward expanding search by allowing forward search from potential roots towards leaves"* [118]. Das fügt somit ausschließlich die Exploration vorwärts gerichteter Kanten hinzu. Die von Kacholia et al. entwickelte bidirektionale Suche ist nicht zu verwechseln mit dem bidirektionalen Ansatz, der in dieser Arbeit entwickelt wurde(siehe Kapitel 9).

Rocha, Schwabe, Poggi de Aragão: **A Hybrid Approach for Searching in the Semantic Web** [190]

Der Ansatz von Rocha et al. beschreibt ein Verfahren für die Suche von Informationen innerhalb einer Ontologie basierend auf der Spreading Activation Technik. Anhand von gegebenen Stichwörtern schließt der Algorithmus auf das relevante Suchergebnis. Dieses muss nicht zwangsläufig in Ontologieelementen resultieren, deren Bezeichner die Stichwörter beinhalten, sondern kann auch auf Ontologieelemente hinweisen, für die dies nicht zutrifft. Die Ontologie wird zunächst in einen Instanzgraphen transformiert. Die Kanten zwischen Knoten werden anhand zweier Metriken initial gewichtet. Das erste Maß ist ähnlich wie das in Abschnitt 12.4.2 vorgestellte Maß. Es gewichtet zum einen die Menge der Instanzrelationen für einen gegebenen Relationstyp zwischen zwei Konzepten im Verhältnis zu allen ausgehenden Instanzrelationen dieses Typs vom ersten Konzept. Zum anderen wird eine Gewichtung für die Ähnlichkeit zweier Konzepte verwendet. Alle Knoten, welche die gegebenen Schlüsselbegriffe repräsentieren, werden initial aktiviert und propagieren ihre Aktivierung an die mit diesem in Beziehung stehenden Knoten. Zuletzt werden die Knoten mit der höchsten Aktivierung zurückgegeben. Diese Knoten müssen nicht zwangsläufig eines der Schlüsselwörter im Bezeichner tragen.

Verfahrensvergleich
Das Verfahren folgt ebenfalls den Grundsätzen von Crestani [53] und Cohen [47]. Im Unterschied zum Verfahren des Autors dieser Arbeit wird die Aktivierung nicht spezifisch je Schlüsselwort weitergegeben und als Resultat kein Graph zurückgegeben, wenngleich dieser durch das Spreading entsteht. Auch wird die Mehrdeutigkeit von Begriffen innerhalb des Prozessierens nicht berücksichtigt. Das Verfahren ist fokussiert auf die Schlussfolgerung des wahrscheinlichsten Ontologieelements, das im Zusammenhang mit den Begriffen steht, *d.h.* das Element, welches am Ende des Spreading-Prozesses den höchsten Aktivierungswert besitzt. Dies ist vergleichbar mit dem Knoten, der im Verfahren des Autors dieser Arbeit den Lösungsbaum, *d.h.* dessen Wurzel, repräsentiert, da dieses ebenfalls den höchsten Gesamtakti-

vierungswert im Graph besitzt. Jedoch sind die Lösungen der Verfahren nicht übereinstimmend, da im Rahmen einer Disambiguierung für die Referenzbestimmung eines Begriffs ein Ontologieelement benötigt wird, das die Referenz der Entität bzw. des gegebenen Bezeichners angibt und dies in der Lösung des Verfahrens von Rocha et al. nicht in allen Fällen gegeben ist.

Hasan: **A Spreading Activation Framework for Ontology-enhanced Adaptive Information Access within Organisations** [105]

Md Maruf Hassan entwickelte einen Ansatz, der es erlaubt Informationen innerhalb eines erweiterten ontologischen Modells aufzuspüren. „Erweitert“ bedeutet, dass es sich zwar um eine zuvor hinterlegte Ontologie handelt, diese jedoch nicht für jeden Entitätsbezeichner ein Ontologieelement mit derselben Bezeichnung enthält. Somit müssen diese Elemente teilweise dynamisch erzeugt werden. Der Aufbau des Graphen erfolgt durch eine Analyse eines gegebenen Textkorpus. Für jedes Dokument wird ein Knoten erstellt und diesem, die aus dem Dokument extrahierten Entitäten, hinzugefügt. Sind die Entitäten in der Ontologie vorhanden, so werden die Informationen (Data-Properties) über diese verwendet. Falls Entitäten im Text erkannt wurden, die nicht in der Ontologie enthalten sind, werden deren möglicherweise zugehörigen Data-Property-Informationen ebenfalls aus dem Text extrahiert und der Ontologie hinzugefügt. Diese Analyse ermöglicht auch die Erweiterung von Informationen zu Entitäten, die bereits in der Ontologie enthalten sind. Das Netzwerk ergibt sich aus den Zusammenhängen zwischen Dokumenten, die dieselben Entitäten enthalten. Bei der Suche werden die Schlüsselbegriffe mit den Data-Properties der Entitäten verglichen. Im Falle eines positiven Vergleichs werden die Entitäten aktiviert. Es erfolgt ein Spreading durch das Netzwerk. Als Resultat wird die Menge der höchstaktivierten Knoten zurückgegeben. Eine adaptive Anpassung durch den Benutzer ist durch eine höhere Gewichtung einzelner Properties möglich. Außerdem ist Feedback durch den Benutzer möglich, welches für die Gewichtung in der darauffolgenden Suche berücksichtigt wird.

Verfahrensvergleich
Dem Benutzer wird das semantische Netzwerk in der Visualisierung angezeigt sowie die Knoten, die als Resultate der Anfrage zurückgegeben werden. Hierbei handelt es sich nicht um die Berechnung eines Lösungsgraphen. Die Informationen innerhalb des berücksichtigten Graphen beschränken sich auf die Relationen vom jeweiligen Dokument zu den in diesem enthaltenen Entitäten sowie den Data-Properties, die mittels String-Vergleich später ausgewertet werden. Direkte Relationen zwischen Entitäten sind nicht vorhanden.[16] Es werden keine spezifischen Aktivierungswerte verwendet, sondern ein allgemeiner Aktivierungswert. Das Verfahren ermöglicht jedoch die Einbindung eines Lernprozesses, der Einfluss auf Gewichtungen der Graphelemente in darauffolgenden Suchanfragen hat. Es folgt den Grundsätzen des in Kapitel 11 vorgestellten Ansatzes mit Bestärkendem Lernen. Hassan verwendet jedoch überwachtes Lernen, *d.h.* der Mensch ist die Quelle des Priorisierens im Verfahren.

Tsatsaronis, Vazirgiannis, Androutsopoulos: **Word Sense Disambiguation with Spreading Activation Networks Generated from Thesauri** [230]

Tsatsaronis et al. stellen – ähnlich zu den zuvor genannten Verfahren – ein Verfahren zur Disambiguierung mittels Spreading Activation vor, das auf einem Thesaurus aufbaut. Sie verwenden übereinstimmend mit zuvor genannten Ansätzen WordNet als Thesaurus. Disambiguiert werden Begriffe innerhalb eines gegebenen Dokumentes. Zunächst wird das Dokument in seine Sätze aufgeteilt und anschließend jeder Satz für sich disambiguiert. Hierbei erfolgt zunächst eine Ermittlung der Begriffe (*z.B.* Nomen, Verben *etc.*), die sowohl im untersuchten Satz als auch im Thesaurus enthalten sind. Übereinstimmend mit dem Ansatz von Banek [17] wird aus den Beziehungen der Thesaurieinträge (*z.B.* Hyponyme, Homonyme *etc.*) das semantische Netzwerk erstellt. Zu Beginn der Analyse erfolgt die Aktivierung aller Knoten, die Begriffe des Satzes

[16] Indirekt sind Relationen über gleiche Data-Property-Werte gegeben.

repräsentieren. Das Verfahren verwendet Spreading Activation, um die Aktivierungen weiterzugeben. Das Spreading wird iterativ fortgesetzt, bis die Schranke des minimal weiterzugebenden Aktivierungswertes einsetzt. Somit bildet der zuletzt aktive Knoten, aus der Menge der Knoten, die denselben Begriff repräsentieren, die Referenz zum Begriff.

Verfahrensvergleich
In diesem Verfahren werden alle Begriffe als gleichwertig betrachtet, *d.h.* es gibt keine ontologische Unterscheidung, *z.B.* in Konzepte, Instanzen, *etc.* Der wesentliche Unterschied zum Verfahren des Autors dieser Arbeit liegt in der Art des Spreading Activation-Algorithmus und in der Auswertung der durch diesen erzeugten Resultate. Insbesondere stellt WordNet bzw. Thesauri im Allgemeinen - wie zuvor bereits dargelegt - eine spezielle Art von Ontologien dar, die auf linguistische Zusammenhänge fokussiert sind. Das Verfahren ist nicht ohne Anpassungen auf domänenspezifische Ontologien, die diese linguistischen Zusammenhänge nicht beinhalten, zu übertragen.

Weitere Graph-Ansätze mit dem Ziel Disambiguierung bzw. Suche Im Allgemeinen lassen sich zwei Vorgehensweisen bei der Graphanalyse unterscheiden. Zum einen versuchen Verfahren durch das Zusammenfügen[17] von Teilgraphen einen Graphen zu erstellen, der gemeinsamen Kriterien entspricht bzw. eine gemeinsame Bedeutung aufweist und damit eine Minimierung der Ambiguität durch Zusammenfassung erreicht wird. Zum anderen erfolgt der Aufbau bzw. die Extraktion eines Graphen durch die sukzessive Exploration eines hinterlegten Graphen. Der Anspruch an diesen (Teil-)Graph ist es, die im Kontext der Applikation korrekten Referenzen zu enthalten.

Die Ansätze von Aggire et al. [2], Byung et al. [168] und Ferret [80] liefern Beispiele für die zuerst genannte Vorgehensweise. Der Ansatz von Aggire et al. [2] reichert korrekt disambiguierte Bezeichner mit Kontextworten aus der Beschreibung der entsprechenden WordNet-Äquivalente an und legt je Referenz ein initiales Kontextcluster an. Diese Kontextcluster werden anschließend sukzessive anhand von

[17] *engl.* clustern

Übereinstimmungen der Kontextwörter zusammengefügt. Letztlich bleiben nur noch die unterschiedlichen Bedeutungen eines ambiguen Wortes übrig. Ähnlich arbeitet auch das Verfahren von Ferret [80]. Hier werden die Zusammenhänge zwischen den Clustern mittels Kosinus-Maß[18] bestimmt. Nach diesem Vorverabeitungsschritt erfolgt die Umwandlung in einen Graphen, dessen Kantengewichte durch dieses Maß beschrieben werden. Die weitere Verarbeitung erfolgt über den Shared Nearest Neighbors-Algorithmus [68]. Byung [168] verwendet ebenfalls eine Ähnlichkeitsmatrix als Grundlage und ermöglicht die Disambiguierung mittels Graphpartitionierung.

Beispiele der zweiten Methodik werden bereits im Abschnitt 14.1 und zu Beginn dieses Abschnitts 14.2 vorgestellt. Hierzu zählen auch die Ansätze von Malaisè et al. [146] und Rada et al. [187, 215]. Malaisè et al. beschreiben ein Verfahren, das ebenfalls auf WordNet als Hintergrundwissen zurückgreift. Eine gegebene Wortmenge wird vom Verfahren auf bereits vorhandene Knoten innerhalb des WordNet-Graphen übertragen. Die Disambiguierung erfolgt abermals durch die Suche nach einem zentralen Knoten, der eine Verbindung zu Referenzen für alle benötigten Begriffe besitzt.

Der Ansatz von Rada et al. basiert auf einem Random Walk[19] Ansatz, *d.h.* der Erstellung eines Graphen, der die initial gegebenen Wörter beinhaltet. Rada setzt zunächst das Maß von Lesk [136] ein. In einer späteren Arbeit [215] evaluiert Rada verschiedene weitere Maße für diesen Ansatz. Für den Anwendungsfall der Schlüsselwortsuche entwickelte Tong den Center-Piece-Subgraph-Algorithmus, *d.h.* ein Verfahren, in dem der Zusammenhang zwischen den Schlüsselwörtern durch einen Graph ausgedrückt wird und dieser durch einen Knoten - Center-Piece - repräsentiert wird. Hierzu entwickelt er einen neuen Random-Walk-Algorithmus, der auch komplexe Anfragen mit „oder“ *etc.* unterstützt.

---

18 Siehe [226] für nähere Informationen bezüglich des Kosinusmaßes.

19 Rada definiert „Random walks“ als *„The random walks are mathematically modeled through iterative graph based algorithms, which are applied on the label graph associated with the given sequence of words, resulting in a stationary distribution over label probabilities. These probabilities are then used to simultaneously select the most probable set of labels for the words in the input sequence.“* [187]. Durch die Definition wird deutlich, dass die Bezeichnung auf die Verfahren dieser Arbeit als auch auf viele der zuvor vorgestellten Ansätze zutrifft, *z.B.* auf die Ansätze PageRank [170] und BANKS [1].

Asawath et al. [15] setzen einen zweigeteilten Spreading-Activation-Algorithmus ein. Zunächst erfolgt ein Spreading für die primären Sinnzuordnungen, danach für die Synonyme. Darauf folgt eine Klassifizierung anhand einer Support-Vector-Engine, die in positive und negative Informationen einteilt, was zu einer unterschiedlichen Resultatauswertung führt.

Thanh et al. [229] entwickelten ein Suchverfahren, dass zusätzlich in der Lage ist, initiale Begriffe auch Kanten zuzuordnen. Der Algorithmus weist hohe Ähnlichkeit zum BANKS-Algorithmus auf [1]. Es wird ein Iterator je Menge der Knoten, die über ein Suchwort aufgefunden werden, verwendet. Die meisten Ansätze verwenden nur einen Iterator für die gesamte Menge der über die Suchwörter aufgefundenen Knoten. Auch werden individuelle Kantenmaße verwendet. Bei Ihrem Ansatz gibt der Benutzer zunächst Stichwörter ein, die seine Suche beschreiben. Der Ansatz ermittelt daraus mögliche Anfragen und bietet diese dem Benutzer an, der diese konkretisieren und somit eine fokussiertere Suche durchführen kann.

