UNIVERSITÄT LEIPZIG
Fakultät für Mathematik und Informatik
Institut für Informatik

Automatische Extraktion von Termhierarchien aus Dokumentenkollektionen für die semantische Strukturierung

Diplomarbeit

Leipzig, Januar 2007

vorgelegt von:
Florian Holz

Inhaltsverzeichnis

Abbildungsverzeichnis

Kapitel 1

Einleitung

Durch die technische Entwicklung und zunehmende Verbreitung der Computertechnik in den letzten Jahren sind immer größere Mengen von Dokumenten auf immer einfachere Art und Weise für den einzelnen Nutzer zugänglich. Das ist insbesondere auf die zunehmende elektronische Existenz von Dokumenten und die Vernetzung der verschiedenen Datenquellen zurückzuführen. Dies betrifft sowohl große, „zentrale“ Datenbestände wie z. B.von Bibliotheken und Verlagen, aber durch den dezentralen Aufbau des Internets auch viele kleine „lokale“ Datenbestände bei den jeweiligen Nutzern. Suchmaschinen wie google bzw. die Nutzung der P2P-Technik ermöglichen es, auch auf diese verteilten lokalen Dokumente direkt zuzugreifen.

Allerdings stellt die riesige Menge der so verfügbaren Dokumente auch ein neues Problem dar.

Zum einen ist es mitunter sehr aufwendig, ein ganz bestimmtes Dokument zu finden, weil die üblichen Suchansätze, sowohl bei Suchmaschinen als auch bei P2P-Clients hauptsächlich über Suchbegriffe, die in den fraglichen Dokumenten vorkommen müssen, oftmals eine sehr große Menge möglicherweise passender Dokumente liefern.

Zum anderen ist es in dem Fall, daß nicht nach einem bestimmten Dokument gesucht wird, sehr aufwendig, die Menge der gefundenen und damit viel leichter als früher verfügbaren Dokumente zu sichten.[1] Das betrifft z. B. Situationen, in denen man sich einen Überblick über ein Themengebiet verschaffen will. Die Rankingalgorithmen, die die relevantesten Dokumente

[1] Als noch weitaus weniger Dokumente, in Form von Büchern in Bibliotheken, leicht verfügbar waren, war damit eine gewisse Vorauswahl getroffen.

ganz nach vorne rücken sollen, funktionieren nicht immer hinreichend gut, und müssen versagen, wenn es zu einem breiten Themengebiet viele gleich relevante, zum Teil sehr spezielle, zum Teil eher allgemeine Dokumente gibt.

In dieser Arbeit wird sich mit dem zweiten Problem beschäftigt und versucht, eine Lösung dafür zu finden, indem Verfahren getestet werden, die dazu dienen sollen, große, unstrukturierte Mengen von Dokumenten so zu ordnen, daß nicht nur die einzelnen betrachteten Teilthemen herausgearbeitet werden, sondern es auch möglich ist, eine Sicht vom Allgemeinen zum Speziellen zu erhalten. Dabei ist es nicht nur das Ziel, die Dokumente zu ordnen und den einzelnen Bereichen zuzuordnen und diese Bereiche miteinander in Beziehung zu setzen, sondern es soll auch versucht werden, den Vorgang der Zuordnung transparent zu machen, die hinter der Zuordung liegende Struktur erkennbar werden zu lassen und möglicherweise jedem Bereich eine Beschreibung zuzuordnen.

Kapitel 2

Vorbetrachtungen

Wie in Kapitel 1 beschrieben besteht die Zielstellung dieser Arbeit darin, Verfahren zu finden, auszuprobieren und zu bewerten, mithilfe derer eine Menge von Texten inhaltlich strukturiert werden kann, und zwar so, daß die entstehende Struktur nachvollziehbar ist und die gruppierten Dokumente beschrieben werden können. Um diese Aufgabe zu bewältigen, sind zwei Dinge zu klären:

- Welche Eigenschaften der Dokumente können/sollen benutzt werden?
- Welcher Art ist die zu extrahierende Struktur, in die die Dokumente gebracht werden sollen?

Die Beantwortung dieser beiden Fragen kann nicht linear erfolgen, d. h. erst die eine und dann die andere, sondern die Antworten bedingen sich gegenseitig. Absolut einschränkend ist natürlich die Tatsache, daß Eigenschaften, die in den Dokumenten nicht vorliegen bzw. aus ihnen nicht erhalten werden können, nicht für die Struktur benutzt werden können.

Abgesehen davon ist auch zu überlegen, welche Struktur einer Repräsentation der inhaltlichen Zusammenhänge in und zwischen den Dokumenten möglichst gut nahekommt.

Für die Beantwortung der beiden Fragen werden im folgenden die nötigen und möglichen Voraussetzungen beschrieben, um danach daraus schlußfolgernd die Entscheidungen für die konkrete Umsetzung der Aufgabenstellung zu treffen.

2.1 Eigenschaften der Struktur

Beschreibungen der Relationen von Dingen, über die gedacht, geschrieben, gesprochen werden kann, stellen Ontologien dar. Die derartige Verwendung des Begriffs „Ontologie“ ist gegenüber seiner ursprünglichen Bedeutung sehr einschränkend. Diese ursprüngliche Bedeutung als „Lehre des Seins“ (als Übersetzung der griechischen Wurzel) ist z. B. in der Philosophie maßgeblich. In der Informatik jedoch ist der Begriff anwendungsorientiert auf Spezifikationen von Konzeptualisierungen und Instanzen davon verengt [4]. Da der philosophische Aspekt zu weit führen würde und das Thema dieser Arbeit ein konkret anwendungsbezogenes Informatikthema ist, wird ab jetzt der Begriff „Ontologie“ nur noch im informatischen Sinne gebraucht.

Diese Ontologien in der Informatik sind üblicherweise hierarchisch organisiert[2]und enthalten meist thematische Relationen wie ALLGEMEINER-ALS oder OBERBEGRIFF-VON. Ähnliche Entitäten werden über eine Ähnlichkeitsrelation identifiziert. Das trifft die Anforderungen für das Zugänglichmachen bzw. Ordnen der Dokumente. Deswegen ist ein Blick auf die bisher existierenden Ontologien zu werfen, ob eine davon für die hiesigen Zwecke einsetzbar ist.

2.1.1 Ontologiemodelle

Bei den anwendungsorientierten Ontologien lassen sich drei Arten unterscheiden. Zwei davon basieren explizit auf einer Hierarchie der Konzeptualisierung, wo die Eigenschaften und Relationen des repräsentierten Sachgebiets in Form der Hierarchie ausgedrückt werden. Die andere basiert auf einem axiomatisch-formalen Ansatz [4].

Eine Hierarchie (auch Polyhierarchie) ist ein gerichteter, azyklischer, zusammenhängender Graph mit einer Wurzel, also einem Knoten ohne eingehende Kanten. Wesentlich für eine Hierarchie ist auch, daß für jeden Knoten alle Pfade von der Wurzel zu diesem Knoten gleichlang sind, sodaß die Hierarchie als in Ebenen geschichtet angesehen werden kann, wobei jeder Knoten eindeutig einer Ebene zuzuordnen ist. Wenn jeder Knoten außer der Wurzel nur eine eingehende Kante hat, heißt die Hierarchie Monohierarchie und entspricht einem graphentheoretischen Baum.

[2]abgesehen von manchmal den Ontologien zugerechneten Wissensspeichern wie Glossaren oder Thesauri

Entgegen der durch diese Beschreibung naheliegenden Konvention meint eine „Hierarchie“ und „hierarchisch“ in dieser Arbeit immer eine „Monohierarchie“ und „monohierarchisch“. Wenn eine „Polyhierarchie“ und „polyhierarchisch“ gemeint ist, wird es explizit benannt.

Axiomatisch formale Ontologien

Eine axiomatisch formale Ontologie repräsentiert das Wissen als Axiome im Rahmen eines logischen Kalküls. Über die im jeweiligen Kalkül gültigen Regeln können weitere Sätze über die repräsentierten Entitäten abgeleitetet werden.

Ein mit den Erfahrungen der hiesigen Biologen nicht ganz übereinstimmendes Beispiel bildet die folgende Formalisierung mit zwei Axiomen:

$$\lambda x\ \textsc{Pinguin}(x)\ , \quad \lambda x\ \textsc{Vogel}(x)\ , \quad \lambda x\ \textsc{Kann-fliegen}(x)\ ,$$
$$\lambda Q \lambda P\ P\ \textsc{ist-ein}\ Q\ ,$$
$$\forall x\ [\textsc{Pinguin}(x) \rightarrow \textsc{Vogel}(x)]\ ,$$
$$\forall P\ \big([\forall x P(x) \rightarrow \textsc{Kann-fliegen}(x)] \rightarrow [P\ \textsc{ist-ein}\ \textsc{Vogel}]\big)\ .$$

Als Kalkül können verschiedene Logiken wie Prädikatenlogiken, meist in der Mächtigkeit beschränkt wie z. B. die monadische Logik um die Berechenbarkeit zu erhalten, oder nichtmonotone Logiken mit an die Realität angepaßten Schlußfolgerungsregeln zum Einsatz kommen.