Der BLINKS Algorithmus von Hao et al. [109] baut ebenfalls auf dem BANKS-Algorithmus von Bhatolia auf. Er verändert jedoch die Auswahl des als nächstes zu explorierenden Knotens. Diese Auswahl orientiert sich am Explorationsumfang je Teilcluster (getrenntem Teilgraph), da diese sich möglichst entsprechen sollten. Daher wird immer das Cluster mit dem geringsten Umfang als nächstes exploriert. Zudem führt er eine Indexstruktur ein, die eine schnellere Exploration gewährleistet. Ebenfalls in diesem Abschnitt explizit zu erwähnen ist das PageRank-Verfahren [170] von Page und Brin, das ebenfalls als Random Walk Algorithmus bezeichnet werden kann. Hierbei wird ein Knoten[20] im Graph durch die Popularität seiner Nachbarknoten rekursiv gewichtet.

[20] Jeder Knoten repräsentiert eine ihm zugeordnete Webseite.

Teil IV.

Schlussbetrachtung

15. Schlussfolgerungen und Ausblick

Durch die Konfrontation mit der Problematik der Ambiguität, *d.h.* der multiplen Vergabe des gleichen Bezeichners für verschiedene Entitäten bzw. der Vergabe mehrerer Bezeichner für eine Entität, wird das Interesse des Autors geweckt. Die Tatsache, dass eine Vielzahl der umfangreichen Arbeiten hinsichtlich dieses Themas in der Vergangenheit überwiegend auf rein textuellem Wissen basierten, war Anlass für den Entwurf eines Verfahrens, das sich durch die Zusammenführung textueller Information mit ontologischen Wissen auszeichnet. Dieses baut auf der Kernannahme auf, dass die Disambiguierung von Entitäten insbesondere durch die Verwendung von ontologischen Zusammenhängen erreicht werden kann (vgl. Abschnitt 1.2). Ausschlaggebend für diese These ist die bereits bestehende Möglichkeit, die Interpretation von textueller Information dazu zu verwenden, um eigene Hintergrundmodelle aufzubauen. Das ist beispielsweise ein wesentlicher Faktor bei Verfahren des Maschinellen Lernens. Jedoch ist das nicht zu vergleichen mit einem auf die Domäne abgestimmten Modell, das individuell erzeugt wird, um die Zusammenhänge der Domäne zu repräsentieren. Die vorliegende Arbeit zeigt auf, dass die solch Modelle selbst die Ambiguität jedoch nicht auflösen, sondern zunächst selbst in sich tragen.

15.1. Schlussfolgerungen

Die in dieser Arbeit vorgestellten Verfahren machen sich das ontologische Modell zunutze und ermöglichen ambigue Bezeichner, die in einem die Domäne betreffenden Zusammenhang genannt werden (*z.B.* in Texten oder in Gesprächen), den korrekten Ontologieelementen zuzuordnen. Diese Arbeit entwirft ein Vorgehensmodell, das die

Disambiguierung mehrdeutiger Bezeichner ermöglicht. Das Modell ist jedoch nicht allein auf mehrdeutige Bezeichner beschränkt, sondern ermöglicht es im Allgemeinen, die gegebenen Bezeichner den korrekten, *d.h.* als Referenz fungierenden, Ontologieelementen zuzuweisen. Diese Zuweisung basiert auf einer Analyse der Zusammenhänge innerhalb der externen Quelle. Zur Erstellung dieses Modells müssen die in Abschnitt 1.2 vorgestellten Forschungsfragen individuell erörtert werden. Die diesbezüglichen Ergebnisse sind im Folgenden zusammengefasst:

Linguistische Ambiguität im ontologischen Modell: Durch die Arbeit wird zunächst die Bedeutung des Ausdrucks „linguistische Ambiguität" aufgezeigt werden. Hierzu werden deren multiple Facetten, *d.h.* Polysemie, Homonymie, *etc.* vorgestellt. Insbesondere werden semantische und strukturelle Mehrdeutigkeiten, besprochen. Diese Arbeit stellt die Übertragung der Problematik Ambiguität auf ein allgemeines Modell vor, das die Bereiche der Intension und Extension für die Beschreibung der Ambiguität verwendet. Das ist zugleich die Grundlage zur Erfassung von Ambiguität in Ontologien, *d.h.* der Art und Weise inwiefern sich Ambiguität in einer Ontologie äußern kann. Aufbauend auf diesem Modell wird insbesondere die Ambiguität von Entitäten in Ontologien analysiert. Hierbei liefert die Intension, die Zuordnung des im Lexikon hinterlegten Bezeichners zu den Ontologieelementen, während die Extension, die in der Ontologie zugeordneten Eigenschaften zu einem die Entität repräsentierenden Ontologieelement, beschreibt. Diese aus der linguistischen Ambiguität für die ontologische Ambiguität folgenden Zusammenhänge sind die Grundlage für den Entwurf der in dieser Arbeit vorgestellten, darauf aufbauenden Disambiguierungsverfahren.

Referenzbestimmung anhand semantischer Zusammenhänge zwischen Entitäten:
Das entworfene Disambiguierungsverfahren macht sich die ontologischen Extensionen der Entitäten zunutze. Zunächst jedoch erfolgt die Bestimmung der möglichen Ontologieelemente je gegebenen Bezeichner über die zugehörige Intension. Der Kern des Verfahrens basiert auf der Annahme eines gegebenen ontologischen Zusammenhanges zwischen den Extensionen von mindestens einer Adresse innerhalb der Intension je Bezeichner.

Diese Annahme ist zum einen mit multiplen Zusammenhängen dieser Art in der gegebenen Ontologie konfrontiert und somit muss ein Priorisieren semantischer Beziehungen anhand festgelegter Kriterien erfolgen. Zum anderen ist sie der Gefahr ausgesetzt, dass kein Zusammenhang ermittelt werden kann. Dies veranlasst die Erstellung eines künstlichen Zusammenhangs, der ein Ontologieelement für jeden Bezeichner aufweist. In dieser Arbeit wird davon ausgegangen, dass dieser Zusammenhang durch einen Steinerbaum beschrieben werden kann. Demzufolge ist das vorgestellte auf Spreading Activation basierende Verfahren dazu optimiert diesen aufzufinden. Für beide Problemstellungen, *d.h.* das Priorisieren und den künstlichen Zusammenhang, werden in der vorliegenden Arbeit Möglichkeiten der Problemlösung vorgestellt.

Möglichkeiten der Qualitätsverbesserung: Die Möglichkeit der Extraktion eines Graphen, der den Zusammenhang zwischen Entitäten ausnutzt und zur Disambiguierung verwendet wird, ist begründet in der Überlappung von Extensionen. Dieses Merkmal ist die Grundlage für den Entwurf eines Verfahrens zur Bestimmung eines Steinerbaums. Eine Optimierung des Basisverfahrens stellt die gleichzeitige Entwicklung von Maßen dar, die auf die semantische Einbettung der Entitäten in der Ontologie Bezug nehmen. Weitere Möglichkeiten der Optimierung finden sich in den Verfahrensvarianten. Die Änderung des Algorithmus durch die Berücksichtigung eines Bidirektionalen Explorationsmodells ermöglicht einen umfassenderen Austausch der Informationen und ergibt eine Verbesserung, die in der Evaluation der Verfahren (vgl. Kapitel 13) nachgewiesen wird. Die Verwendung von lokaler Kohärenz ermöglicht eine spezifische Optimierung hinsichtlich lokal gebündelter Information. Insbesondere ist der Umfang der lokalen Kohärenz, *d.h.* die Definition von „lokal“ ein spannendes Feld zur Qualitätsverbesserung. Wiederkehrende Informationen, *d.h.* die Wiederholung von Teilen vorhandenen Wissens in Texten, gab Anlass für die Einführung eines Ansatzes mit Bestärkendem Lernen. Dieser ermöglicht eine Steigerung der Ergebnisqualität durch die Verwendung zusätzlichen Wissens in der Analyse neuer Texte, das auf zuvor durchgeführte Analysen zurückgeht. Neben der Adaption durch die vorgestellten Verfahrensvarianten ergibt eine Fokussierung der

Exploration auf Aktivierungswerte eine weitere Optimierung des Basisverfahrens.

15.2. Ausblick

Das im Rahmen dieser Arbeit entwickelte Verfahren ist darauf ausgelegt, dass es individuell adaptiert werden kann. Durch die flexible Einbindung neuer Maße, sowie durch die Möglichkeit Ontologieelemente zuvor individuell zu gewichten. Unabhängig davon gibt es eine Vielzahl möglicher Erweiterungen und Änderungen, die Chancen zur Steigerung der Qualität des vorgestellten Verfahrens bieten.

Textanalyse: Ein konkretes und spannendes Forschungsfeld, das direkt mit dieser Arbeit in Zusammenhang steht, ist die Auswertung textueller Informationen. Zusätzlich zu den bisher berücksichtigten Informationen zu Instanzen, kann eine Analyse weiterer Informationen hinsichtlich der Ontologieelemente erfolgen. Der Zusammenhang zwischen Ontologiebezeichner und Entitätsbezeichner im Text ist oftmals nicht direkt, *d.h.* eine vollständige Übereinstimmung aller Buchstaben liegt nicht in allen Fällen vor. Dabei helfen Verfahren zur Ermittlung von Wortzusammenhängen, *z.B.* der Zuordnung von Namensvarianten, Spitznamen, Geschlechtsbezeichnern *etc.* Neben den Standardmaßen (*z.B.* Levenshtein) sind insbesondere Heuristiken von Bedeutung. Hierbei ist darauf zu achten, dass ontologische Merkmale ebenfalls durch Begriffe beschrieben werden, *z.B.* Firma, Person *etc.* Letztere ermöglichen die Einschränkung der Referenzsuche auf Instanzen bestimmter Konzepte. Insgesamt gilt es Verfahren der textuellen Analyse zu entwerfen, die eine Abbildung der textuellen Information auf die Ontologieelemente ermöglichen bzw. zusätzlich Hinweise zur Ermittlung des Zusammenhangs in der Ontologie berücksichtigen.

Zusätzliche Wissensbasen: Neue Möglichkeiten bietet die Erweiterung der Informationen innerhalb der zu berücksichtigenden Ontologie durch weitere Wissensbasen. Diese besitzen großes Potential, da sie zusätzliche Informationen bereitstellen können. Hierbei muss ermittelt werden, welche Ontologieelemente der verschiedenen Wissensbasen miteinander in Beziehung

stehen. Diese Problematik kann als separat vom Kernproblem der existierenden Ambiguität betrachtet werden. In der Übersicht der verwandten Arbeiten werden bereits Verfahren vorgestellt, die Informationen aus allgemeinen Lexika, *z.B.* WordNet oder Wikipedia, versuchen aufzugreifen und für die Disambiguierung zu verwenden. Jedoch stehen diese Lexika nicht generell als Ontologien zur Verfügung stehen. Die Erweiterung ontologischer Information ist ebenfalls durch eine aus der Analyse der Texte folgende Übernahme textueller Information in die Wissensbasis möglich, das betrifft Informationen, die in dieser Basis zuvor nicht enthalten sind. Voraussetzung hierfür ist die angesprochene zusätzliche Disambiguierung, um die innerhalb des Integrationsprozesses auftretende Ambiguität zu überwinden.

Maße: In Bezug zur algorithmischen Vorgehensweise zeigen die eingesetzten Maße direkte Auswirkung auf die Qualität des Ergebnisses der Disambiguierung. Daher ist einerseits die Entwicklung von semantischen Maßen von großer Relevanz für das vorgestellte Verfahren, da diese versuchen die semantischen Zusammenhänge anhand numerischer Werte zu beschreiben. Kriterien hierfür sind Konzeptzugehörigkeit, Kantentypen, Relationsverteilungen *etc.* Andererseits ist neben der Verwendung semantischer Maße auch die Verwendung textueller Maße von Bedeutung. Diese sind nicht auf die initiale Gewichtung von einzelnen Ontologieelementen beschränkt, sondern können ebenfalls individuelle Relationen zwischen Entitäten, Konzeptzugehörigkeiten *etc.* anhand von Gewichten ausdrücken. Auch hinsichtlich des Ansatzes der lokalen Kohärenz ist deren Umfang sowie inhaltlicher Fokus eine spannendes Feld zur Qualitätsverbesserung.

Lernverfahren: In dieser Arbeit wird ein Verfahren des Bestärkenden Lernens verwendet. Neben diesem kann auch ein überwachtes Lernverfahren, *z.B.* Support Vector Machine *etc.*, zum Einsatz kommen. Es ist möglich, sowohl textuelle als auch ontologische Merkmale zu berücksichtigen. Die Aufgabe besteht bei einem solchen Verfahren in der Zusammenführung beider Informationen, um die Qualität des Algorithmus zu steigern. Auch unüberwachte Lernverfahren können hierbei zum Einsatz kommen. So kann beispielsweise via Clustering der zu analysierende Korpus vorprozessiert werden. Das kann eine

Zusammenführung der maximalen Informationen je Bezeichner bewirken, die im späteren Algorithmus in die Analyse eingebettet werden können. Auch ist die Erweiterung des Ansatzes des bestärkenden Lernens möglich, *z.B.* durch die Interaktion mit Menschen, die mögliche Feedback-Informationen an das Programm weitergeben.

Algorithmus: Ansatzpunkte für eine algorithmische Optimierung liegen in der Weitergabe und Initialisierung der Aktivierungswerte. Möglichkeiten der Erweiterung sind beispielsweise ein paralleler Aktivierungsaustausch je Bezeichner. Der Spreading Activation Algorithmus gilt als Näherung zum menschlichen Prozessieren von Problemstellungen, *d.h.* der Konfrontation mit initialen Merkmalen und deren individueller Nachverfolgung. Letztlich wird das Resultat das Ende der Aktivierungskette bestimmt. Eine exakte Analyse menschlicher Disambiguierungsprozesse und deren Übertragung auf das vorgestellte algorithmische Vorgehensmodell bietet ebenfalls die Möglichkeit einer Steigerung der Ergebnisgüte.

Anwendungsbereiche: Ein möglicher Anwendungsbereich für das in dieser Arbeit vorgestellte Verfahren existiert, sobald eine Identifikation von semantischem Inhalt ausgehend von natürlichsprachlichen Medien notwendig bzw. gewünscht wird. Das birgt viele Möglichkeiten, insbesondere weil der größte Umfang der heutzutage verfügbaren Informationen in natürlicher Sprache vorliegt. Beispiele für Anwendungen sind:

- Überprüfung der Konsistenz des Inhalts, *z.B.* die Überprüfung von Aussagen durch Beurteilung des Graphaufbaus oder automatisches Schlussfolgern nach Identifikation der Ontologieelemente.
- Erweiterung des gegebenen Wissens
  - Das Verfahren kann in der Ontologie weitere Instanzen von Relevanz aufzeigen, die nicht im Text genannt, jedoch im Lösungsgraphen enthalten sind.

  - Wird eine Disambiguierung anhand verschiedener Ontologien basierend auf dem gleichen Text durchgeführt, so kann dies unterschiedliche Zusammenhänge hervorbringen. Das kann als Ausgangspunkt für Erweiterungen innerhalb einer der Ontologien verwendet werden.

- Ordnung von Wissen, *z.B.* in welchen Texten werden dieselben Zusammenhänge zwischen Entitäten benannt (Disambiguierung ist hier Voraussetzung). Das ist auch sprachübergreifend möglich.

Die aufgelisteten Beispiele zeigen nur einen Ausschnitt möglicher Anwendungsfälle. Die Möglichkeit der Verarbeitung von textuellem Inhalt und dessen Zuordnung zu semantischen Wissensbasen birgt die Zuordnung prinzipiell unstrukturiertem zu strukturiertem Wissen. Disambiguierung bildet hierbei die Ausgangsbasis für jeden Anwendungsfall. Die Entwicklung von Anwendungen und zugleich das damit verbundene Potential weiterer an den Anwendungsfall angepassten Forschungsfragen stellt ein spannendes Feld für Weiterentwicklungen bereit.

Teil V.

Anhang

A. Ambiguität

Im Folgenden ist eine Übersicht über die in Kapitel 5 vorgestellten Arten von Ambiguität dargestellt.[1] Lexikalische und struktuelle Ambiguität werden unterschieden und die zugeordneten Unterarten vorgestellt. Die Darstellung A.1 ermöglicht einen direkten Vergleich zwischen den verschiedenen Arten von Ambiguität. Die wichtigsten Merkmale je Ambiguitätsart werden dargestellt und durch zugehörige Beispiele ergänzt.

Auf diese Strukturierung der verschiedenen Arten von Ambiguität folgt Darstellung A.2, die sich den wichtigsten Kriterien zur Auflösung von Polysemie, Homonymie und syntaktischer Ambiguität widmet. Diese enthält jeweils eine Beschreibung der Vorgehensweise zur Disambiguierung, welche auf die auf die angegebenen Ambiguitätsvariante jeweils fokussiert ist und an Beispielen erläutert wird.

[1] An dieser Stelle möchte ich mich bei Frau Joanna Hareza bedanken mit deren Hilfe und großem Engagement diese Übersicht entstand. Als Wissen, das als Ausgangspunkt zur Erstellung dieser Übersicht diente, sind die schriftlichen Quellen [13, 28, 192, 114, 203, 90, 91, 85, 102, 121, 144, 144, 165, 169, 177, 182, 189, 201, 138, 232, 237, 239] und die Online Quellen `http://www.uni-leipzig.de/~doelling/veranstaltungen/semprag4.pdf` [letzter Zugriff am 12.09.2011] (Uni Leipzig, Ambiguität), `http://lexikologie.perce.de/wb/?l=742B78764B&v=` [letzter Zugriff am 12.09.2011] (Lexikologie.de, Mehrdeutigkeit) sowie `http://www.fask.uni-mainz.de/inst/iaspk/Linguistik/Morphologie/Wortbildung.html` [letzter Zugriff am 12.09.2011] und `http://indigo2.de/fhw/master/seminar/4.html` [letzter Zugriff am 12.09.2011] zu nennen. Viele dieser Quellen werden bereits an anderen Stellen innerhalb dieser Arbeit genannt.

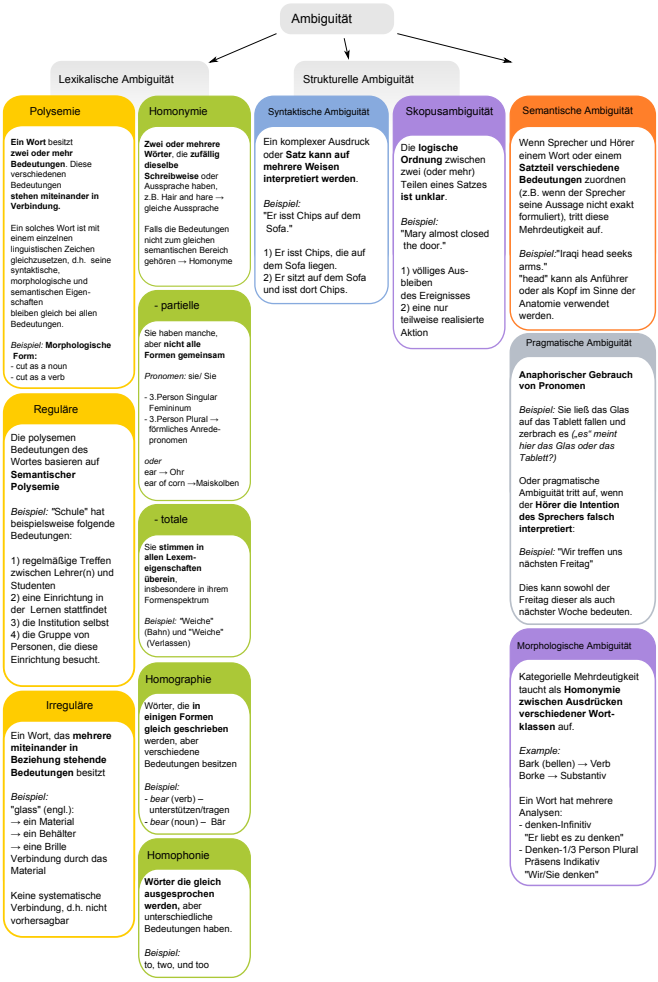


Abbildung A.1.: Typen von Ambiguität

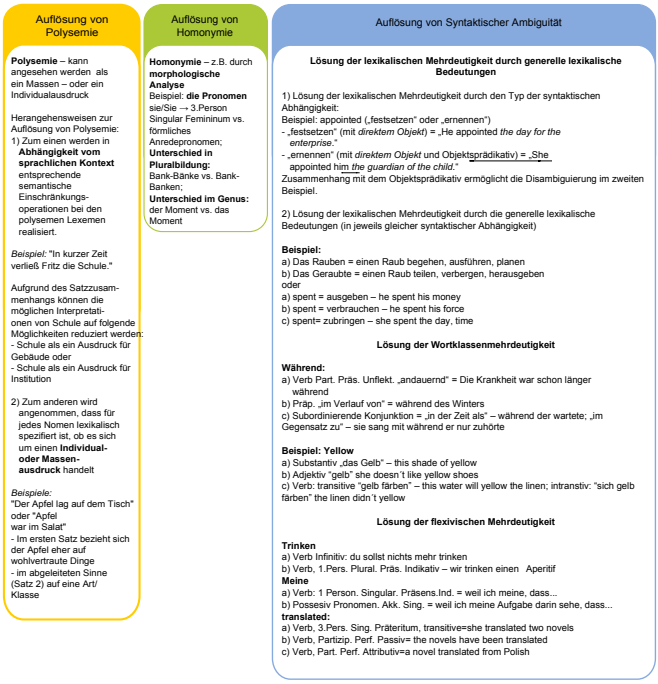


Abbildung A.2.: Auflösung von Polysemie, Homonymie und syntaktischer Ambiguität

B. KIM Ontologie

B.1. Verteilung der Entitäten innerhalb der KIM-Ontologie

In Tabelle B.1 sind die von Ontotext [113] genannten statistischen Werte bezüglich der in der KIM-Ontologie genannten Entitäten dargestellt. Hervorzuheben ist, dass durchschnittlich jede Entität über 1,42 zugewiesene Namen verfügt. Dies belegt eine vorhandene Mehrdeutigkeit hinsichtlich der Entitätsbezeichner. Weiterhin zeigt sich der allgemeine, *d.h.* nicht domänenfokussierte Charakter der Ontologie über die verwendeten Entitätstypen, *d.h.* Ontologiekonzepte.

Konzept	Anzahl
Entitäten	77561
Aliase von Entitäten	110308
Orte	49348
Städte	4720
Firmen	7906
Öffentliche Firmen	5150
Schlüsselpersonen (Positionen)	5500
Organisationen (mit Firmen)	8365

Tabelle B.1.: Übersicht der Entitätsverteilung innerhalb der KIM-Ontologie (entnommen von [113])

B.2. Hinzugefügte Tripel

Nguyen [163] gibt an, die verwendete KIM-Ontologie um zusätzliche Tripel erweitert zu haben. Wie zuvor bereits angesprochen, war

es Nguyen nicht möglich auf Anfrage die von ihm vorgenommene Erweiterung dem Autor dieser Arbeit zur Verfügung zu stellen. Als Konsequenz daraus nimmt der Autor dieser Arbeit eine eigene Erweiterung der KIM-Ontologie vor. Die zum Einsatz gekommenen Tripel sind in Tabelle B.2 aufgelistet.

Subjekt	Prädikat	Objekt
South Georgia and the South Sandwich Islands	smw://protonu#historyGovernedBy	Falkland Islands
South Georgia and the South Sandwich Islands	smw://protonu#country	United Kingdom of Great Britain and Northern Ireland
Princeton University	smw://protont#locatedIn	United States
Massachusetts Institute of Technology	smw://protont#locatedIn	United States
Stanford University	smw://protont#locatedIn	United States
John McCarthy - Computer Scientist	smw://protont#withinOrganization	Princeton University
John McCarthy - Computer Scientist	smw://protont#withinOrganization	Massachusetts Institute of Technology
John McCarthy - Computer Scientist	smw://protont#withinOrganization	Stanford University
UFC	smw://protont#locatedIn	Las Vegas
John McCarthy	smw://protont#hasPosition	John McCarthy - Referee
John McCarthy - Referee	smw://protont#withinOrganization	Los Angeles Police Department

Tabelle B.2.: Erweiterung der KIM-Ontologie

C. Geonames Datensatz

In Tabelle C.1 ist die Zugehörigkeit der in den Texten enthaltenen Ontologieentitäten zu ihren Ontologiekonzepten dargestellt. Die Tabelle zeigt auf, dass die Dokumente überwiegend Entitäten mit den Konzepten „Country“ bzw. „Populated Place“ enthalten. Dies resultiert aus dem Charakter der verwendeten Texte, da es sich bei diesen um Nachrichtentexte allgemeinen Inhalts handelt und somit die Nachrichten vorwiegend Länder- und Städtenamen beinhalten.

KonzeptID	Konzeptname	Dokumente je Konzept	Prozent	Anzahl der Dokumente
2000000334	Hydrography	286	28,54%	1002
2000050356	GeographicFeature	13	1,30%	1002
2000000386	Spot	132	13,17%	1002
2006490772	Continent	122	12,18%	1002
2000000332	Hypsographic	200	19,96%	1002
2000000348	Locality	119	11,88%	1002
2000000330	Undersea	21	2,10%	1002
2000000000	Country	813	81,14%	1002
2000000352	Vegetation	17	1,70%	1002
2000000338	AdministrativeRegion	660	65,87%	1002
2000000336	PopulatedPlace	856	85,43%	1002

Tabelle C.1.: Verteilung der Konzeptzugehörigkeit innerhalb des Testdatensatzes

D. Evaluation KIM Datensatz

Im Folgenden wird näher auf die Resultate des von Nguyen vorgestellten Referenzdatensatzes eingegangen, der in seinem Artikel [163] vorgestellt wird und freundlicherweise dem Autor dieser Arbeit zur Verfügung gestellt wurde. In Abschnitt D.1 werden die Evaluationsergebnisse erörtert, die durch eine Nachimplementierung des von Nguyen et al. vorgestellten Algorithmus erzielt werden. Diese Nachimplementierung kommt zu leicht veränderten Ergebniswerten gegenüber den innerhalb des Artikels [163] präsentierten Ergebniswerten. In Abschnitt D.2 sind die ausführlichen Ergebnisse aufgelistet, die durch die vom Autor dieser Arbeit entwickelten Algorithmus auf dem KIM-Datensatz erzielt werden.

D.1. Ergebnisse der Nachimplementierung des Nguyen Ansatzes

In den Tabellen D.1, D.2 und D.3 sind die Ergebnisse der Nachimplementierung des Nguyen Ansatzes aufgeführt. Diese werden jeweils mit den von ihm in Artikel [163] genannten Ergebnissen verglichen.

Zunächst erfolgt die Erstellung eines Wortvektors für jede Ontologieinstanz und eines Wortvektors für jeden identifizierten Textbereich, der einen Instanzbezeichner beinhaltet. Erstere enthalten die direkte Umgebung der Ontologieentitäten, *d.h.* der Data-Properties und der direkt zugeordneten Object-Properties. In Tabelle D.1 sind die Ergebnisse dargestellt. Auffallend ist, dass die Ergebnisse der Nachimplementierung teilweise starke Abweichungen zu den von Nguyen erzielten Resultaten aufweisen. Diese Abweichungen sind höchstwahrscheinlich Folge der unterschiedlichen Tripel, die als Erweiterung der Daten, die durch die KIM-Ontologie zur Verfügung gestellt wurden, verwendet werden. Wie zuvor bereits erwähnt war es nicht möglich

die von Nguyen im Ansatz [163] verwendeten Tripel zu erhalten. Es wurden daher die in Tabelle B.2 angegebenen Tripel der Ontologie hinzugefügt.