Terminologien/Taxonomien

Hierarchien von Begriffen, die durch die Oberbegriff-Unterbegriff-Relation bzw. die IST-EIN-Relation in Beziehung stehen, heißen Terminologien bzw. Taxonomien.

Eine Beispieltaxonomie aus der Gebäudedomäne ist in Abbildung 2.1 gegeben.

Prototypbasierte Ontologien

Prototypbasierte Ontologien stellen instanzbasierte Kategorisierungen dar, wobei eine Kategorie durch die Beispielentitäten, die sie enthält, definiert wird und die Zusammenfassung aller ihrer Subkategorien ist. Die Relation

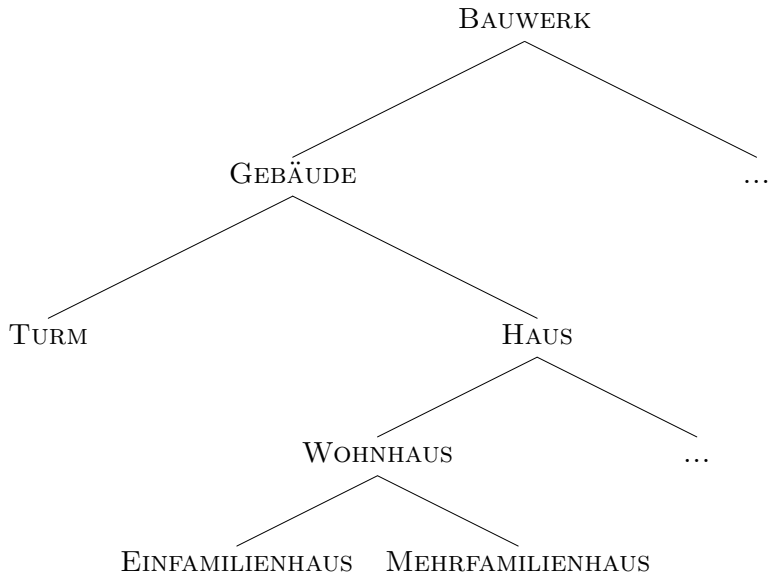

Abbildung 2.1: Eine Beispieltaxonomie

zwischen den Super- und Subkategorien ist die Teil(menge)-von-Relation. Prototypbasierte Ontologien stellen ein Dendrogramm der Entitäten dar.

Das Beispiel der Taxonomie ist in Abbildung 2.2 als prototypbasierte Ontologie gegeben.

2.1.2 WordNet

Abgesehen von den im vorigen Abschnitt beschriebenen prinzipiellen Modellen für Ontologien gibt es schon ausformulierte, also mit Inhalt gefüllte Ontologien. Die meisten davon sind sachgebietsspezifisch. Da allerdings Verfahren gesucht werden, die beliebige Dokumentenkollektionen strukturieren sollen, sind diese hier nicht verwendbar.

Ein möglicher Kandidat für allgemeines Vorwissen ist hingegen WordNet [12]. WordNet ist ein strukturiertes Lexikon der englischen Sprache. Es basiert auf Mengen synonymer Wörter, sogenannte Synsets, die in bestimmten Relationen stehen. Bei den Wörtern werden generell Substantive, Verben, Adjektive und Adverben unterschieden, d. h. in einem Synsets sind immer nur Wörter der selben Wortart. Mögliche Relationen zwischen Substantivsynsets sind Hyponymie (ist-ein), Meronymie (Teil-von) und coordinate terms (zwei Synsets haben ein gemeinsames Hyperonym). Polyseme Wörter treten in verschiedenen Synsets auf, und repräsentieren in jedem Synset eine ihrer Bedeutungen.

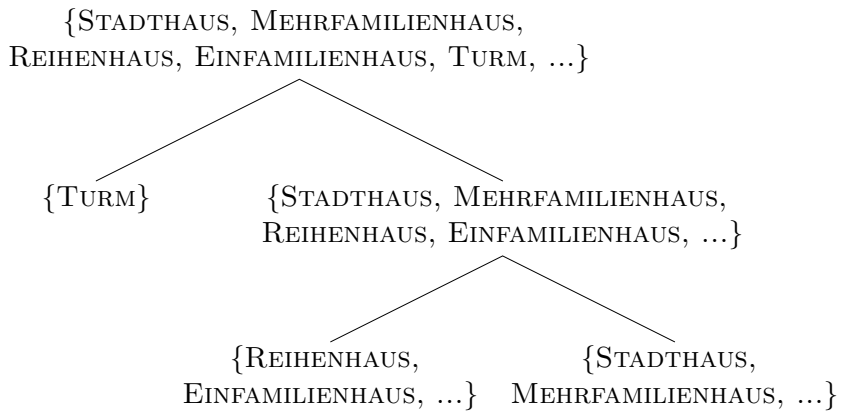

Abbildung 2.2: Ein Beispiel einer prototypbasierten Ontologie

In der aktuellen Version enthält WordNet 155 327 Wörter in 117 597 Synsets, davon 117 097 Substantive in 145 104 Synsets, deckt also einen großen Teil der englischen Allgemeinsprache ab [34].

2.2 Eigenschaften von Dokumentenkollektionen und Dokumenten

Um abschätzen zu können, welche Informationen für die Strukturierung bzw. die entstehende Struktur überhaupt verfügbar sind, ist zu klären, welche Informationen in der Dokumentenkollektion und den einzelnen Dokumenten enthalten sind bzw. welche Eigenschaften diese aufweisen und welche Annahmen getroffen werden können.

Ein Dokument ist (ein) Text, bestimmte Eigenschaften hat. Wesentliche Eigenschaften sind:

- Textsorte
- Sachgebiet
- Repräsentation
- Struktur
- statistische Eigenschaften
- linguistische Eigenschaften

Diese Merkmale werden im folgenden darauf untersucht, inwieweit ihre Verwendung für die Strukturierung einer Dokumentenkollektion sinnvoll erscheint bzw. sie bei der Auswahl der einzusetzenden Verfahren berücksichtigt werden müssen.

2.2.1 Textsorte und Sachgebiet

Es kann nicht gesagt werden, daß die Textsorte und das Sachgebiet der Dokumente einer zu analysierenden Kollektion keine Rolle spielen. Es ist vielmehr zu erwarten, daß Dokumente, die technischerer Natur sind, und somit die unten erläuterten und später verwendeten statistischen Eigenschaften stärker aufweisen, besser zur Strukturierung geeignet sind als Dokumente, bei denen auszeichnende Merkmale nicht in der Art enthalten sind, wie sie die eingesetzten Verfahren erwarten.

Da die Optimierung der Verfahren z. B. auf bestimmte Textsorten oder Sachgebiete nicht Gegenstand dieser Arbeit ist, wird diesen beiden Eigenschaften nur dahingehend Aufmerksamkeit gewidmet, daß Experimente mit Dokumenten verschiedener Textsorten aus verschiedenen Sachgebieten durchgeführt werden, um die Eignung der eingesetzten Verfahren vergleichen zu können (siehe Abschn. 5.2.3).

2.2.2 Repräsentation und Struktur

Für die Repräsentation elektronisch vorliegender Dokumente, also für ihr Datenformat, gibt es unzählige Möglichkeiten: XML, HTML, PDF, PS, TXT, Datenbanken und auch viele Formate sogenannter Textverarbeitungsprogramme.[3] Auch wenn bei den meisten Datenquellen mit wissenschaftlichem Anspruch wie z. B. Bibliotheken eine einheitliche Datenstruktur zu erwarten ist, kann diese trotzdem für mehrere Datenquellen verschieden sein.

Das Hauptproblem bei den verschiedenen Datenformaten ist die unterschiedliche Repräsentation der Dokumentstruktur. Allein XML ist von den obengenannten darauf ausgelegt, Inhalte mit Struktur zu verwalten, und auch diese wird durch eine DTD festgelegt, die wiederum variabel ist. Bei den anderen Formaten erfolgen gliedernde Auszeichnung auf individuelle Art

[3] Mit OpenDocument (ODF) existiert für letztere erst seit Dezember 2006 ein offizieller ISO-Standard.

und Weise auf graphischer Ebene, die erst aufwendig wiedererkannt werden müssen.