Tabelle D.1 nennt die Ergebnisse ohne Stemming[1] und der in Kapitel 13 vorgestellten globalIE-Variante. Erfolgt die Berücksichtigung des allgemeinen Entitätsbezeichners „Columbia“ anstatt der in den meisten Texten vorkommenden exakten Entitätsbezeichnern, *z.B.* „District of Columbia“[2] oder „Columbia University“[3] resultiert dies in einem Rückgang des F-Measure Wertes von 73, 12 auf 59, 7. Die Unterschiede bezüglich des Resultats von „Georgia“ sind mit großer Wahrscheinlichkeit auf die unterschiedlichen, zusätzlich verwendeten Tripel zurückzuführen. Jedoch kann auch ohne Verwendung dieser Erweiterung möglich höherer Resultatwert als der des Orginalansatzes erzielt werden.[4] Der Leistungseinbruch bei der Disambiguierung von „John McCarthy“ ist auf die Erzeugung der Vektorrepräsentation der Ontologieentität zurückzuführen. Die Entität besitzt zwar eine Object-Property „hasJobPosition“, jedoch erst über eine weitere Object-Property ausgehend vom Wert der JobPosition kann ein Zusammenhang zu einer Organisation identifiziert werden.Somit ist dieser Zusammenhang erst durch eine weiterführende Analyse identifizierbar, *d.h.* diese wird im direkten Kontext der Entität nicht erwähnt und ist somit nicht im Entitätsvektor aufgeführt. Diese Tatsache begründet das schlechte Disambiguierungsergebnis. Durch zusätzliche Tripel kann dieser Missstand behoben werden und dies deutet darauf hin, dass diese von Nguyen bereitgestellt wurden.

In Übereinstimmung mit der von Nguyen erzielten Verbesserung durch die Verwendung von BaseNounPhrases erzielt auch die Nachimplementierung bei Berücksichtigung dieser Noun Chunks bessere F-Measure Werte. Tabelle D.2 stellt diese dar.

Bei der Durchführung eines Wort-Stemmings als Voranalyse wird eine Verbesserung der Ergebnisse für „Columbia“ und „Georgia“ erzielt.

---

1 Mit „Stemming“ bezeichnet man den Vorgang der Überführung von Worten in ihre Grundform bzw. auf ihren Wortstamm. Beispielsweise werden „gehst“, „gehen“ *etc.* auf in Grundform „gehen“ überführt.

2 Entität wird in allen 5 Texten mit „District of Columbia“ genannt.

3 Entität wird in 4 von 5 Texten mit „Columbia University“ erwähnt.

4 Hierbei sei angemerkt, dass auch durch zusätzliche Tripel die Qualität des Ergebnisses gemindert werden kann.

	Columbia	Columbia (nur Columbia)	Georgia (mit künstlichen Tripel)	Georgia (ohne künstlichen Tripel)	John McCarthy
Precision	81,25	71,429	80,556	60	100
Recall	66,667	51,282	40,845	16,901	3,226
F-Measure	73,115	59,701	54,205	26,374	6,249
Von Nguyen genannte Resultate in Artikel [163]					
F-Measure	92,5	-	15,27	-	82,14

Tabelle D.1.: KIM - Nur Benannte Entitäten in Texten (ohne Stemming)

	Columbia	Georgia (mit künstlichen Tripel)	Georgia (ohne künstlichen Tripel)	John McCarthy
Precision	85,714	81,018	61,905	100
Recall	76,923	42,253	18,309	9,677
F-Measure	81,081	55,556	28,261	17,647
Von Nguyen genannte Resultate in Artikel [163]				
F-Measure	92,5	-	16,67	82,14

Tabelle D.2.: KIM - Benannte Entitäten und Noun Chunk (ohne Stemming)

Diese ist darauf zurückzuführen, dass beispielsweise die verschiedenen Flexionsformen der Properties in eine übereinstimmende Grundform überführt wurden.

D.2. Ergebnisse der innerhalb dieser Arbeit entwickelten Verfahren

In Tabelle D.5, D.7 und D.9 sind die Ergebnisse der vom Autor dieser Arbeit entwickelten Verfahrensvarianten aufgeführt. Die Tabelle D.5 berücksichtigt kein Kantenmaß, während die Tabelle D.7 das semantische Kantenmaß und die Tabelle D.9 das heuristische Kantenmaß verwenden.

Die Tabellen sind wie folgt strukturiert:

- Explorationsmethode: 0 - Unidirektional (siehe Kapitel 8); 1 - Bidirektional (siehe Kapitel 9)
- Fokussiert auf Aktivierungswerte: 0 - Nein; 1 - Ja
  Eine Fokussierung bedeutet hierbei, dass die Exploration aktivierungsbasiert vorgenommen wird. Falls nicht, ist der Distanzwert entscheidend.
- Bestärkendes Lernen: 0 - Nein; 1 - Ja
  Verwendung des Bestärkenden Lernens (siehe Kapitel 11)
- One Connection: Die Exploration wird nur über Ontologieelemente fortgeführt, die nur eine ausgehende Kante besitzen bzw. über die mit den geringsten ausgehenden Kanten.

Die besten Werte für „Columbia“ und „John McCarthy“ werden unter Berücksichtigung des semantischen Kantenmaßes erzielt. Beiden zu Eigen ist die Verwendung der bidirektionalen Exploration, des Bestärkenden Lernens und der auf die Distanz fokussierten Vorgehensweise. Das beste Ergebnis für „Georgia“ wird mittels unidirektionaler Exploration unter dem Ausschluss des Bestärkenden Lernens, distanzfokussierter Exploration und der Berücksichtigung des heuristischen Kantenmaßes erzielt.

	Columbia	Georgia (mit künstlichen Tripel)	Georgia (ohne künstlichen Tripel)	John McCarthy
Precision	93,939	75,758	58,824	100
Recall	79,487	35,211	14,085	3,226
F-Measure	86,111	48,084	22,727	6,25

Tabelle D.3.: KIM - Benannte Entitäten (mit Stemming)

Explorationsmethode	Fokussiert auf Aktivierungswerte	Bestärkendes Lernen	One connection	Columbia	Georgia	John McCarthy	Durchschnitt
0	1	0	1	51,282	84,038	81,183	72,167
0	1	0	0	51,282	84,038	81,183	72,167
0	1	1	1	48,718	81,925	81,720	70,788
0	1	1	0	46,154	81,925	81,720	69,933
0	0	0	1	51,282	84,038	81,183	72,167
1	1	1	0	43,590	82,629	81,720	69,313
0	0	0	0	51,282	84,038	81,183	72,167
0	0	1	1	46,154	84,038	81,720	70,637
0	0	1	0	48,718	83,333	81,720	71,257
1	1	0	1	48,718	78,404	75,806	67,643
1	1	0	0	48,718	78,404	75,806	67,643
1	1	1	1	43,590	82,629	81,720	69,313
1	0	0	1	48,718	76,995	76,882	67,532
1	0	0	0	48,718	76,995	76,882	67,532
1	0	1	1	51,282	81,925	81,720	71,642
1	0	1	0	51,282	81,925	81,720	71,642

Tabelle D.4.: KIM - Resultate ohne Kantenmaße (Teil 1)

Explorationsmethode	Fokussiert auf Aktivierungswerte	Bestärkendes Lernen	One connection	Columbia	Georgia	John McCarthy	Durchschnitt
lokale Kohärenz (Kontextgröße 10)							
0	1	0	1	48,718	79,812	76,882	68,471
0	1	0	0	48,718	79,812	76,882	68,471
0	1	1	1	51,282	79,812	73,656	68,250
0	1	1	0	51,282	79,812	73,656	68,250
0	0	0	1	48,718	84,038	76,882	69,879
0	0	0	0	48,718	84,038	76,882	69,879
0	0	1	1	51,282	84,038	73,656	69,659
0	0	1	0	51,282	84,038	73,656	69,659
1	1	0	1	46,154	79,812	79,032	68,333
1	1	0	0	46,154	79,812	79,032	68,333
1	1	1	1	48,718	76,995	79,032	68,249
1	1	1	0	48,718	76,995	79,032	68,249
1	0	0	1	46,154	78,404	76,882	67,146
1	0	0	0	48,718	84,038	76,882	69,879
1	0	1	1	48,718	75,587	76,882	67,062
1	0	1	0	48,718	75,587	76,882	67,062

Tabelle D.5.: KIM - Resultate ohne Kantenmaße (Teil 2)

Explorationsmethode	Fokussiert auf Aktivierungswerte	Bestärkendes Lernen	One connection	Columbia	Georgia	John McCarthy	Durchschnitt
0	1	0	1	58,974	74,178	83,333	72,162
0	1	0	0	58,974	74,178	83,333	72,162
0	1	1	1	58,974	74,178	80,108	71,087
0	1	1	0	58,974	74,178	80,108	71,087
0	0	0	1	58,974	81,221	83,333	74,509
0	0	0	0	58,974	81,221	83,333	74,509
0	0	1	1	58,974	81,221	76,882	72,359
0	0	1	0	58,974	81,221	76,882	72,359
1	1	0	1	58,974	74,178	83,333	72,162
1	1	0	0	58,974	74,178	83,333	72,162
1	1	1	1	58,974	75,587	83,333	72,632
1	1	1	0	58,974	75,587	83,333	72,632
1	0	0	1	56,757	77,451	80,864	71,691
1	0	0	0	56,757	77,451	80,864	71,691
1	0	1	1	59,459	77,114	87,821	74,798
1	0	1	0	59,459	77,114	87,821	74,798

Tabelle D.6.: KIM - Resultate mit semantischem Kantenmaß (Teil 1)

Explorationsmethode	Fokussiert auf Aktivierungswerte	Bestärkendes Lernen	One connection	Columbia	Georgia	John McCarthy	Durchschnitt
Kontextgröße 10							
0	1	0	1	58,974	72,770	83,333	71,693
0	1	0	0	58,974	72,770	83,333	71,693
0	1	1	1	56,410	72,770	83,333	70,838
0	1	1	0	56,410	72,770	83,333	70,838
0	0	0	1	58,974	79,812	83,333	74,040
0	0	0	0	58,974	79,812	83,333	74,040
0	0	1	1	53,846	79,812	83,333	72,331
0	0	1	0	53,846	79,812	83,333	72,331
1	1	0	1	58,974	72,770	83,333	71,693
1	1	0	0	58,974	72,770	83,333	71,693
1	1	1	1	53,846	74,178	83,333	70,453
1	1	1	0	53,846	74,178	83,333	70,453
1	0	0	1	56,757	75,980	80,864	71,200
1	0	0	0	56,757	75,980	80,864	71,200
1	0	1	1	54,054	77,114	80,864	70,678
1	0	1	0	54,054	77,114	80,864	70,678

Tabelle D.7.: KIM - Resultate mit semantischem Kantenmaß (Teil 2)

Explorationsmethode	Fokussiert auf Aktivierungswerte	Bestärkendes Lernen	One connection	Columbia	Georgia	John McCarthy	Durchschnitt
0	1	0	1	51,282	81,221	54,301	62,268
0	1	0	0	51,282	81,221	54,301	62,268
0	1	1	1	51,282	76,995	73,656	67,311
0	1	1	0	51,282	77,949	71,605	66,945
0	0	0	1	51,282	86,854	73,656	70,597
0	0	0	0	51,282	86,854	73,656	70,597
0	0	1	1	51,282	82,629	73,656	69,189
0	0	1	0	51,282	82,629	73,656	69,189
1	1	0	1	46,154	79,812	54,301	60,089
1	1	0	0	46,154	79,812	54,301	60,089
1	1	1	1	48,718	75,587	76,882	67,062
1	1	1	0	48,718	75,587	76,882	67,062
1	0	0	1	46,154	78,404	76,282	66,947
1	0	0	0	46,154	78,404	76,282	66,947
1	0	1	1	50,000	75,587	83,333	69,640
1	0	1	0	50,000	75,587	83,333	69,640

Tabelle D.8.: KIM - Resultate mit heuristischem Kantenmaß (Teil 1)

Explorationsmethode	Fokussiert auf Aktivierungswerte	Bestärkendes Lernen	One connection	Columbia	Georgia	John McCarthy	Durchschnitt
Kontextgröße 10							
0	1	0	1	46,154	78,404	54,301	59,620
0	1	0	0	46,154	78,404	54,301	59,620
0	1	1	1	48,718	78,404	60,753	62,625
0	1	1	0	48,718	78,404	60,753	62,625
0	0	0	1	46,154	85,446	73,656	68,419
0	0	0	0	46,154	85,446	73,656	68,419
0	0	1	1	48,718	84,038	76,882	69,879
0	0	1	0	48,718	84,038	76,882	69,879
1	1	0	1	41,026	77,778	54,301	57,701
1	1	0	0	41,026	77,778	54,301	57,701
1	1	1	1	43,590	76,667	60,753	60,336
1	1	1	0	43,590	76,667	60,753	60,336
1	0	0	1	41,026	76,995	76,282	64,768
1	0	0	0	41,026	76,995	76,282	64,768
1	0	1	1	43,590	75,238	80,128	66,319
1	0	1	0	43,590	75,238	80,128	66,319

Tabelle D.9.: KIM - Resultate mit heuristischem Kantenmaß (Teil 2)

Für alle Varianten des Kantenmaßes (ohne, semantisches Kantenmaß und heuristisches Kantenmaß) wird unter Verwendung der unidirektionalen Exploration und des Bestärkenden Lernens für „Georgia“ das beste Ergebnis erzielt. Für „John McCarthy“ ist trifft dies für die bidirektionale Exploration und Bstärkenden Lernen zu. Für „Columbia“ lässt sich kein Verhalten bestimmen, das über die verschiedenen Maße hinweg Gültigkeit besitzt.

Die Berücksichtigung lokaler Kontexte resultiert in schlechteren Ergebnissen im Gegensatz zur Verwendung eines Kontextes für den gesamten Text. Dieser Effekt trat unabhängig von der Verwendung des Bestärkenden Lernens auf.

Die aus der vorhergehenden Analyse hervorgehenden vorteilhafteren Ergebnisse für distanzfokussierte Exploration lässt sich auf die Struktur der Ontologie sowie auf die Tatsache zurückführen, dass es sich bei „Georgia“ und „Columbia“ um geographische Entitäten handelt. In Ontologien werden geographische Entitäten, die miteinander in Beziehung stehen oftmals direkt, *d.h.* über geringe Distanz, in Relation zueinander gesetzt. Aufgrund dessen ist die Berücksichtigung kurzer Wege vorteilhaft für die Disambiguierung. Basierend auf der hohen Anzahl von Dokumenten ähnlichen Inhalts stellt sich Bestärkendes Lernen ebenfalls als vorteilhaft heraus, da eine erfolgreiche Identifikation von Entitäten in nachfolgenden Disambiguierungen wiederverwendet werden können.

In Tabelle D.10 sind die Ergebnisse angegeben, die der Algorithmus des Autor dieser Arbeit bei exakter Berücksichtigung der im Text enthaltenen Bezeichner erreicht. Demzufolge werden in Texten, die *z.B.* „District of Columbia“ enthalten, der vollständige Bezeichner und nicht – wie im Standardfall – eine Beschränkung auf „Columbia“ durchgeführt. Letztere ist nötig, um die Ambiguität der Bezeichner zu erhalten.

Der Algorithmus erzielt eine Verbesserung im Testfall „Columbia“. Hier war es möglich 89.10% F-Measure zu erzielen. Ohne die Berücksichtigung des exakten Bezeichner konnten zuvor lediglich 59, 5% erreicht werden. Für „Georgia“ und „John McCarthy“ können indes beinahe dieselben Ergebnisse erzielt werden. Die geringfügige Verschlechterung lässt sich auf die veränderten Teilgraphen zurückführen, die durch den Wechsel der Bearbeitungsmethodik berechnet wurden.

Explorationsmethode	Fokussiert auf Aktivierungswerte	Bestärkendes Lernen	One connection	Kantenmaß	Columbia	Georgia	John McCarthy
0	1	0	1	11	11	11	11
0	1	0	0	semantisch	89.10	69.01	81.72
0	0	0	1				
0	0	1	0	ohne	78.21	86.151	78.5
1	0	1	1				
1	0	1	0	heuristisch	85.89	81.92	81.72

Tabelle D.10.: KIM - Resultate mit exakten Bezeichnern

E. Evaluation Geonames Datensatz (distanzbasiert)

Innerhalb dieses Kapitels werden die Ergebnisse der in dieser Arbeit entwickelten Ansätze unter Anwendung der distanzfokussierte Exploration (siehe Algorithmus 6) vorgestellt.

Bevor jedoch auf die einzelnen Ergebnisse eingegangen wird, wird der Unterschied zwischen der distanzfokussierten und der aktivierungsfokussierten Explorationsmethodik nochmals hervorgehoben. Bei der distanzfokussierten Exploration werden in der Liste der besten Vorgängerknoten $P_{u,l}$ des Knotens u für einen Bezeichner l diejenigen gespeichert, welche die geringste Distanz zum Knoten u aufweisen. Bei einer aktivierungsfokussierten Exploration handelt es sich hingegen um die Knoten mit der höchsten Aktivierung. Eine Berücksichtigung des Aktivierungswertes findet im distanzfokussierten Ansatz erst gegen Ende des Algorithmus statt. Dort wird aus der Liste der Vorgängerknoten nur derjenige mit der höchsten Aktivierung und mit der geringsten Distanz verwendet. Bei der aktivierungsfokussierten Exploration hingegen ist die Distanz zwar einer der Werte der Aktivierungsbestimmung, jedoch ist die Bindung zwischen geringster Distanz und höchster Aktivierung nicht zwangsläufig gegeben.

Beim direkten Vergleich ist es abhängig von der Ontologie und von der Formel, die den Aktivierungswert berechnet, welche Art der Explorationsmethode die vorteilhafteren Ergebnisse erzielt. Die Resultate sind in Tabelle E.1 aufgelistet.

Explorationsmethode	Bestärkendes Lernen	Kontextgröße 10	Recall	Precision	F-Measure
Fokussiert auf Distanzwerte – ohne Ranking					
0	0	0	79,799	58,661	67,616
1	0	0	80,491	59,495	68,418
0	0	1	80,067	58,907	67,876
1	0	1	79,755	59,655	68,256
0	1	0	78,871	64,373	70,888
1	1	0	76,365	65,503	70,518
0	1	1	78,765	60,655	68,534
1	1	1	77,496	62,487	69,187
Fokussiert auf Distanzwerte – semantisches Maß					
0	0	0	74,389	69,679	71,957
1	0	0	77,468	71,063	74,127
0	0	1	75,112	70,326	72,640
1	0	1	73,488	70,099	71,754
0	1	0	75,296	72,144	73,686
1	1	0	73,447	71,341	72,379
0	1	1	74,602	70,767	72,634
1	1	1	73,915	71,312	72,560
Fokussiert auf Distanzwerte – heuristisches Maß					
0	0	0	76,311	69,947	72,990
1	0	0	76,863	72,866	74,811
0	0	1	76,866	70,587	73,593
1	0	1	76,659	72,767	74,662
0	1	0	75,271	70,759	72,945
1	1	0	75,869	73,696	74,767
0	1	1	75,626	70,581	73,016
1	1	1	75,849	73,012	74,403

Tabelle E.1.: Geonames - Resultate des auf Distanz fokussierten Ansatzes

Bei der Analyse der erzielten Resultate stellt sich heraus, dass im Standardfall, *d.h.* der Bearbeitung ohne lokale Kohärenz und Bestärkenden Lernens, die bidirektionale Exploration der unidirektionalen überlegen ist. Dieses Phänomen ist unabhängig von dem verwendeten Kantenmaß. Auch unter Berücksichtigung der lokalen Kohärenz erzielt die Bidirektionale Exploration für den Verzicht auf ein Kantenmaß bzw. dem heuristischen Kantenmaß die besseren Resultate. Bei der Verwendung des semantischen Kantenmaßes tritt dieser Effekt nicht auf. Jedoch ist die Abweichung im Resultat minimal. Der Einsatz bestärkenden Lernens ermöglicht eine Steigerung der Resultatgüte und erzielt das beste Ergebnis unter Anwendung der bidirektionalen Exploration und des heuristischen Kantenmaß.

Schlussfolgerung: Die distanzfokussierte Evaluation ermöglicht die Erzielung von besseren Ergebnissen gegenüber der aktivierungsfokussierten Evaluation (siehe Abschnitt 13.3.2). Dies ist auf die Domäne der Ontologie zurückzuführen. Die Geonames-Ontologie als Wissensbasis geographischer Entitäten enthält eine enge Bindung zwischen miteinander in Beziehung stehenden Entitäten. Diese enge Bindung führt innerhalb der Graphdarstellung zu kurzen Entfernungen. Die verwendeten Texte enthalten weitgehend Entitäten, für die diese enge Bindung zutrifft und daher ermöglicht die distanzfokussierte Exploration ein vorteilhafteres Ergebnis. Es muss an dieser Stelle jedoch darauf hingewiesen werden, dass die Auswahl anhand der höchsten Aktivierungswerte und insofern der Bestimmung der Aktivierungswerte parallel zur distanzfokussierten Exploration die hohen Werte für Precision und Recall ermöglicht. Bei reiner Anwendung distanzfokussierter Explorationsmethodik *ohne* die zuletzt vorgenommene Auswahl anhand des Aktivierungswertes wäre der Wert für Precision wesentlich geringer und würde kein verwendbares Ergebnis ermöglichen.

Literaturverzeichnis

[1] B. Aditya, G. Bhalotia, S. Chakrabarti, A. Hulgeri, C. Nakhe, Parag, and S. Sudarshan. BANKS: Browsing and Keyword Searching in Relational Databases. In *VLDB, Proceedings 28th International Conference on Very Large Data Bases*, pages 1083–1086, Hong Kong, China, 2002. Morgan Kaufmann.

[2] E. Agirre and G. Rigau. Word Sense Disambiguation Using Conceptual Density. In *Proceedings 16th Conference on Computational Linguistics*, pages 16–22, Copenhagen, Denmark, 1996. Association for Computational Linguistics.

[3] R. Agrawal and R. Srikant. Mining Sequential Patterns. In P. S. Yu and A. S. P. Chen, editors, *Proceedings of the 11th International Conference on Data Engineering*, pages 3–14, Taipei, Tawan, 1995. IEEE Computer Society.

[4] T. E. Ahlswede and D. Lotand. Word Sense Disambiguation by Human Subjects: Computational and Psycholinguistic Applications. In *Proceedings of a Workshop Sponsored by the Special Interest Group on the Lexicon of the Association for Computational Linguistics*, pages 1–9, Ohio State University, Columbus, 1993.

[5] B. Aleman-Meza, C. Halaschek-Wiener, I. B. Arpinar, and A. P. Sheth. Context-aware semantic association ranking. In I. F. Cruz, V. Kashyap, S. Decker, and R. Eckstein, editors, *Proceedings of SWDB'03, The first International Workshop on Semantic Web and Databases*, pages 33–50, Berlin, Germany, 2003.

[6] S. S. Alhir. *Guide to Applying the UML*. Springer-Verlag, 2002.

[7] G. Altmann and M. Steedman. Interaction with Context during Human Sentence Processing. *Cognition*, 30(3):191 – 238, 1988.

[8] J. R. Anderson. A Spreading Activation Theory of Memory. *Journal of Verbal Learning and Verbal Behavior*, 22:261–295, 1983.

[9] J. R. Anderson. *Kognitive Psychologie*. Spektrum, 2001.

[10] K. Anyanwu, A. Maduko, and A. Sheth. SemRank: Ranking Complex Relationship Search Results on the Semantic Web. In *Proceedings of the 14th International Conference on World Wide Web*, WWW '05, pages 117–127, Chiba, Japan, 2005.

[11] K. Anyanwu and A. Sheth. P-queries: Enabling querying for semantic associations on the semantic web. In *Proceedings of the 12th International Conference on World Wide Web*, pages 690 – 699, Budapest, Hungary, 2003. ACM Press New York, NY, USA.

[12] J. D. Apresjan. Regular Polysemy. *Linguistics*, 142:5–32, 1973.

[13] J. D. Apresjan. Leksiceskaja semantika. Sinonimiceskie sredstva jazyka. *Moskva*, 1974.

[14] J. D. Apresjan. *Die semantische Sprache als Mittel der Erklärung lexikalischer Bedeutungen*, pages 22–48. Niemeyer Max Verlag GmbH, Tübingen, 1976.

[15] D. Aswath, S. T. Ahmed, J. D'cunha, and H. Davulcu. Boosting Item Keyword Search with Spreading Activation. *Web Intelligence, IEEE / WIC / ACM International Conference on*, 0:704–707, 2005.

[16] A. Azevedo and M. F. Santos. KDD, SEMMA and CRISP-DM: A parallel Overview. In *IADIS European Conference Data Mining*, pages 182–185, Algarve, Portugal, 2008.

[17] M. Banek, B. Vrdoljak, and A. M. Tjoa. Word Sense Disambiguation as the Primary Step of Ontology Integration. In *Proceedings International Conference on Database and Expert Systems Applications*, Turin, Italy, 2008. Springer-Verlag.

[18] H. Berger. *Activation on the Move: Adaptive Information Retrieval via Spreading Activation*. PhD thesis, Technischen Universität Wien, June 2003.

[19] R. Berger. *Warum der Mensch spricht - Eine Naturgeschichte der Sprache*. Eichborn Verlag, 2008.

[20] S. Berndler. Klassifikation der Namen. In A. Brendler and S. Brendler, editors, *Namenarten und ihre Erforschung: Ein Lehrbuch für das Studium der Onomastik, anlässlich des 70. Geburtstags von Karlheinz Hengst*. baar Verlag, 2004.

[21] T. Berners-Lee, R. Fielding, and L. Masinter. RFC2396 Uniform Resource Identifier (URI): Generic Syntax. `http://tools.ietf.org/html/rfc3986`, 1998. [Online; letzter Zugriff am 12.09.2011].

[22] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web: a new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American*, H.284:34–43, may 2001.

[23] M. R. Berthold, U. Brandes, T. Kötter, M. Mader, U. Nagel, and K. Thiel. Pure Spreading Activation is Pointless. In *Proceeding of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 1915–1918, Hong Kong, China, 2009.

[24] G. Bhalotia, A. Hulgeri, C. Nakhe, S. Chakrabarti, and S. Sudarshan. Keyword Searching and Browsing in Databases using BANKS. In *ICDE 2002, Proceedings International Conference on Data Engineering*, San Jose, California, USA, 2002. IEEE Computer Society.

[25] E. Bick. A Named Entity Recognizer for Danish. In *Conference on Language Resources and Evaluation*, Lisbon, Portugal, 2004.

[26] M. Bierwisch. Semantische und konzeptuelle Repräsentation lexikalischer Einheiten. *Untersuchungen zur Semantik [=studia grammatica]*, 22:61–101, 1983.

[27] M. Bierwisch and E. Lang. Semantik der Graduierung. In M. Bierwisch and E. Lang, editors, *Grammatikalische und konzeptuelle Aspekte von Dimensionsadjektiven*, pages 91–283. Akademie-Verlag, Berlin, 1987.

[28] R. Binnik. Ambiguity and Vagueness. *Papers form the Sixth Regional Meeting of the Chicago Linguistic Society, Chicago*, pages 147–153, 1970.

[29] B. Bishop, A. Kiryakov, D. Ognyanoff, I. Peikov, Z. Tashev, and R. Velkov. Owlim : A family of scalable semantic repositories. *Education*, 2(1):33–42, 2011.

[30] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1st ed. 2006. corr. 2nd printing edition, 2007.

[31] C. Bizer, T. Heath, et al. Interlinking Open Data on the Web. www.wiwiss.fu-berlin.de/bizer/pub/LinkingOpenData.pdf. Stand 12.5.2009.

[32] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. DBpedia - A crystallization point for the Web of Data. *Web Semantics*, 7:154–165, September 2009.

[33] P. F. P.-S. Boris Motik and B. Cuenca Grau. OWL 2 Web Ontology Language: Direct Semantics. World Wide Web Consortium (W3C) Recommendation, 2009.

[34] W. N. Borst. *Construction of Engineering Ontologies for Knowledge Sharing and Reuse*. PhD thesis, Universiteit Twente, Enschede, September 1997.