Da für diese Arbeit eine Machbarkeitsstudie und noch keine Festlegung oder Optimierung für bestimmte Dokumentquellen vorgesehen ist, sondern im Gegenteil die Verfahren ersteinmal für verschiedene Eignungen zu evaluieren sind, wird bei der Verarbeitung der Dokumente von ihrer Repräsentation und inneren Struktur abstrahiert. Die Dokumente werden auf den kleinsten gemeinsamen Nenner des einfach fortlaufenden Textes ohne spezielle Hervorhebungen wie Kapitel- und Absatzeinteilungen, Überschriften, Schlagworte, Fettdruck, etc. gebracht. Daraus, daß durch diese Hervorhebungen besonders häufig für das jeweilige Dokument gut beschreibende Wörter markiert sind, könnte bei weiteren Arbeiten durchaus Nutzen gezogen werden.

2.2.3 Statistische Eigenschaften

Für die statistische Betrachtung der Verteilung von Wörtern in Text(en) hat sich das Zipfsche Gesetz als grundlegend erwiesen [14, S. 87ff]. Sortiert man die Wörter absteigend nach ihrer Frequenz f und weist jedem Wort somit einen Rang r zu, dann gilt:

$$f * r = \text{konstant} . \tag{2.1}$$

Daraus folgt, daß einige wenige Wörter, die am häufigsten vorkommen, den Hauptteil des Textes ausmachen, wohingegen ungefähr die Hälfte der Wörter nur einmal auftritt [20, S. 20ff.].[4]

Die am häufigsten auftretenden Wörter sind Artikel, Präpositionen, Konjunktionen, etc, also Funktionswörter bzw. Synsemantika, die nur im Kontext der anderen Wörter bedeutsam sind, aber nicht zum Thema des Dokuments gehören[5] und nicht zu seiner Auszeichnung gegenüber anderen Dokumenten beitragen. Diese Synsemantika werden auch als Stopwörter bezeichnet und treten nicht nur in jedem Text besonders häufig auf, sondern auch in allen Texten gleichmäßig häufig, was ihre fehlende Relevanz für das konkrete Thema belegt. Dadurch können sie für die weitere Verarbeitung der Dokumente ignoriert werden.

[4] Zur genaueren Differenzierung zwischen Wörtern und ihrem Auftreten siehe Abschnitt 3.1.1.

[5] Es sein denn natürlich, daß das Thema des Textes ebenjene Wörter sind.

Im Gegensatz zu den erwartet häufigen Stopwörtern sind unter den weniger häufigen Wörtern einige, die gemessen an ihrer Gesamthäufigkeit in einer Sprache überraschend oft im konkreten Text auftreten. Diese Wörter sind sehr oft fachsprachlich, gehören hauptsächlich zu den Autosemantika, meist zu den Wortarten Substantiv oder Adjektiv, und sind auf jeden Fall gute Kandidaten für die Beschreibung, die spezifische Terminologie des Dokumentes, um es vor anderen auszuzeichnen. Wie die Überraschung (Signifikanz) gemessen wird, ist in Abschnitt 3.1.1 beschrieben. Eine genauere Analyse fachsprachlicher Termini und der Extraktion dokumentspezifischer Wörter findet sich in [33]. Dort wurde auch die hier verwendete Terminologieextraktion entwickelt.

Das Zipfsche Gesetz und dessen Folgerungen gelten nicht nur für ein Dokument, sondern für Text allgemein und damit auch für Text, der über mehrere Dokumente verteilt ist, also die Zusammenfassung von Dokumenten zu einem großen Text. Wenn somit mehrere spezielle Dokumente eines allgemeineren Themas zusammen analysiert werden, sollten ihnen allen die Wörter, die das allgemeinere Thema beschreiben, gemeinsam sein, und es sollten diese Wörter auch häufiger, als aufgrund des gesamtsprachlichen Durchschnitts zu erwarten wäre, auftreten.

Somit ist davon auszugehen, daß durch die statistische Analyse nicht nur die ganz speziellen Wörter eines Dokumentes gefunden werden, sondern auch Wörter, die das Dokument mit anders spezialisierten Dokumenten desselben Sachgebiets teilt. Beim Vorgehen von *Makagonov* et al. [18], um ganz dokumentspezifische Wörter herauszufiltern, wurden deshalb die Kanditatenwörter nicht nur mit einem allgemeinsprachlichen Frequenzwörterbuch, sondern auch mit einem Frequenzwörterbuch, welches das weitere Fachgebiet abdeckte, gefiltert (siehe Abschnitt 3.2.1).

Es ist allerdings nicht zu erwarten, daß die beschriebenen statistischen Eigenschaften für alle Texte gleichstark ausgeprägt sind. So wird die Signifikanz von überdurchschnittlich auftretenden Wörtern in fachsprachlichen Texten, wie z. B.wissenschaftlichen Veröffentlichungen, weitaus höher sein, als in allgemeinsprachlichen Alltagstexten, wie sie z. B. in der Kategorie „Vermischtes“ in einer Tageszeitung zu finden sind. Somit ist bei der Evaluation der Verfahren nicht nur der Schwellwert für die Signifikanz zu variieren, sondern auch die Textsorte (siehe Abschn. 5.2).

2.2.4 Linguistische Eigenschaften

Bei der Betrachtung von Text auf Wortebene sind verschiedene linguistische Effekte zu berücksichtigen.

Auf der einen Seite gibt es linguistische Relationen wie das syntagmatische und das paradigmatische Auftreten von Wörtern. Zwei Wörter stehen in syntagmatischer Relation, wenn sie nebeneinander auftreten. Interessant sind diese gemeinsamen Auftreten, wie sie statistisch signifikant, also nicht nur zufällig sind. Statistisch-syntagmatisch auftretende Wörter sind Kandidaten für Mehrwortbegriffe (s. u.) [14, S. 20ff]. Zwei Wörter stehen in paradigmatischer Relation, wenn sie im selben Kontext auftreten. Ist das Auftreten im selben Kontext statistisch signifikant, spricht man von statistisch-paradigmatisch auftretenden Wörtern. Diese gehören oft zu einem gemeinsamen Wortfeld und sind Kandidaten für verschiedene semantische Relationen wie Synonymie und Hyponymie [14, S. 39ff], [33, S. 27ff.].

Über diese Relationen ist die Extraktion semantischer Zusammenhänge, wie sie auch für den Aufbau von Ontologien verwendet werden, aus einer großen Dokumentenkollektion möglich. Das würde aber den Rahmen dieser Arbeit sprengen, sodaß die Extraktion dieser Relationen zum Aufbau der thematischen Struktur nicht in Frage kommt.

Auf der anderen Seite gibt es die linguistischen Phänomene der Polysemie, Synonymie, Mehrwortbegriffe und Morphologie, die auf die Bedeutung eines Wortes im Text Einfluß haben, zu berücksichtigen.

Unerkannte Polysemie ist ein schwerwiegendes Problem, wenn sie auftritt, da sie Texte aufgrund des gemeinsamen Wortes als ähnlich erscheinen läßt, die mitunter gar nichts miteinander zu tun haben. Allerdings ist Polysemie umso seltener, je spezialisierter und fachsprachlicher das Vokabular ist [18, 33].

Unerkannte Synonymie mehrerer Wörter ist für die in dieser Arbeit verfolgten Zwecke nicht so hinderlich. Dadurch, daß statt desselben Wortes eine synonymes in einem anderen Dokument vorkommt, sinkt die berechnete Ähnlichkeit aufgrund weniger gemeinsamer Wörter. Wie die Polysemie ist auch Synonymie umso seltener, je spezialisierter und fachsprachlicher das Vokabular ist [18, 33].

Unerkannte Mehrwortbegriffe sind ebenso nicht schwerwiegend, da sie im Zweifel als mehrere Einzelwörter behandelt werden und diese Einzelwörter oftmals ebenso im selben Wortfeld enthalten sind [20, S. 129f].

Bei der Betrachtung des Phänomens der Morphologie liegt im Prinzip die gleiche Perspektive vor wie bei der Synonymie. Wenn die verschiedenen Formen eines flektierten Wortes nicht auf eine gemeinsame Grundform reduziert werden und als verschiedene, unabhängige Wörter aufgefaßt werden, wird ein bestimmter Anteil von Gemeinsamkeiten zwischen Dokumenten verdeckt. Allerdings gibt es im Deutschen im Schnitt mehr verschiedene flektierte Wortformen eines Worters als Synonyme zu diesem Wort, sodaß der Effekt stärker als bei der Synonymie zum Tragen kommt. Auf der anderen Seite ist im Deutschen meist die häufigste Form eines Wortes mit der Grundform identisch, sodaß in jedem Dokument auch mit der Grundform jedes Wortes gerechnet werden kann.[6]

Für die Verarbeitung anderer Sprachen stellt sich dieses Problem natürlich in ganz verschiedenem Maße. Bei isolierenden Sprachen wie dem Englischen ist das Problem geringer als bei stark flektierenden und agglutierenden Sprachen, wo Worte satzwertig sein können. In diesen Fällen kann eine morphologische Zerlegung hilfreich sein.