[35] R. Brachman and H. Levesque. *Knowledge Representation and Reasoning (The Morgan Kaufmann Series in Artificial Intelligence)*. Morgan Kaufmann, May 2004.

[36] R. J. Brachman and T. Anand. The Process of Knowledge Discovery in Databases. In *Advances in Knowledge Discovery and Data Mining*, pages 37–57. AAAI Press, 1996.

[37] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, and F. Yergeau. Extensible Markup Language (XML) 1.0 4th ed. (W3C Recommendation 16 August 2006). http://www.w3.org/TR/2006/REC-xml-20060816, Aug. 2006.

[38] D. Brickley and R. Guha. RDF Vocabulary Description Language 1.0: RDF Schema. `http://www.w3.org/TR/rdf-schema/`, 2004. [Online; letzter Zugriff am 12.09.2011].

[39] D. Buscaldi and P. Rosso. Geo-WordNet: Automatic Georeferencing of WordNet. In K. C. B. M. J. M. J. O. S. P. D. T. Nicoletta Calzolari (Conference Chair), editor, *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may 2008. European Language Resources Association (ELRA).

[40] C. M. Busler. Über elliptische Konstruktionen im gesprochenen Deutsch. Master's thesis, Universität München, 1998.

[41] H. Bußmann. *Lexikon der Sprachwissenschaft*. Kröner Verlag, Stuttgart, 1990.

[42] K.-U. Carstensen, C. Ebert, C. Ebert, S. Jekat, R. Klabunde, and H. Langer, editors. *Computerlinguistik und Sprachtechnologie: Eine Einführung*. Spektrum, Heidelberg, 3. edition, 2009.

[43] B. Chandrasekaran, J. R. Josephson, and V. R. Benjamins. What Are Ontologies, and Why Do We Need Them? *IEEE Intelligent Systems*, 14:20–26, 1999.

[44] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, and R. Wirth. CRISP-DM 1.0 Step-by-step Data Mining Guide. Technical report, The CRISP-DM consortium, 2000.

[45] P. P. Chen. The Entity-Relationship Model - Toward a Unified View of Data. *ACM Trans. Database Syst.*, 1(1):9–36, 1976.

[46] S. Coates-Stephens. The Analysis and Acquisition of Proper Names for the Understanding of Free Text. In *Computers and Humanities*, volume 26, pages 441–456, San Francisco, USA, 1992. Morgen Kaufmann Publishers.

[47] P. R. Cohen and R. Kjeldsen. Information Retrieval by Constrained Spreading Activation in Semantic Networks. *Inf. Process. Manage.*, 23:255–268, July 1987.

[48] W. W. Cohen, P. Ravikumar, and S. E. Fienberg. A Comparison of String Metrics for Matching Names and Records. In *The Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Workshop on Data Cleaning and Object Consolidation*, 2003.

[49] A. M. Collins and E. F. Loftus. A Spreading-Activation Theory of Semantic Processing. *Psychological Review*, 82(6):407–428, 1975.

[50] A. M. Collins and M. Quillian. Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, 8(2):240–247, April 1969.

[51] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, second edition, 2002.

[52] R. Courant and H. Robbins. *What is Mathematics?* Oxford University Press, NY, 1941.

[53] F. Crestani. Application of Spreading Activation Techniques in Information Retrieval. *Artificial Intelligence Review*, 11:453–482, 1997.

[54] F. Crestani, M. Laimas, and J. C. van Rijsbergen. *Information Retrieval: Uncertainty and Logics*. Kluwer Academic Publishers, 1998.

[55] S. Cucerzan. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. In *CoNLL 2007, 11th Conference on Computational Natural Language Learning*, pages 708–716, Portland, Oregon, USA, 2007. ACL.

[56] S. Dar, G. Entin, S. Geva, and E. Palmon. DTL's DataSpot: Database Exploration as Easy as Browsing the Web. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 590–592, Seattle, Washington, USA, 1998.

[57] S. Dar, G. Entin, S. Geva, and E. Palmon. DTL's DataSpot: Database Exploration Using Plain Language. In *VLDB'98, Proceedings of the 24th International Conference on Very Large Data Bases*, pages 645–649, New York City, USA, 1998.

[58] G. Das, R. Fleischer, L. Gasieniec, D. Gunopulos, and J. Kärkkäinen. Episode Matching. In *Proceedings of the 8th Annual Symposium on Combinatorial Pattern Matching*, CPM '97, pages 12–27, Arhus, Denmark, 1997. Springer-Verlag.

[59] T. Davenport and L. Prusak. *Working Knowledge: How Organizations Manage What They Know*. Harvard Business School Press, Cambridge, MA, 1998.

[60] P. Deane. Polysemy and Cognition. *Lingua*, 75(4):325–361, 1988.

[61] J. Diedrich. Spreading Activation and Connectionist Models for Natural Language Processing. Technical report, International Computer Science Institute, 1990.

[62] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[63] J. Dölling. Grundlagen der Bedeutungsvarianten. `http://www.uni-leipzig.de/~doelling/veranstaltungen/bedeutvariat1.pdf`, 2009. [Online; letzter Zugriff am 12.09.2011].

[64] G. Drosdowski, editor. *Duden Grammatik der deutschen Gegenwartssprache*. Bibliographisches Insitut, Manheim, Wien, Zürich, 6 edition, 1998. Band 4 von: Der Duden in 10 Bänden Das Standardwerk zur deutschen Sprache.

[65] D. Z. Du, F. K. Hwang, and S. C. Chao. Steiner minimal tree for points on a circle. *Proceedings Amer. Math. Soc.*, 95:613–618, 1985.

[66] J. L. Elman. Finding structure in time. *COGNITIVE SCIENCE*, 14(2):179–211, 1990.

[67] R. Engels. *Component-Based User Guidance in Knowledge Discovery and Data Mining.*, volume 211 of *DISKI*. Infix, 1999.

[68] L. Ertz, M. Steinbach, and V. Kumar. Finding Topics in Collections of Documents: A Shared Nearest Neighbor Approach. In *In Proceedings of Text Mine'01, First SIAM International Conference on Data Mining*, Chicago, USA, 2001.

[69] O. Etzioni, M. Cafarella, D. Downey, A.-M. Popescu, T. Shaked, S. Soderland, D. S. Weld, and A. Yates. Unsupervised Named-entity Extraction from the Web: an Experimental Study. *Artiicial Intelligence*, 165(1):91–134, 2005.

[70] Eurostat. Eurostat Jahrbuch der Regionen. Technical report, Eurostat, 2010. Seite 127, zugreifbar via `http://epp.eurostat.ec.europa.eu/cache/ITY_OFFPUB/KS-HA-10-001/DE/KS-HA-10-001-DE.PDF`, Letzter Zugriff am 12.09.2011.

[71] J. Euzenat and P. Shvaiko. *Ontology Matching*. Springer, Berlin, Heidelberg, 2007.

[72] R. Evans. A Framework for Named Entity Recognition in the Open Domain. In *In Proceedings of the Recent Advances in Natural Language Processing (RANLP)*, pages 137–144, Borovets, Bulgaria, 2003.

[73] G. Fauconnier. Mental Spaces. Technical report, Cambridge, MA: MIT Press, 1985.

[74] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, 17(3):37–54, 1996.

[75] L. Feigenbaum, I. Herman, T. Hongsermeier, E. Neumann, and S. Stephens. The Sematic Web in Action. *Scientific American*, 2007.

[76] C. Fellbaum, editor. *WordNet an Electronic Lexical Database*. The MIT Press, 1998.

[77] D. Fensel and R. M. Bürkner. *Ontologies: A silver bullet for knowledge management and electronic commerce*. Springer-Verlag New York, Inc., New York, NY, USA, 2001.

[78] N. Fernández, D. Fuentes, L. Sánchez, and J. A. Fisteus. The NEWS ontology: Design and applications. *Expert Syst. Appl.*, 37:8694–8704, 2010.

[79] P. Ferragina and U. Scaiella. TAGME: On-the-fly Annotation of Short Text Fragments (by Wikipedia Entities). In J. Huang, N. Koudas, G. J. F. Jones, X. Wu, K. Collins-Thompson, and A. An, editors, *CIKM*, pages 1625–1628. ACM, 2010.

[80] O. Ferret. Discovering Word Senses from a Network of Lexical Cooccurrences. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Geneva, Switzerland, 2004.

[81] J. Firth. A synopsis of linguistic theory 1930-1955. *Studies in linguistic analysis*, pages 1–32, 1957.

[82] M. Fleischman. Automated Subcategorization of Named Entities. In *Conference of the European Chapter of Association for Computational Linguistic*, Toulouse, France, 2001.

[83] L. Floridi, editor. *Blackwell Guide to Philosphy of Computing and Information*. Wiley-Blackwell, 2003.

[84] R. J. Fogelin and T. Honrich, editors. *Wittgenstein (The Arguments of the Philosophers)*. Routledge, second edition, 1995.

[85] N. Fries. *Ambiguität und Vagheit. Einführung und kommentierte Bibliographie*. Niemeyer, Tübingen, 1980.

[86] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais. The Vocabulary Problem in Human-System Communication. *COMMUNICATIONS OF THE ACM*, 30(11):964–971, 1987.

[87] W. A. Gale, K. W. Church, and D. Yarowsky. One Sense per Discourse. In *Proceedings of the Workshop on Speech and Natural Language*, HLT '91, pages 233–237, Harriman, New York, 1992.

[88] N. F. García, J. M. B. del Toro, L. Sánchez, and A. Bernardi. IdentityRank: Named Entity Disambiguation in the Context of the NEWS Project. In *Proceedings European Semantic Web Conference*, 2007.

[89] N. F. García, J. M. B. del Toro, L. Sánchez, and V. L. Centeno. Semantic Annotation of Web Resources Using IdentityRank and Wikipedia. In *Proceedings of the 5th Atlantic Web Intelligence Conference*, Fontainbleau, France, 2007.

[90] D. Geeraerts. *Diachronic prototype semantics: a contribution to historical lexicology*. Oxford studies in lexicography and lexicology. Oxford University Press, 1997.

[91] R. Gibbs. Why many concepts are metaphorical. *Cognition*, 61(3):309–319, 1996.

[92] M. Graves, A. Constabaris, and D. Brickley. FOAF: Connecting People on the Semantic Web. *Cataloging Classification Quarterly*, pages 191–202(12), 2007.

[93] H. P. Grice. Logic and Conversation. In P. Cole and J. Morgan, editors, *Syntax and Semantics*, volume 3, pages 41–58. Academic Press, New York, 1975.

[94] B. Griebel. *Semantische Beschreibung metaphorischer und metonymischer Bedeutungsbeziehungen zwischen Sememen polysemer Substantivlexeme*. PhD thesis, Universität Greifswald, 1984.

[95] R. Grishman and B. Sundheim. Message Understanding Conference - 6: A Brief History. In *Proceedings International Conference on Computational Linguistics*, Copenhagen, Denmark, 1996.

[96] T. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[97] T. R. Gruber. Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Deventer, The Netherlands, 1993. Kluwer Academic Publishers.

[98] N. Guarino. Formal Ontology, Conceptual Analysis and Knowledge Representation. *INTERNATIONAL JOURNAL OF HUMAN AND COMPUTER STUDIES*, 43:625–640, 1995.

[99] N. Guarino. Understanding, Building and Using Ontologies. A commentary to 'Using Explicit Ontologies in KBS Development'. *International Journal of Human and Computer Studies*, 46:293–310, 1997.

[100] N. Guarino. Formal Ontology and Information Systems. In *FOIS'98, Formal Ontology in Information Systems*, pages 3–15, Saarbrücken, Germany, 1998. IOS Press.

[101] H. Guo, J. Jiang, G. Hu, and T. Zhang. Chinese Named Entity Recognition Based on Multilevel Linguistic Features. In *IJCNLP, International Joint Conference on Natural Language Processing*, pages 90–99, Hinan Island, China, 2004.

[102] L. Hakulinen. Über polysemie. *Acta Linguistica*, 10(24):157–165, 1974.

[103] D. J. Hand, P. Smyth, and H. Mannila. *Principles of Data Mining*. MIT Press, Cambridge, MA, USA, 2001.

[104] E. Hansack. Das Wesen des Namens. In A. Brendler and S. Brendler, editors, *Namenarten und Ihre Erforschung: Ein Lehrbuch für das Studium der Onomastik, anlässlich des 70. Geburtstags von Karlheinz Hengst*, pages 51–69. baar Verlag, 2004.

[105] M. M. Hasan. A Spreading Activation Framework for Ontology-Enhanced Adaptive Information Access within Organisations. In L. van Elst, V. Dignum, and A. Abecker, editors, *AMKM 2003,*

Agent Mediated Knowledge Management, International Symposium, volume 2926, Stanford, California, USA, 2003. Springer.

[106] J. Hassell, B. Aleman-Meza, and I. B. Arpinar. Ontology-Driven Automatic Entity Disambiguation in Unstructured Text. In *Proceedings Intenational Semantic Web Conference*, Athens, Georgia, 2006.

[107] R. R. Hausser. *Grundlagen der Computerlinguistik: Mensch-Maschine-Kommunikation in natürlicher Sprache*. Springer-Verlag, Berlin, Heidelberg, Juli 2000.

[108] J. Hayes and C. Gutiérrez. Bipartite Graphs as Intermediate Model for RDF. In *International Semantic Web Conference*, pages 47–61, Hiroshima, Japan, 2004.

[109] H. He, H. Wang, J. Yang, and P. S. Yu. BLINKS: Ranked Keyword Searches on Graphs. In *SIGMOD 1998, Proceedings ACM SIGMOD International Conference on Management of Data*, New York, NY, USA, 2007. ACM Press.

[110] J. A. Hendler. *Integrating marker-passing and problem-solving: a spreading-activation approach to improved choice in planning*. PhD thesis, Brown University, Providence, RI, USA, 1986. UMI order no. GAX86-17575.

[111] J. A. Hendler. Marker-passing and Microfeatures. In *Proceedings of the 10th International Joint Conference on Artificial Intelligence - Volume 1*, pages 151–154, Milan, Italy, 1987. Morgan Kaufmann Publishers Inc.

[112] I. Herman. W3C Semantic Web Activity. `http://www.w3.org/2001/sw/`, 2010. [Online; letzter Zugriff am 12.09.2011].