Bei durch Komposition gebildeten Wörtern gibt es zwei Aspekte. Zum einen ist zu erwarten, daß der Kopf des Kompositums auch als Einzelwort vorkommt; zum anderen stellt ein Kompositum meist einen themengebietsspezifischen Ausdruck dar, der hilft, das Dokument auszuzeichnen.

Die Derivation ist gar kein problematischer Effekt, da durch sie ein neues Wort in einer neuen Wortart entsteht.

2.3 Schlußfolgerungen

Wie aus den Vorbetrachtungen zu möglichen Strukturen in Abschnitt 2.1 erkennbar, wird sich die ordnende Struktur am thematischen Zusammenhang der Dokumente orientieren. Dabei gibt es die Möglichkeit, die Dokumente mithilfe einer vorgegebenen Ontologie wie WordNet in einen thematischen Zusammenhang zu bringen, oder die thematische Hierarchie aus der Dokumentenkollektion zu extrahieren.

Für die Extraktion der dokumentbeschreibenden Eigenschaften sind aus den Vorbetrachtungen folgende Schlußfolgerungen zu ziehen: Die Dokumente werden durch Termvektoren beschrieben, und die Terme werden aufgrund

[6] Zur genaueren Differenzierung zwischen Wörtern und den Formen ihres Auftretens siehe Abschnitt 3.1.1.

einer statistischen Analyse ausgewählt (siehe Abschn. 3.1.1). Dazu werden die Wörter eines Dokumentes mithilfe einer Stopwortliste gefiltert und danach grundformreduziert. Zum Teil wurde noch nach Wortarten gefiltert, um einzuschätzen, ob die Betrachtung von Substantiven ausreicht. Damit könnte der Berechnungsaufwand erheblich gesenkt werden, weil weniger Terme pro Dokumente zu verarbeiten sind.

Nötiges Vorwissen

Für die Berechnung der Signifikanz ist ein Frequenzwörterbuch der jeweiligen Sprache erforderlich, daneben die Stopwortliste und morphologisches Wissen für die Grundformreduktion. Außerdem ein POS-Tagger für die Filterung der Substantive. Ansonsten wurde auf jegliches einzelsprachliche Vorwissen verzichtet, um die mögliche Andwendungsbreite der getesten Verfahren nicht schon im Vorfeld einzuschränken. Wenn es sich zeigt, daß der gewählte Ansatz funktioniert und ein konkretes Anwendungsgebiet vorliegt, können dahingegehend immer noch Optimierungen vorgenommen werden.

Ob das Frequenzwörterbuch der Allgemeinsprache eventuell auch durch die Dokumentkollektion als ganzes ersetzt werden kann, sollte ausprobiert werden (siehe Abschn. 5.2.4). Wenn dann weiterhin auf die Stopwortliste (was mit einem passenden Schwellwert für die Signifikanz möglich ist), die Beschränkung auf Substantive und die Grundformreduktion verzichtet werden kann, liegt ein Verfahren vor, mit dem ohne jedes einzelsprachspezifische Vorwissen Dokumentenkollektionen strukturiert werden können, um z. B. nicht einfach irgendwo mit dem Übersetzen anfangen zu müssen.

Beim WordNet-basierten Ansatz ist natürlich außerdem noch WordNet nötig, was den Einsatz auf englische Dokumente beschränkt.[7]

Ein datengetriebener Ansatz

Wenn auf eine vorgebene Ontologie verzichtet wird, ist der folgende Ansatz denkbar.

Die Dokumente werden mithilfe ihrer Termvektoren semantisch gruppiert. In Abschnitt 3.1.1 werden Verfahren untersucht, die eine derartige Gruppierung leisten.

[7] bzw. auf Sprachen für die ein Pendant existiert (z. B. GermaNet für Deutsch, EuroNet für einige europäische Sprachen)

Da es für die Verteilung von Termen verschiedene Niveaus gibt, jenachdem wie speziell ein Term ist (siehe die Unterscheidung von allgemeinsprachlichem, technischem, sachgebiets- und dokumentspezifischem Vokabular in [18]), ist davon auszugehen, daß die Gruppierung auf verschiedenen Ebenen semantisch sinnvoll erfolgen kann. Durch Iteration der Gruppierung über die Ebenen von unten nach oben wird die thematische Struktur der Dokumentenkollektion aufgedeckt. Um nun aber die Dokumente einer bestimmte Ebene zuzuweisen, müssen zuerst die Terme in dieser Struktur den jeweiligen Ebenen zugewiesen werden. Dabei entsteht eine Termhierarchie (s. u.). Die Dokumentenhierarchie mit derselben Struktur wird gebildet, indem die Dokumente in der Termhierarchie den ihnen ähnlichsten Knoten zugewiesen werden.

Nun können die Terme an den Knoten (der Termhierarchie) gleich auch als Beschreibung der Dokumentenmengen in dem jeweiligen Knoten dienen.

Die Evaluierung erfolgt, indem als Ausgangsdaten statt einer Dokumentenkollektion eine Dokumentenhierarchie genommen wird, mit der die Ergebnishierarchie verglichen werden kann. Die ausführliche Diskussion möglicher Evaluierungen und der passenden Maße findet in Abschnitt 5.1 statt.

Die Termhierarchie

Die Termhierarchie ist eine hierarchische Struktur ähnlich den in Abschnitt 2.1.1 beschriebenen Ontologien. Die Knoten stellen wie bei der prototypbasierten Ontologie Wortmengen dar.

Sie ist somit keine axiomatisch formale Ontologie. Eine Extraktion derer würde über den Rahmen dieser Arbeit weit hinausgehen.[8] Außerdem wird die Informationsfülle im beschriebenen Ansatz gar nicht benötigt.

Die Termhierarchie ist aber auch keine Taxonomie oder prototypbasierte Ontologie. Die Wortmengen stehen nicht in einer strengen IST-EIN- oder IST-TEIL-VON-Relation.

Vielmehr werden die Wörter nach den semantischen Gruppen, die in der Dokumentenkollektion gefunden wurden, grob sortiert, ohne semantische oder linguistische Relationen zu berücksichtigen. Wörter, die in einem Knoten zusammen sind, zeichnen gemeinsam die unter diesem Knoten gruppierten Dokumente vor den anderen Dokumenten aus. Dabei treten die Wörter

[8] Ontologien dieser Art sind nachwievor alle handgemacht.

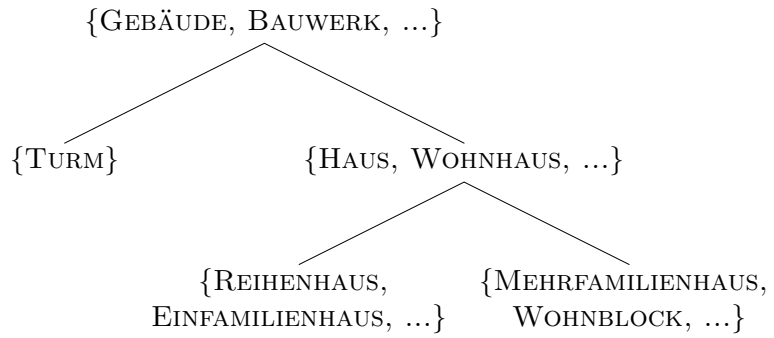

Abbildung 2.3: Ein Beispiel einer Termhierarchie

nicht wie bei der prototypbasierten Ontologie auf jeder Ebene auf, sondern nur auf genau der, wo sie zur Differezierung der Dokumentenmenge dienen. Damit ergibt sich eine Art Abstraktheit für jedes Wort, die sich in der Ebene, auf der das Wort auftaucht, ausdrückt.

Auf diese Weise sind in der Termhierarchie genau die Informationen enthalten, mit denen die gegebene Dokumentenkollektion semantisch strukturiert werden kann. Aus der Perspektive des Trainierens eines Klassifikators oder allgemein eines machine-learning-Systems beschreibt die Wahl der Termhierarchie als Zwischenergebnis eine Einschränkung der Features, die gelernt werden können. Der Vorteil bei dieser Betrachtungsweise liegt für die Termhierarchie darin, menschenlesbar zu sein und auch als Beschreibung der gefundenen Dokumentenmengen dienen zu können.

Ein idealisiertes handgemachtes Beispiel ist in Abbildung 2.3 gegeben. Derartig rein werden die real entstehenden Termhierarchien natürlich nicht sein. Bei den in realistischer Zeit verarbeitbaren Dokumentenmengen dürfte die Statistik nicht gut genug sein. Daneben gibt es auch andere Effekte, die zur Verunreinigung der Termhierarchie führen können. Wenn z. B. bei der Verarbeitung einer Dokumentenkollektion über Logik in einem Text ein Sachverhalt an einem Beispiel erläutert wird, wie beispielsweise logische Schaltungen mithilfe einer Kaffeemaschine, und dabei fachfremdes, d. h. nichtlogisches Vokabular aus dem Kaffeemaschinenbereich verwendet wird, welches aber nur in diesem Beispiel oder ein paar Beispielen in Dokumenten zum selben speziellen Thema und sonst gar nicht vorkommt, dann werden diese Wörter als themenspezifische Wörter der jeweiligen Dokumentengruppe zugeordnet.

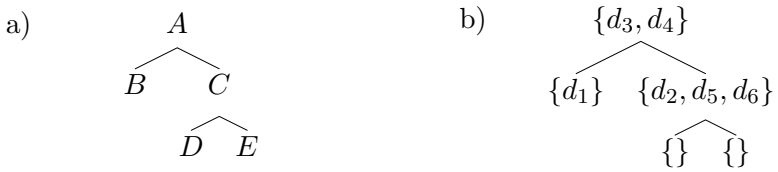

Abbildung 2.4: Beispielthemen (a) und Beispieldokumente (b)

Ein formaler, vorwissenbasierter Ansatz

Unter Verwendung einer gegebenen Ontologie wie WordNet ist folgender Ansatz denkbar.

Die enstehende Struktur der Dokumentkollektion ist monohierarchisch, und die thematischen Beziehungen der Dokumente schlagen sich in der ALLGEMEINER-ALS-Relation, die eine Halbordnung auf den Dokumenten darstellt, und der GENAUSO-ALLGEMEIN-WIE-Relation, die eine Äquivalenzrelation ist, nieder. D. h. für jedes Paar von Dokumenten mit thematisch vergleichbaren, also zum Teil überschneidenden Inhalten muß entschieden werden, welches Dokument allgemeineren Inhalts ist oder ob beide gleichallgemeinen Inhalts sind.

Das folgende Beispiel zeigt, wie eine derartige Entscheidung ausfallen kann. Gegeben sei die thematische Strukturierung der Welt aus Abbildung 2.4a, die Knoten stellen Synsets aus dem WordNet dar und die Kanten die Hyponym-Hyperonym-Beziehung zwischen ihnen. Wenn sich ein Dokument d_1 nur mit Thema B beschäftigt und ein Dokument d_2 nur mit Thema C, so haben sie keinen gemeinsamen Inhalt und sind nicht vergleichbar. Ein Dokument d_3, welches sich mit B und C beschäftigt, ist allgemeiner als d_1 und allgemeiner als d_2 und genauso allgemein wie ein Dokument d_4, welches sich mit A beschäftigt. Ein Dokument d_5, welches sich mit C und D zum Inhalt hat, ist genauso allgemein wie d_2 und wie ein Dokument d_6, welches sich mit C und E beschäftigt.

Daraus ergäbe sich für die Beispieldokumente die in Abbildung 2.4b gezeigte Hierarchie, ohne daß die zugrundeliegende Ontologie hierarchisch strukturiert sein muß.

Dieser Ansatz hat aber ein Problem:

- Es ist nicht klar entscheidbar, ob ein Dokument ein Thema umfaßt oder ob nicht.

Wenn z. B. die o. g. Dokumente d_1 und d_2 ein neues Dokument durch

Verkettung bilden $d_{12} = d_1 \cdot d_2$, würde sich d_{12} mit B und C beschäftigen. Was aber wäre, wenn man nur die Hälfte oder gar nur einen Satz von d_2 an d_1 anfügt? Oder wenn in d_1 Begriffe aus A und C zu Abgrenzung gegenüber diesen vorkommen? Somit können die Dokumente auf diese Weise nicht einfach durch binäre Entscheidungen den Themen zugeordnet werden. Wenn dieser Ansatz weiterverfolgt werden sollte, müßten entsprechende Maß- und Gewichtungsfunktionen entwickelt und begründet werden, die die Dokumente im ersten Schritt den Themen zuordnen, oder statt der Gewichtungsfunktion müßte ein Schwellwert zum Einsatz kommen.

Eine andere Möglichkeit ist, die Dokumente gleich nur dem ähnlichsten Thema zuzuordnen. womit aber der strukturiende Aspekt dieses Ansatz hinfällig wird und einfach die Struktur der zugrundeliegenden Ontologie reproduziert wird. Das entspricht dem im vorigen Abschnitt beschriebenen Vorgehen auf Basis der dort extrahierten Termhierarchie.

Abgesehen davon hat dieser Ansatz noch weitere Nachteile. Zum einen ist man durch die Verwendung von WordNet oder eines seiner Pendants auf einige wenige Sprachen beschränkt. Zum anderen passen handgemachte Ontologien wie WordNet erfahrungsgemäß selten zu den Ergebnissen automatischer Verfahren, wie auch verschiedene handgemachte Ontologien untereinander stark differieren [8]. Daneben enthält WordNet zwar einen großen Teil der englischen Allgemeinsprache, aber viele fachspezifische Termini, welche gerade Dokumente gut beschreiben, nicht. Wegen der dadurch zu erwartenden prinzipiell schlechten Ergebnisse bei einer Evaluierung gegenüber einer ursprünglich zugrundeliegenden Dokumentenhierarchie können keine sinnvollen Aussagen über die Eignung der Verfahren gemacht werden.

Aufgrund dieser Probleme wird dieser Ansatz verworfen.

2.4 Ideen zu Anwendungen

Wie in der Einleitung schon geschrieben, geht es darum, große Dokumentenmengen leichter handhabbar zu machen, indem sie semantisch strukturiert werden.

Diese Handhabbarkeit kann dann ganz direkt mithilfe einer GUI erreicht werden, die das hierarchische Browsen durch die strukturierte Dokumentenkollektion ermöglicht. Dabei können die Terme in der Termhierarchie als zielführende Schlagworte bzw. Linklabels dienen.

Die Handhabbarkeit ist aber auch technisch interpretierbar, indem die Dokumentenhierarchie als Metastruktur über den Dokumenten zur Weiterverarbeitung, z. B. im Rahmen strukturierter Ansätze wie des Semantic Webs, benutzt wird. Auch hier können die Terme in der Termhierarchie als explizit ausgezeichnete Schlagworte bzw. Indexterme dienen.

Die Termhierarchie selber kann aber beispielsweise auch als eine Art Inhaltsverzeichnis der Dokumentenkollektion angesehen werden. Die unter einem Eintrag zu findenden Dokumente sind die, die dem Knoten mit den entsprechenden Termen zugeordnet sind. Wenn man als Dokumente Abschnitte eines Buches benutzt und das Buch vorher passend strukturiert war, sollten die eingesetzten Verfahren diese Struktur möglichst gut reproduzieren.

Kapitel 3

Stand der Forschung

Um die im vorangegangen Kapitel beschriebene Strukturierung einer Dokumentenkollektion unter Zuhilfenahme einer zu extrahierenden Termhierarchie leisten zu können, müssen entsprechende Verfahren gefunden, d.h. aus bestehenden ausgewählt oder neu entwickelt werden. Dafür werden in diesem Kapitel die nötigen analytischen und algorithmischen Grundlagen in Abschnitt 3.1 dargelegt sowie schon bestehende Ansätze zur Strukturierung von Dokumentenmengen und zur Erstellung von Ontologien u. ä. in Abschnitt 3.2 vorgestellt.

3.1 Grundlagen

3.1.1 Allgemeines

Um Dokumente zu verarbeiten, muß geklärt sein, welche Merkmale und Eigenschaften der Dokumente berücksichtigt werden sollen bzw. für die Verarbeitung nötig sind, und wie diese im Modell und als Datenstruktur abgebildet werden sollen.