[113] S. G. Holding. Kim - knowledge and information and management. `http://www.ontotext.com/kim`. [Online; Letzter Zugriff 12.09.2011].

[114] D. Homberger. *Sachwörterbuch zur Sprachwissenschaft*. Universal-Bibliothek. Reclam, 2003.

[115] F. K. Hwang, D. S. Richards, and P. Winter. *The Steiner Tree Problem*, volume 53 of *Annals of Discrete Mathematics*. Elsevier, 1992.

[116] I. Ibarretxe-Antuñano. Cross-linguistic polysemy in tactile verbs. In J. Luchjenbroers, editor, *Cognitive Linguistics Investigations across Languages, Fields, and Philosophical Boundaries.* John Benjamins, Philadelphia, 2006.

[117] D. Jurafski and J. H. Martin. *Speech and Language Processing*. Prentice Hall, 2000.

[118] V. Kacholia, S. Pandit, S. Chakrabarti, S. Sudarshan, R. Desai, and H. Karambelkar. Bidirectional Expansion for Keyword Search on Graph Databases. In *Proceedings Conference on Very Large Data Bases*, Trondheim, Norway, 2005.

[119] R. Karp. Reducibility Among Combinatorial Problems. In R. Miller and J. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103, New York, USA, 1975. Plenum Press.

[120] J. J. Katz. Semantic Theory. In D. D. Steinber and L. A. Jakobovits, editors, *Semantics - An Interdisciplinary Reader in Philosophy, Linguistics and Psychology*, chapter Linguistics 2, pages 297–307. Cambridge University Press, Cambridge, 2010.

[121] F. Kiefer. Some Implications for Lexicography. In T. Magay and J. Higany, editors, *Linguistic, Conceptual and Encyclopedic Knowledge*, pages 1–10, Budapest, September 1990.

[122] M. Kifer and G. Lausen. F-Logic: A Higher-Order language for Reasoning about Objects, Inheritance, and Scheme. In *Proceedings ACM SIGMOD International Conference on Management of Data*, pages 134–146, Portland, Oregon, 1989.

[123] W. Kintsch and T. A. van Dijk. Toward a model of text comprehension and production. *Psychological Review*, 85(5):363–394, 1978.

[124] H. Kittel, editor. *Übersetzung : ein internationales Handbuch zur Übersetzungsforschung*. Handbücher zur Sprach- und Kommunikationswissenschaft 26. de Gruyter Mouton, Berlin, 2004.

[125] O. Klaus. *Aristotels: Kategorien*. Aristoteles Werke. Buchgesellschaft, Darmstadt, 1984. Übers. und erläutert von Klaus Oehler.

[126] J. Klavans. Polysemy, Ambiguity and Generativity. Technical report, AAAI Symposium, 1995.

[127] J. Klostermeier. Peinliche Software-Panne sorgt für Einwohnerschwund. http://www.cio.de/public-ict/2217457/index2.html, 2007. [Online; letzter Zugriff am 12.09.2011].

[128] G. Klyne and J. J. Carroll. Resource Description Framework (RDF): Concepts and Abstract Syntax. http://www.w3.org/TR/rdf-primer/, 2004. [Online; letzter Zugriff am 12.09.2011].

[129] S. Kripke. Speaker's Refererence and Semantic Reference. *Midwest Studies in Philosophy*, 2:255–296, 1977.

[130] S. Kripke. Naming and Necessity. *Harvard University Press*, 1982.

[131] S. Kulkarni, A. Singh, G. Ramakrishnan, and S. Chakrabarti. Collective Annotation of Wikipedia Entities in Web Text. In J. F. E. IV, F. Fogelman-Soulié, P. A. Flach, and M. J. Zaki, editors, *KDD*, pages 457–466. ACM, 2009.

[132] G. Lakeoff. The Contemporary Theory of Metaphor. In A. Ortony, editor, *Metaphor and Thought*, pages 202–251. Cambridge University Press, New York, second edition, 1993.

[133] D. N. Le and A. Goh. Current Practices in Measuring Ontological Concept Similarity. In *Third International Conference on Semantics, Knowledge and Grid*, pages 266–269, Washington, DC, USA, 2007.

[134] T. B. Lee. Information Management: A Proposal. http://aiweb.techfak.uni-bielefeld.de/content/bworld-robot-control-software/, 1989. [Online; letzter Zugriff am 12.09.2011].

[135] P. D. Leenheer and A. D. Moor. Context-driven Disambiguation in Ontology Elicitation. In *Context and Ontologies: Theory, Practice, and Applications. Proceedings of the 1st Context and Ontologies Workshop, AAAI/IAAI 2005*, pages 17–24, Pittsburgh, Pennsylvania, 2005. AAAI Press.

[136] M. Lesk. Automatic Sense Disambiguation using Machine Readable Dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 5th annual international conference on Systems documentation*, SIGDOC '86, pages 24–26, Toronto, Ontario, Canada, 1986.

[137] V. Levenshtein. Binary Codes Capable of Correcting Deletions, Insertions, and Reversals. *Soviet Physics Doklady*, 10(8):707–710, 1966.

[138] T. Lewandowski. *Linguistisches Wörterbuch*. Heidelberg, 3 bde edition, 1980.

[139] G. Limaye, S. Sarawagi, and S. Chakrabarti. Annotating and Searching Web Tables Using Entities, Types and Relationships. *PVLDB*, 3(1):1338–1347, 2010.

[140] Loebner. *Semantik. Eine Einführung*. Gruyter, 2003.

[141] P. Ludlow. *Readings in the Philosophy of Language*. Massachuettes Institute of Technology, 1997.

[142] W. G. Lycan. *Philosophy of Language: A Contemporary Introduction (2th)*. Routledge, 2008.

[143] J. Lyons. *Einführung in die moderne Linguistik*. Beck, München, 8 edition, Feb. 1995.

[144] B. MacWhinney. Competition and Lexical Categorization. In R. Corrigan, F. Eckman, and M. Noonan, editors, *Linguistic Categorization: Vol 61 of Current Issues in Linguistic Theory*, pages 195–241. John Benjamins, Amsterdam/Philidelphia, 1989.

[145] J. Makhoul, F. Kubala, R. Schwartz, and R. Weischedel. Performance Measures For Information Extraction. In *In Proceedings of DARPA Broadcast News Workshop*, pages 249–252, Herndon, VA, 1999.

[146] V. Malais, L. Gazendam, and H. Brugman. Disambiguating Automatic Semantic Annotation Based on a Thesaurus Structure. In *Proceedings Traitement Automatique des Langues Naturelle*, Toulouse, France, 2007.

[147] N. A. A. Manaf, S. Bechhofer, and R. Stevens. A survey of Identifiers and Labels in OWL Ontologies. In *OWLED*, San Francisco, 2010.

[148] R. Manivannan and S. K. Srivatsa. Semi Automatic Method for String Matching. *Information Technology Journal*, 10(1):195–200, 2011.

[149] C. D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, 1999.

[150] A. McCallum and C. Sutton. An Introduction to Conditional Random Fields for Relational Learning. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning*. MIT Press, 2006.

[151] K. Mehlhorn and P. Sanders. *Algorithms and Data Structures - The Basic Toolbox*. Springer-Verlag, August 2008.

[152] A. Mikheev, C. Grover, and M. Moens. XML Tools and Architecture for Named Entity Recognition. *Markup Lang.*, 1:89–113, June 1999.

[153] G. A. Miller. WordNet: A Lexical Database for Englisch. *Communications of the ACM*, 38(11):39–41, 1995.

[154] D. N. Milne and I. H. Witten. Learning to Link with Wikipedia. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, pages 509–518, 2008.

[155] T. M. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.

[156] H. M. Müller. *Arbeitsbuch Linguistik*. Ferdinand Schöningh, Paderborn, 2002.

[157] D. Nadeau and S. Sekine. A Survey of Named Entity Recognition and Classification. *Linguisticae Investigationes*, 30(1):3–26, 2007.

[158] D. Nadeau, P. D. Turney, and S. Matwin. Unsupervised Named-Entity Recognition: Generating Gazetteers and Resolving Ambiguity, 2006.

[159] G. Navarro. A guided Tour to Approximate String Matching. *ACM Comput. Surv.*, 33:31–88, March 2001.

[160] W. T. Neill. Lexical Ambiguity and Context: An Activation-suppression Model. *Resolving semantic ambiguity*, pages 63–84, 1989.

[161] H. T. Nguyen and T. H. Cao. A Knowledge-Based Approach to Named Entity Disambiguation in News Articles. In *Proceedings International Conference of Artificial Intelligence*, Cambridge, England, 2007.

[162] H. T. Nguyen and T. H. Cao. Named Entity Disambiguation: A Hybrid Statistical and Rule-Based Incremental Approach. In *ASWC, Proceedings Asian Semantic Web Conference*, Bangkok, China, 2008.

[163] H. T. Nguyen and T. H. Cao. Named Entity Disambiguation on an Ontology Enriched by Wikipedia. In *RIVF 2008, International Conference on Computing and Communication Technologies*, Ho Chi Minh City, Vietnam, 2008.

[164] H. T. Nguyen and T. H. Cao. Exploring Wikipedia and Text Features for Named Entity Disambiguation. In *2nd Asian Conference on Intelligent Information and Database Systems*, pages 11–20, Hue, Vietnam, 2010.

[165] E. A. Nida. *Componential Analysis of Meaning. An Introduction to Semantic Structures*. The Hauqe, Paris: Mouton, 1975.

[166] D. Oberle, A. Ankolekar, P. Hitzler, P. Cimiano, M. Sintek, M. Kiesel, B. Mougouie, S. Baumann, S. Vembu, M. Romanelli, P. Buitelaar, R. Engel, D. Sonntag, N. Reithinger, B. Loos, H.-P. Zorn, V. Micelli, R. Porzel, C. Schmidt, M. Weiten, F. Burkhardt, and J. Zhou. DOLCE ergo SUMO: On Foundational and Domain Models in the SmartWeb Integrated Ontology (SWIntO). *Web Semantics*, 5(3):156–174, 2007.

[167] C. Ogden and D. S. Richards. *Die Bedeutung der Bedeutung: Eine Untersuchung über den Einfluss der Sprache auf das Denken über die Wissenschaft des Symbolismus*. Frankfurt am Main, 1974.

[168] B.-W. On and D. Lee. Scalable Name Disambiguation using Multi-level Graph Partition. In *SDM 2007, Proceedings of the 7th SIAM International Conference on Data Mining*, Minneapolis, Minnesota, USA, 2007.

[169] E. Padučeva. Paradigma reguljarnoj mnogoznačnosti glagolov zvuka. *Voprosy jazykoznanija*, 5:3–23, 1998.

[170] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Stanford Digital Library Technologies Project, 1998.

[171] E. Palmon and S. Geva. Associative search method for heterogeneous databases with an integration mechanism configured to combine schema-free data models such as hyperbase. United States Patent Number 5,829,264, granted October 6, 1998, filled in 1995. Available at www.uspto.gov, 1998.

[172] H. Pelz. *Linguistik - Eine Einführung*. Hoffmann und Campe Vlg Gmbh, 1996.

[173] G. Pethö. Konzeptuelle Fokussierung. Bemerkungen zur Behandlung der Polysemie in der Zwei-Ebenen-Semantik. In *Meta Linguitica 7*, Frankfurt, 2001.

[174] G. Pethö. What Is Polysemy? A Survey of Current Research and Results. In K. Bibok and E. Németh, editors, *Pragmatics and the Flexibility of Word Meaning*, pages 175–224, Amsterdam, 2001. Elsevier.

[175] M. Pfuhl. *Case-Based Reasoning auf der Grundlage Relationaler Datenbanken - Eine Anwendung zur strukturierten Suche in Wirtschaftsnachrichten*. PhD thesis, Universität Marburg, 2003.

[176] G. Piatetsky-Shapiro. Knowledge Discovery in Real Databases: A Report on the IJCAI-89 Workshop. *AI Magazine*, 11(5):68–70, 1991.

[177] M. Pinkal. Vagheit und ambiguität. *Smantik. Ein internationales Handbuch der zeitgenössischen Forschung*, pages 250–259, 1991.

[178] R. Poli. Res, Ens and Aliquid. *Poli and Simons*, 1993.

[179] B. Popov, A. Kiryakov, A. Kirilov, D. Manov, D. Ognyanoff, and M. Goranov. KIM – Semantic Annotation Platform. *Journal of Natural Language Engineering*, 10(3-4):375–392, 204.

[180] S. Preece. *A Spreading Activation Network Model for Information Retrieval*. PhD thesis, University of Illinois, Urbana-Champaign, 1985.

[181] H. J. Prömel and A. Steger. *The Steiner Tree Problem*. Advanced Lectures in Mathematics. vieweg, 2002.

[182] J. Pustejovsky. Type Coercion and Lexical Selection. In J. Pustejovsky, editor, *Semantics and the Lexicon*, pages 73–94. Kluwer, London, 1993.

[183] D. Quathamer. *Kohärenzbildung beim Lesen von Texten – Nutzung und Funktion von Überblicksdiagrammen*. PhD thesis, Gerhard Mercator Universität, Gesamthochschule Duisburg, 15.12.1998.

[184] M. Quillian. A Revised Design for an Understanding Machine. *Mechanical Translation*, 7(1):17–29, 1962.

[185] M. Quillian. *Semantic Memory*. PhD thesis, Carnegie Instiute of Technology, 1966. Unpublished but later reprinted in part in M. Minsky [ed.], Semantic information processing. Cambridge, Mass.: MIT Press, 1968.

[186] R. Quirk, S. Greenbaum, G. Leech, and J. Svartvik. *A Comprehensive Grammar of the English Language*. Longman Group Limited, New York, 1985.

[187] M. Rada. Unsupervised Large-Vocabulary Word Sense Disambiguation with Graph-based Algorithms for Sequence Data Labeling. In *Proceedings Human Language Technology Conference*, Vancouver, Canada, 2005. ACL.

[188] L. Ramshaw and M. Marcus. Text Chunking Using Transformation-Based Learning, 1995.