Beschreibung der Dokumente

Bei den in Rahmen dieser Arbeit gemachten Experimente werden Dokumente nur als Mengen von Wörtern betrachtet. Andere Merkmale, wie z. B. die Auszeichnung bestimmter Wörter als Überschrift, in einer Schlagwortbox am Rand, durch Auftreten innerhalb bestimmter Wortgruppen oder technisch durch das Vorkommen im Dateinamen, die möglicherweise auf eine beson-

dere Bedeutung dieser Wörter für das Dokument schließen lassen, werden nicht berücksichtigt (siehe Kap. 2.2).[9]

Für die Betrachtung der Dokumente als Mengen von Wörtern ist allein die Häufigkeit des Auftretens der einzelnen Wörter im Dokument interessant. Hier sind ein paar Worte zur Klärung der folgenden Verwendung von Begriffen nötig. Ein „Wort“ ist die abtrakte Zusammenfassung all seiner semantisch identischen und durch morphologische und grammatische Effekte verschiedenen Wortformen [14]. Es wird üblicherweise durch die morphologische Grundform aller seiner Wortformen repräsentiert. Die übliche Verwendung von „Wort“ und „Wörter“ folgt allerdings nicht diesem Schema, weswegen die Begriffe „Token“, „Type“ und „Wortform“ der Klarheit halber vorzuziehen sind. Der Terminus „Token“ bezeichnet ein Wort und sein Auftreten an einer bestimmten Stelle. Der Terminus „Type“ bezeichnet ein Wort unabhängig vom Ort, der Häufigkeit und der Form seines Auftretens. Einem Type kann somit eine Frequenz, mit der er innerhalb eines Dokumentes auftritt, zugewiesen werden. In dem Satz *Eines der Häuser ist ein gelbes Haus.* treten die verschiedenen Wortformen *der*, *ein*, *eines*, *gelbes*, *Haus*, *Häuser* und *ist* auf. Die auftretenden Types sind *d–*[10], *ein*, *gelb*, *Haus*, *sein*, wobei die Types *ein* und *Haus* jeweils zweimal, der Rest jeweils einmal vorkommt. Das dritte Token des Satzes ist *Haus*. In einigen Zusammenhängen hat sich die Verwendung des Begriffes „Term“ eingebürgert, z. B. die Dokument-Term-Matrix. Dieser ist in seiner Bedeutung zwar nicht so eindeutig wie die obigen, wird aber hier nur in der Bedeutung „Type“ verwendet, da keine gesonderte Betrachtung verschiedender Wortformen erfolgt. Im Falle möglicher Unklarheiten werden die o.g. unmißverständlichen Begriffe benutzt.

Desweiteren werden auch keine Strukturinformationen aus den Dokumenten, wie die Aufteilung in Abschnitte und Sätze und die daraus resultierenden unterschiedlichen Beziehungen zwischen den Wortformen bzw. Types, berücksichtigt.

Die verarbeiteten Dokumente werden einfach als Mengen der vorkommenden Terme, also Types, und deren Frequenzen bzw. anderer Maßzahlen betrachtet. Diese Maßzahlen heißen Termgewichte.

[9] zu weiteren Möglichkeiten der Merkmalsextraktion siehe [14, S. 217ff]

[10] *d–* ist die genusübergreifende Grundform der definiten Artikel.

Termgewichte

Um das Gewicht bzw. die Bedeutung eines Terms für das Dokument, in dem er vorkommt, zu ermessen, werden in dieser Arbeit folgende Maße betrachtet:

- Frequenz
- tf-idf
- Signifikanz

Die Frequenz ist die Häufigkeit, die Anzahl, mit der der Term im jeweiligen Dokument vorkommt.

Die tf-idf ist ein Maß aus zwei Teilen. tf steht für Termfrequenz. idf steht für inverse Dokumentfrequenz und mißt, in wievielen Dokumenten der Term vorkommt. Die Idee dahinter ist, daß ein Term nicht nur beschreibend für ein Dokument ist, wenn er im selben häufig vorkommt, sondern er zugleich insgesamt selten ist, also das Dokument durch sein Vorkommen besonders auszeichnet.

$$
\begin{aligned}
tf.idf_d(t) &= tf_d(t) * idf(t) \ , \qquad (3.1)\\
\text{mit } tf_d(t) &= \text{ Frequenz des Term } t \text{ im Dokument } d \ ,\\
idf(t) &= \log \frac{|D|}{df(t)} \ ,\\
D &= \text{Menge der Dokumente} \ ,\\
df(t) &= \text{Anzahl der Dokumente, in denen der Term } t \text{ vorkommt} \ .
\end{aligned}
$$

Die Signifikanz eines Termes t im Dokument d ist das Ergebnis eines statistischen Tests, der prüft, wie signifikant das Auftreten von t in d mit $tf_d(t)$ von der zu erwartenden Häufigkeit abweicht. Je höher der Signifikanzwert, desto wahrscheinlicher ist es, daß t nicht nur zufällig so oft in d vorkommt. Für diese Berechnungen gibt es verschiedene Möglichkeiten, die verschiedene (oder gar keine) Annahmen für die zu erwartende Verteilung der Terme über die Dokumente zugrundelegen [33], [20, S. 544ff.]. Um die zu erwartende Häufigkeit einschätzen zu können, wird ein Referenzkorpus benutzt. Die dabei wesentliche Information ist eine Termliste mit den zu den Termen gehörenden beobachteten Anzahlen des Auftretens der Terme, die auf einer großen Dokumentenkollektion basiert und den realen Sprachgebrauch abbildet [33], [14, S. 331f.]. Für verschiedene Zwecke können verschiedene Eigenschaften

dieser Dokumentenkollektion wichtig sein, wie z. B. die Anteile verschiedener Textsorten oder die Beschränkung auf einen bestimmten Zeitabschnitt der Veröffentlichung der Dokumente. In dieser Arbeit wurde mit einem allgemeinsprachlichen Referenzkorpus und mit der zu verarbeitenden Dokumentenkollektion selbst als Referenzkorpus gearbeitet (siehe Abschn. 5.2.4).

Für diese Arbeit wurden die Signifikanzen während der Vorverarbeitung mithilfe der Terminologieextraktion von *Witschel* [33] berechnet (siehe auch Abschn. 4.1). Dabei wird der Likelihood-ratio-Test benutzt, der zu folgender Formel für die Signifikanz führt:

$$\begin{aligned}
\mathrm{sig}(t) = 2\big[&\log L(p_1, k_1, n_1) + \log L(p_2, k_2, n_2) \\
&- \log L(p, k_1, n_1) - \log L(p, k_2, n_2)\big] \,, \qquad (3.2)\\
\text{mit } L(p, k, n) &= p^k (1-p)^{n-k} \,, \\
n_1 &= \text{Häufigkeit von } t \text{ im Dokument } d \,, \\
k_1 &= \text{Länge des Dokuments } d \,, \\
n_2 &= \text{Häufigkeit von } t \text{ im Referenzkorpus} \,, \\
k_2 &= \text{Länge des Referenzkorpus} \,, \\
p_1 &= \frac{k_1}{n_1} \,, \quad p_2 = \frac{k_2}{n_2} \,, \quad p = \frac{k_1 + k_2}{n_1 + n_2} \,.
\end{aligned}$$

Das Vektorraummodell

Um die so, wie in den vorigen beiden Abschnitten beschrieben, dargestellten Dokumente zu verarbeiten, bietet sich das Vektorraummodell an. Das ist ein, vorallem im Information Retrieval, weitverbreitetes Modell zur Analyse von Dokumenten und deren Beziehungen, insbesondere deren Ähnlichkeiten.

Der verwendete Vektorraum wird durch die in den Dokumenten vorkommenden Terme, meist wie hier im Sinne von Types, manchmal auch als Wortformen, aufgespannt. Jeder Term stellt eine Dimension dar. Dokumente werden nun als Vektoren repräsentiert, wobei die Vektorkomponente einer Dimension dem Termgewicht des Termes dieser Dimension entspricht.

Das Beispiel

Dokument 1 = „Hund und Katze“ ,

Dokument 2 = „Biene und Honig“

ergibt mit der Zuordung der Terme zu den Dimensionen in der Reihenfolge des Auftretens und der Frequenz als Termgewicht

$$d_1 = (1, 1, 1, 0, 0) ,$$
$$d_2 = (0, 1, 0, 1, 1) .$$

Die Darstellung kann auch als Dokument-Term-Matrix erfolgen

$$\big(f(d,t)\big)_{d,t} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

Auch möglich ist, den Vektorraum durch die Dokumente aufspannen zu lassen und die Terme als Vektoren aufzufassen:

$$t_1 = (1, 0) , \quad t_2 = (1, 1) , \quad t_3 = (1, 0) , \quad t_4 = (0, 1) , \quad t_5 = (0, 1) .$$

Für die Verarbeitung der so gegebenen Dokumente und Terme können alle Erkenntnisse über Vektorräume und Matrizen verwendet werden. Die entsprechenden Details werden in den Abschnitten zu den einzelnen Verfahren erläutert.

3.1.2 Clustern

Als Clustern bezeichnet man das Gruppieren von Elementen einer Menge, mit dem Ziel, ähnliche Elemente derselben Gruppe (*Cluster*), unähnliche hingegen verschiedenen Clustern zuzuordnen.[11] Dabei muß jedes Element einem Cluster zugeordnet werden, wobei es natürlich in seinem Cluster allein sein kann. Im Gegensatz zur Klassifikation (siehe Abschn. 3.1.4) ist über die Cluster im voraus nichts bekannt.

Im einfachsten Fall kann eine Element zu einem oder mehreren Clustern gehören. Die Cluster sind dann einfach Teilmengen der Gesamtmenge. D.h. für eine Gesamtmenge X wird eine *scharfe* Clusterung (Gruppierung) $\tilde{X}$ gesucht, sodaß

$$\forall x \in X \big[\exists \tilde{x} \in \tilde{X}(x \in \tilde{x})\big] , \quad \bigcup_{\tilde{x}\in\tilde{X}} \tilde{x} = X . \tag{3.3}$$

Sind die $\tilde{x}_i, \tilde{x}_j \in \tilde{X}$ disjunkt, heißt die Clusterung *nichtüberlappend*, sonst *überlappend*.