[189] B. Rieger. Vagheit als Problem der linguistischen Semantik. *Semantik und Pragmatik*, pages 91–101, 1977.

[190] C. Rocha, D. Schwabe, and M. P. Aragao. A hybrid Approach for Searching in the Semantic Web. In *Proceedings of the 13th International Conference on World Wide Web*, WWW '04, pages 374–383, New York, USA, 2004.

[191] G. Rodriguez. Bundesweite Vornamenstatistik 2007. `http://www.uni-leipzig.de/vornamen/wcms/index.php?option=com_content&view=article&id=45&Itemid=62`, 2007. [Online; letzter Zugriff am 12.09.2011].

[192] C. Römer and B. Matzke. *Lexikologie des Deutschen. Eine Einführung*. Gunter Narr Verlag, Tübingen, 2 auflage edition, 2005.

[193] W. D. Ross. *Aristotle's Metaphysics*, chapter VIII, part 6. Clarendon Press, Oxford, 1924.

[194] D. Rummelhart and J. L. McClelland. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. MIT Press, 1986.

[195] S. Saha, S. Narayan, S. Sarkar, and P. Mitra. A Composite Kernel for Named Entity Recognition. *Pattern Recognition Letters*, 31(12):1591–1597, 2010. Pattern Recognition of Non-Speech Audio.

[196] G. Salton and C. Buckley. On the Use of Spreading Activation Methods in Automatic Information Retrieval. In *SIGIR '88 Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Grenoble, France, 1988.

[197] M. Sanderson. *Word sense disambiguation and information retrieval*. PhD thesis, University of Glasgow, England, September 1996.

[198] D. Sankoff and J. B. Kruskal, editors. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison-Wesley Publishing Co, Reading, Massachusetts, 1983.

[199] R. C. Schank. *Conceptual Information Processing*. Elsevier Science Inc., New York, NY, USA, 1975.

[200] P. Scheir. *Assoziative Suche fur das Semantic Web*. PhD thesis, Technische Universität Graz, 2008.

[201] T. Schippan. *Einführung in die Semasiologie*. Bibliographisches Institut, 1972.

[202] T. Schippan. *Lexikologie der deutschen Gegenwartssprache*. Niemeyer, Tübingen, 1992.

[203] E. W. Schneider. *Variabilität, Polysemie und Unschärfe der Wortbedeutung*, volume 1 of *Theoretische und methodische Grundlagen*. Niemeyer, Tübingen, 1988.

[204] M. Schneider. OWL 2 Web Ontology Language RDF-Based Semantics. *W3C Recommendation 27 October 2009*, 2009. [Online; letzter Zugriff am 12.09.2011].

[205] W. Schnotz. *Aufbau von Wissensstrukturen : Untersuchungen zur Kohärenzbildung beim Wissenserwerb mit Texten*. Fortschritte der psychologischen Forschung ; 20. Beltz, Weinheim, 1994. Wolfgang Schnotz. graph. Darst 21 cm. Literaturverz. S. 315 - 362. 5*GBV*.

[206] M. Schömann. Menschliche Informationsverarbeitungsprozesse bei der Disambiguierung, January 1995. Report 53 im Rahmen des Verbundvorhabens Verbmobil.

[207] S. Schrauwen. Maschine Learning Approaches to Sentiment Analysis using the Dutch Netlog Corpus. *Computational Linguistics & Psycholinguistics*, CTRS-001, July 2010.

[208] M. Schwarz. *Einführung in die Kognitive Linguistik*. Francke, Tübingen, 1992.

[209] J. Searle. *Speech acts: an essay in the philosophy of language*. Cambridge University Press, 2 edition, 1969.

[210] S. Sekine and H. Isahara. IREX: IR and IE Evaluation project in Japanese. In *Conference on Language Resources and Evaluation*, Athens, Greece, 2000.

[211] S. Sekine and C. Nobata. Definition, Dictionaries and Tagger for Extended Named Entity Hierarchy. In *Conference on Language Resources and Evaluation*, Lisbon, Portugal, 2004.

[212] S. Sekine and E. Ranchhod. *Named Entities*. Benjamins, 2009.

[213] J. Shen. Named Entity Recognition of Ontology Elements. Master's thesis, Karlsruhe Institute of Technology, March 2010. Betreuer: Rudi Studer, Joachim Kleb.

[214] E. Simoudis, B. Livezey, and R. Kerber. Integrating Inductive and Deductive Reasoning for Data Mining. In *Advances in Knowledge Discovery and Data Mining*, pages 353–373. MIT Press, 1996.

[215] R. Sinha and M. Rada. Unsupervised Graph-based Word Sense Disambiguation Using Measures of Word Semantic Similarity. In *First IEEE International Conference on Semantic Computing*, Irvine, California, 2007. IEEE Computer Society.

[216] S. Small, G. Cottrell, and M. Tanenhaus, editors. *Lexical Ambiguity Resolution - Perspectives from Psycholinguistics, Neuropsychology & Artificial Intelligence*. Morgan Kaufmann, 1988.

[217] Solso. *Kognitive Psychologie*. Springer-Verlag, 2003.

[218] J. F. Sowa. *Knowledge Representation: logical, philosophical and computational foundations*. Brooks/Cole Publishing Co., Pacific Grove, CA, USA, 2000.

[219] S. Staab and R. Studer, editors. *Handbook on Ontologies (2nd edition)*. International Handbooks on Information Systems. Springer, 2008.

[220] R. Steinberger and B. Pouliquen. Cross-lingual Named Entity Recognition. *Linguisticae Investigationes*, 30(1):3–26, January 2007.

[221] R. Steinberger, B. Pouliquen, and E. van der Goot. An Introduction to the Europe Media Monitor Family of Applications . In N. K. Gey and J. Narlgren, editors, *Information Access in a Multilingual World* , Proceedings of the SIGIR 2009 Workshop (SIGIR-CLIR'2009), pages 1–8, Boston, USA, July 2009.

[222] R. Studer, V. R. Benjamins, and D. Fensel. Knowledge Engineering: Principles and Methods. *Data and Knowledge Engineering*, 25(1-2):161–197, 1998.

[223] R. Studer, S. Grimm, and A. Abecker, editors. *Semantic Web Services, Concepts, Technologies, and Applications*. Springer-Verlag, Berlin, Heidelberg, 2007.

[224] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

[225] M. syan Chen, J. Han, and P. S. Yu. Data Mining: An Overview from a Database Perspective. *IEEE Transactions on Knowledge and Data Engineering*, 8:866–883, 1996.

[226] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. Addison Wesley, us ed edition, 2005.

[227] I. Terziev, A. Kiryakov, D. Manov, et al. Base Upper-Level Ontology (BULO) Guidance1 . `http://proton.semanticweb.org/D1_8_1.pdf`, 2003. [Online; Letzter Zugriff 12.09.2011].

[228] M. S. C. Thomas and J. L. McClelland. Connectionist Models of Cognition. In R. Sun, editor, *Handbook on Ontologies*, Handbook of Computational Psychology, pages 23–30. Campridge University Press, 2008.

[229] T. Tran, H. Wang, S. Rudolph, and P. Cimiano. Top-k Exploration of Query Candidates for Efficient Keyword Search on Graph-Shaped RDF Data. In *Proceedings International Conference on Data Engineering*, Shanghai, China, 2009. IEEE.

[230] G. Tsatsaronis, M. Vazirgiannis, and I. Androutsopoulos. Word Sense Disambiguation with Spreading Activation Networks Generated from Thesauri. In *Proceedings of the International Conference on Data Engineering*, Istanbul, Turkey, 2007.

[231] E. Tulving. Episodic and Semantic Memory. In W. Donaldson, editor, *Organiszation of Memory*, pages 381–403. New York: Academic Press, 1972.

[232] S. Ullmann. *Grundzüge der Semantik: die Bedeutung in sprachwissenschaftlicher Sicht*. De Gruyter Lehrbuch. Brockmeyer, Unviersitätsverlag, 1972.

[233] W. Ulrich. *Wörterbuch Linguistische Grundbegriffe*. Hirt's Stichwortbücher. Bortraeger, 5 edition, 2002.

[234] J. C. van Rijsbergen. *Information Retrieval*. Butterworth-Heinemann Ltd, second edition, 1979.

[235] J. Veronis and N. M. Ide. Word Sense Disambiguation With Very Large Neural Networks Extracted From Machine Readable Dictionaries. In *13th International Conference on Computational Linguistic*, Helsinki, Finnland, 1990. ACL.

[236] W3C. Linking open data. `http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData#Project_Description`. [Online; Letzter Zugriff 12.09.2011].

[237] H. R. Walpole. *Semantics:The Nature of Words and their Meanings*. WW Norton, 1941.

[238] D. L. Waltz and J. B. Pollack. Massively parallel parsing: A strongly interactive model of natural language interpretation. *Cognitive Science*, 9:51–74, 1985.

[239] M. Wandruszka. Polymorphie und Polysemie. *Festschrift für Hugo Moser zum 60. Geburtstag am 19. Juni 1969*, pages 218–232, 1969.

[240] M. P. Wolf. Kripke, Putnam and the Introduction of Natural Kind Terms. *Acta Analytica*, 17(28):51–70, 2002.

[241] W. A. Woods. What's in a Link: Foundations for Semantic Networks. In D. Bobrow and A. Collins, editors, *Representation and Understanding*. Academic Press, 1975.

[242] World Wide Web Consortium. *SKOS Simple Knowledge Organization System Reference*, 2009.

[243] World Wide Web Consortium (W3C). Web Ontology Language (OWL). `http://www.w3.org/2004/OWL/`, 2004. [Online; letzter Zugriff am 12.09.2011].

[244] World Wide Web Consortium (W3C). Web Ontology Language (OWL). `http://www.w3.org/2007/OWL/`, 2007. [Online; letzter Zugriff am 12.09.2011].

[245] B. Y. Wu and K.-M. Chao, editors. *Spanning Trees and Optimization Problems*. Discrete Mathematics and Its Applications. Chapman & Hall/CRC, 2004.

[246] D. Wunderlich and A. Stechow. *Semantik: ein internationales Handbuch der zeitgenössischen Forschung*. Handbücher zur Sprach- und Kommunikationswissenschaft. Mouton De Gruyter, 1991.

[247] H. Xiao and I. F. Cruz. Integrating and Exchanging XML Data Using Ontologies. *Journal on Data Semantics VI: Special Issue on Emergent Semantics*, 4090:67–89, 2006.

[248] J. X. Yu, L. Qin, and L. Chang. *Keyword Search in Relational Databases*. Synthesis Lectures on Data Management. Morgan & Claypool, 2010.

[249] J. X. Yu, L. Qin, and L. Chang. Keyword Search in Relational Databases: A Survey. *IEEE Data Eng. Bull.*, 33(1):67–78, 2010.

[250] U. Zernik. TRAIN1 vs. TRAIN2: Tagging word senses in corpus. In *Proceedings of RIAO 91, Intelligent Text and Image Handling*, pages 567–585, Barcelona, Spain, 1991.

[251] J. Zhu, V. Uren, and E. Motta. ESpotter: Adaptive Named Entity Recognition for Web Browsing. In *WM-2005 – Conference on Professional Knowledge Management. Workshop on Intelligent IT Tools for Knowledge Management*, Kaiserslautern, Germany, 2005.

[252] X. Zhu and A. B. Goldberg. *Introduction to Semi-Supervised Learning*. Morgan and Claypool Publishers, 2009.

Eigene Veröffentlichungen

[253] J. Henss, J. Kleb, and S. Grimm. A Protege 4 Backend for native OWL Persistence. In *Proceedings of the International Protege Conference*, Amsterdam, Netherlands, June 2009.

[254] J. Henss, J. Kleb, S. Grimm, and J. Bock. A Database Backend for OWL. In *Proceedings of the 5th International Workshop on OWL: Experiences and Directions (OWLED 2009)*, volume 529, Chantilly, VA, United States, 2009.

[255] J. Kleb and A. Abecker. Entity Reference Resolution via Spreading Activation on RDF-Graphs. In *Proceedings of the 7th Extended Semantic Web Conference*, Heraklion, Crete, Greece, 2010.

[256] J. Kleb and A. Abecker. Disambiguating Entity References within an Ontological Model. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics (WIMS)*, Sandgal, Norway, 2011.

[257] J. Kleb and R. Volz. Ontology based Entity Disambiguation with Natural Language Patterns. In *Proceedings Fourth International Conference on Digital Information Management*, Ann Arbor, USA, 2009.

[258] M. Spahn, J. Kleb, S. Grimm, and S. Scheidl. Supporting Business Intelligence by providing Ontology-based End-User Information Self-Service. In *Proceedings of the First International Workshop on Ontology-supported Business Intelligence*, Karlsruhe, Germany, 2008.

[259] R. Volz, J. Kleb, and W. Müller. Towards Ontology-based Disambiguation of Geographical Identifiers. In *Proceedings of the WWW Workshop I3*, Banff, Canada, 2007.

Ontologien, die in modernen Anwendungssystemen, als Hintergrundwissen zur Anwendung kommen, verlangen zwingend die eindeutige Identifikation der darin beschriebenen Elemente. Ontologien finden zudem immer weitere Verbreitung, da diese die Möglichkeit bieten, Wissen über ein Netzwerk logischer Relationen zu repräsentieren. Insofern ist das Auftreten von Mehrdeutigkeit innerhalb von Ontologien formal ausgeschlossen. Aufgrund der zunehmenden Verbreitung von Ontologien steigt die Notwendigkeit Informationen in natürlicher Sprache in Ontologien zu integrieren bzw. mit den darin enthaltenen Informationen in Einklang zu bringen. Mit der Einbindung natürlicher Sprache erhält auch die Thematik der Mehrdeutigkeit Einzug in die formal geordnete Darstellung. Die Suche nach relevanter Information basierend auf natürlicher Sprache ist in der Konsequenz nicht mehr eindeutig möglich.

Die Vorliegende Arbeit beschäftigt sich mit dieser Thematik, insbesondere mit dem genannten Problem der Ambiguität, die bei der Zusammenführung natürlich-sprachlicher Informationen mit dem durch Ontologien repräsentieren Wissen auftritt und zeigt ein neues Verfahren zur Lösung des Problems auf.