[11] Eine Gruppe ist hier nicht notwendigerweise eine algebraische Gruppe, sondern z. B. eine Teilmenge der Gesamtmenge.

Die Clusterung kann aber auch *unscharf* sein, d.h. das Ergebnis des Clusterns sind Wahrscheinlichkeiten $p(\tilde{x}|x) \in [0|1]$, mit der $x \in X$ zum Cluster $\tilde{x} \in \tilde{X}$ gehört (mit $\forall x \in X \sum_{\tilde{x}\in\tilde{X}} p(\tilde{x}|x) = 1$).

Wesentlich für das Clustern ist, daß die Elemente eines Clusters sich ähnlicher sein sollen als die Elemente verschiedener Cluster. Somit braucht man zum Clustern in jedem Falle eine Ähnlichkeitsfunktion $\mathrm{sim}(x_i, x_j)$ zwischen den zu clusternden Elementen. Üblicherweise ist $\mathrm{sim}(x_i, x_j) \in [0|1]$, wobei $\mathrm{sim}(x_i, x_j) = 1$ bedeutet, daß $x_i = x_j$.

Es kann auch direkt nur eine Abstandsfunktion $d(x_i, x_j)$ zwischen den Elementen von X gegeben sein, da sich mit jeder Abstandsfunktion eine Ähnlichkeitsfunktion definieren läßt (z. B. $\mathrm{sim}(x_i, x_j) = \exp[-d(x_i, x_j)]$ oder $\mathrm{sim}(x_i, x_j) = 1/[1 + d(x_i, x_j)]$). Abgesehen davon basieren die meisten Clusteralgorithmen auf Rangfolgen nach Ähnlichkeit. In diesem Fall nimmt man statt der zwei Ähnlichsten beispielweise die beiden mit dem geringsten Abstand.

Ähnlichkeitsfunktionen

Um nicht nur die zu clusternden Elemente vergleichen zu können, sondern auch Elemente mit Clustern und Cluster untereinander, muß der Definitionsbereich der o. g. sim-Funktion erweitert werden. Dafür und für die Auswahlmöglichkeiten der Clusterverfahren ist es relevant, ob die Elemente in einem absoluten Rahmen positioniert sind (z. B. durch Merkmalsvektoren in einem Vektorraum), oder ob nur die paarweisen Ähnlichkeitswerte der Elemente zur Verfügung stehen (z. B. in Form einer $|X| \times |X|$-Matrix oder eines Graphen mit den Elementen als Knoten und ähnlichkeitsgewichteten Kanten dazwischen).

Paarweise Ähnlichkeitswerte In diesem Fall kann $\mathrm{sim} : \tilde{X} \times \tilde{X} \to \mathbb{R}$ u. a. auf folgende Arten definiert werden:

- complete-link: Zwei Cluster sind sich so ähnlich, wie ihre zwei unähnlichsten Elemente

$$\mathrm{sim}_{\mathrm{cl}}(\tilde{x}_i, \tilde{x}_j) = \min_{x_i\in\tilde{x}_i, x_j\in\tilde{x}_j} \mathrm{sim}(x_i, x_j) \tag{3.4}$$

- single-link: Zwei Cluster sind sich so ähnlich, wie ihre zwei ähnlichsten

Elemente

$$\mathrm{sim}_{\mathrm{sl}}(\tilde{x}_i, \tilde{x}_j) = \max_{x_i \in \tilde{x}_i, x_j \in \tilde{x}_j} \mathrm{sim}(x_i, x_j) \tag{3.5}$$

- average-link: Zwei Cluster sind sich so ähnlich, wie der Durchschnitt der Ähnlichkeiten ihrer Elemente

$$\mathrm{sim}_{\mathrm{al}}(\tilde{x}_i, \tilde{x}_j) = \frac{1}{|\tilde{x}_i||\tilde{x}_j|} \sum_{x_i \in \tilde{x}_i, x_j \in \tilde{x}_j} \mathrm{sim}(x_i, x_j) \tag{3.6}$$

Analog ergeben sich für $\mathrm{sim} : X \times \tilde{X} \to \mathbb{R}$

$$\mathrm{sim}_{\mathrm{cl}}(x_i, \tilde{x}_j) = \min_{x_j \in \tilde{x}_j} \mathrm{sim}(x_i, x_j) \,, \tag{3.7}$$

$$\mathrm{sim}_{\mathrm{sl}}(x_i, \tilde{x}_j) = \max_{x_j \in \tilde{x}_j} \mathrm{sim}(x_i, x_j) \,, \tag{3.8}$$

$$\mathrm{sim}_{\mathrm{al}}(x_i, \tilde{x}_j) = \frac{1}{|\tilde{x}_j|} \sum_{x_j \in \tilde{x}_j} \mathrm{sim}(x_i, x_j) \,. \tag{3.9}$$

Darstellung der Elemente in einem Vektorraum Ist eine Darstellung der Elemente in einem absoluten Raum gegeben, so können auch den Clustern Positionen darin zugewiesen werden (auch zufällig, um z. B. Startpositionen für den k-means (s. u.) zu finden). Außerdem sind dann alle auf diesem Raum zur Verfügung stehenden Ähnlichkeits- und Abstandsfunktionen auf die Elemente und die Cluster anwendbar.

Sind die Dokumente durch Termvektoren im Vektorraummodell (siehe Abschn. 3.1.1) gegeben ($d_i = (t_{1i}, \ldots, t_{|T|i}) \in \mathbb{R}^{|T|}$), können übliche geometrische Abstandmaße wie der Euklidische Abstand

$$d_{\mathrm{Eu}}(d_i, d_j) = \sum_{k=1}^{|T|} (t_{ki} - t_{kj})^2 \tag{3.10}$$

und die Blockdistanz (auch Manhattendistanz oder L_1-Norm)

$$d_{\mathrm{BD}}(d_i, d_j) = \sum_{k=1}^{|T|} |t_{ki} - t_{kj}| \tag{3.11}$$

verwendet werden. Allerdings ist bei semantisch motivierten Termgewichten

der Cosinus als Ähnlichkeitsmaß oft sinnvoller:

$$\mathrm{sim}(d_i, d_j) = \cos(d_i, d_j) = \frac{d_i \cdot d_j}{|d_i||d_j|} \; . \tag{3.12}$$

Bei Bedarf kann das Skalarprodukt $d_i \cdot d_j$ vom Standardskalaprodukt abweichend definiert werden. Wird z. B. nicht die Annahme, daß die Terme alle paarweise orthogonal, also völlig bedeutungsverschieden sind, getroffen, und sind die Ähnlichkeiten der Terme bekannt, bietet sich folgende Definition an:[12]

$$d_1 \cdot d_2 = \sum_{i,j=1}^{|T|} t_{i1} t_{j2} \mathrm{sim}(i, j). \tag{3.13}$$

Um nun ein Element mit einem Cluster oder zwei Cluster untereinander zu vergleichen, wird jedem Cluster $\tilde{d} = \{d_1, \ldots, d_{|\tilde{d}|}\}$ ein Vektor $\vec{\tilde{d}} \in \mathbb{R}^{|T|}$ zugewiesen. Es könnten natürlich auch (3.4)–(3.6) benutzt werden, aber die Berechnung eine Vektors für den Cluster ist meist semantisch treffender und verringert die Berechnungskomplexität für die Ähnlichkeitsfunktion. Der Zentroid ist für die Berechnung des Clustervektors der übliche Ansatz[13]:

$$\vec{\tilde{d}} = \frac{1}{|\tilde{d}|} \sum_{d_i \in \tilde{d}} \vec{d_i} \; . \tag{3.14}$$

Der Cluster kann auch durch die Summe der Elemente repräsentiert werden. Das ist allerdings bei der Wahl des Ähnlichkeitsmaßes zu berücksichtigen.

$$\vec{\tilde{d}} = \sum_{d_i \in \tilde{d}} \vec{d_i} \; . \tag{3.15}$$

Bei den für diese Arbeit implementierten Clusterverfahren wurde diese Repräsentation gewählt. Als Ähnlichkeitsmaß wurde der Cosinus benutzt (siehe Abschn. 4.4).

Andere Darstellungen und Maße Abgesehen von der Repräsentation der Elemente als Vektoren von Merkmalsausprägungen, wie z. B. Dokumente als Termfrequenzvektoren, gibt es natürlich noch viele andere Möglich-

[12] Das übliche Skalarprodukt ist ein Spezialfall davon mit $\mathrm{sim}(i, j) = \delta_{ij}$.

[13] In dieser Formel habe ich die Vektoren ausnahmsweise der Übersichtlichkeit halber mit Pfeilen dargestellt. $\vec{d_i}$ ist der Termvektor vom Dokument d_i.

keiten der Repräsentation und damit auch der Berechnung der Ähnlichkeiten zwischen den Elementen. Beispielsweise werden Dokumente oft auch als Termmengen aufgefaßt und daraufaufbauende Ähnlichkeitsmaße sind das Jaccardmaß und der Dicekoeffizient. Da diese Darstellung der zu clusternden Elemente in dieser Arbeit aber nicht relevant ist, werden sie hier nicht weiter betrachtet. Für weitere Details siehe [14, S. 213ff] und [20, S. 298ff].

In [32] wurde ein Vergleich von Ähnlichkeitsmaßen anhand der als ähnlich eingestuften Elemente durchgeführt und gezeigt, daß die verschiedenen Maße verschiedene Eigenschaften der Verteilungen in den Elementvektoren hervorheben. Hierbei wurden hier Terme und zwar als Vektoren ihrer Kookkurrenzen betrachtet.

Verfahren

Steht die zu verwendende Ähnlichkeitsfunktion fest, kann danach ein Clusterverfahren ausgewählt werden. Hier gibt es nur einen kurzen Überblick, für mehr Details siehe im allgemeinen [14, 20, 28] und die bei den Beispielen angegeben Quellen.

Die hier betrachteten Clusterverfahren lassen sich in mehrere Gruppen aufteilen:

- hierarchisch
- k-means-Familie
- graphbasiert
- Mehrwegeverfahren

Hierarchische Verfahren Die hierarchischen Verfahren basieren auf dem Verschmelzen und Teilen von Clustern und man erhält im Laufe des Verfahrens einen Baum oder einen Wald von Clustern (Dendrogramm), wobei jeder Cluster die Vereinigung der Cluster, die im Baum seine Kindknoten sind, darstellt. Jeder Schnitt durch diesen Graphen, bei dem alle Blätter auf derselben Seite sind, stellt eine scharfe Clusterung im oben definierten Sinne dar. Für das hierarchische Clustern gibt es zwei Vorgehensweisen:

- hierarchisch-agglomeratives Clustern (HAC): Das Verfahren beginnt damit, daß jedes der zu clusternden Elemente als einlementiger Clu-

ster betrachtet wird. Nun werden in jeder Iteration die zwei ähnlichsten Cluster verschmolzen, und zwar solange, bis die Abbruchbedingung erfüllt ist. Das kann eine vorgegebene Clusteranzahl sein oder ein Ähnlichkeitsschwellwert[14], der von keinem verbleibenden Clusterpaar erreicht wird.

- hierarchisch-divisives Clustern (HDC): Dieses Verfahren beginnt mit der Gesamtmenge der zu clusternden Elemente als einem großen Cluster. Nun wird in jeder Iteration derjenige Cluster aufgespalten, der den geringsten inneren Zusammenhalt hat (z. B. der mit der geringsten durchschnittlichen/maximalen/minimalen[15] paarweisen Ähnlichkeit seiner Elemente). Die Teilung des Clusters in, meist zwei, Teile kann mithilfe eines beliebigen anderen Clusterverfahrens erfolgen. Als Abbruchkriterium für das HDC kann eine vorgegebene Clusteranzahl oder ein Schwellwert für den inneren Zusammenhalt dienen.

k-means-Familie Für einen Algorithmus der k-means-Familie ist es notwendig, die Cluster als eigene Elemente in einem Merkmalsraum darstellen zu können. Der Algorithmus startet mit k im Elementraum plazierten Clusterzentroiden, die entweder zufällig festgelegt werden oder sich aus einer initialen Clusterung der Elemente ergeben. Die Position der Zerntroiden und die Zuordnung der Elemente zu diesen wird iterativ optimiert. In jeder Iteration werden erst alle zu clusternden Elemente dem Cluster des nächstliegenden Zentroiden zugeordnet und dann die Positionen der Clusterzentroiden aus den dem Cluster nun zugeordeten Elementen neu berechnet. Als Abbruchbedingung kann man z. B. die Bewegungen der Zentrioden messen oder zählen, wieviele Elemente in der letzten Iteration ihren Cluster gewechselt haben. Das Ergebnis ist eine scharfe Clusterung.

Graphbasiert Graphbasierte Verfahren legen nicht eine Repräsentation der zu clusternden Elemente in einem Merkmalsraum zugrunde, sondern

[14] Bei genauer Betrachtung müßte es statt „Schwellwert“ „Schwellenwert“ heißen, da eine nicht zu über- oder unterschreitende Schwelle (z.B. Der Deich hält nur bis zum Wasserstand von 10 m.) und nicht, wie stark etwas anschwillt (z.B. Bei einer konstanten Wasserstandserhöhung von 5mm/min hält der Deich noch zwölf Stunden.), angegeben wird. Allerdings ist „Schwellwert“ im Fachsprachgebrauch üblich und wird somit auch in dieser Arbeit verwendet.

[15] analog den oben vorgestellten Maßen average,- complete- und single-link zur Zusammenfassung zweier Cluster

betrachten die Elemente als Knoten eines Graphen. Die Kanten und deren Gewichte in diesem Graphen ergeben sich aus den paarweisen Beziehungen der Elemente, üblicherweise aus deren Ähnlichkeit. Dabei muß nicht jedes Element mit jedem verglichen worden oder überhaupt vergleichbar sein. Ist die Ähnlichkeit zweier Elemente 0 oder unbekannt, so existiert zwischen den sie repräsentierenden Knoten keine Kante.

Graph-factorization Clustering basiert auf der Idee, zu dem zu clusternden Graphen einen bipartiten Graphen zu finden, indem die eine Partition die zu clusternden Elemente, die andere Partition die Cluster, zu denen die Elemente zugeordnet werden, als Knoten enthält [35]. Iteriert und damit hierarchisiert werden kann dieses Clusterverfahren, indem zwischen den Clusterknoten des bipartiten Graphen Kanten eingefügt werden, deren Gewicht sich aus den verschieden Möglichkeiten, über einen Knoten der anderen Partition, also über ein geclustertes Element, von einem Clusterknoten zum anderen zu kommen, ergibt. Für die nächste Iteration wird nur die Partition der Clusterknoten und mit ihren innere Kanten betrachtet. Die entstehende Clusterung ist unscharf.

Ein anderer graphbasierter Clusteralgorithmus ist Chinese Whipsers [5]. Er basiert auf der Ausbreitung und gegenseitigen Verdrängung der den Knoten zugeordneten Clustern, indem gemessen wird, welcher Cluster in der Nachbarschaft eines Knotens überwiegt. Anfangs wird jedem Knoten ein eigener Cluster zugewiesen. Dann wird in zufälliger Reihenfolge für jeden Knoten entschieden, welchem Cluster von denen, denen seine Nachbarknoten zugeordnet sind, er zugewiesen wird. Die Cluster der Nachbarknoten üben dabei über die Kanten und deren Gewichte einen meß- und vergleichbaren Einfluß aus, wobei der stärkste gewinnt. Eine Hierarchisierung ist über die Zusammenfassung von zusammengeclusterten Knoten zu neuen Knoten und Berechnung der Kanten zwischen diesen neuen Knoten möglich. Die entstehende Clusterung ist wahlweise scharf oder unscharf.

Auch andere Verfahren wie z. B. HAC sind graphbasiert implementierbar.

Mehrwegeverfahren Die Mehrwegeverfahren sind informationstheoretische Clusteransätze. Sie betrachten das Auftreten der Merkmale der zu clusternden Elemente und der Elemente selber als Zufallsgrößen. Für das Clustern von Dokumenten stellt die normalisierte Dokument-Term-Matrix die Verbundwahrscheinlickeiten $p(d, t)$ der Zufallsgrößen D und T dar.

Der erste derartige Ansatz war die Information-Bottleneck-Methode, die noch ein „klassisches“ HAC-Verfahren darstellt [31]. Hierbei wird die Transinformation $I(\tilde{D};T)$ zwischen den geclusterten Dokumenten und den Termen maximiert (bzw. umgekehrt, wenn Terme geclustert werden).[16] Die entstehende Clusterung kann wahlweise scharf als auch unscharf sein.

Darauf aufbauend schlagen *Slonim* und *Tishby* ein Zweiwegeverfahren zum Clustern von Dokumenten vor [30]. Zuerst werden die die Terme aufgrund ihrer Kookkurrenzen in den Dokumenten geclustert. Dann werden die Dokumentenvektoren auf Basis der Termfrequenzen durch Vektoren auf Basis der erhaltenen Termcluster und deren Frequenzen ersetzt. Mithilfe dieser Termcluster-Dokument-Matrix werden nun die Dokumente geclustert, wobei die Maximierung von $I(\tilde{D};\tilde{T})$ das Ziel ist. Der Vorteil liegt darin, daß ein Großteil des Rauschens der Dokument-Term-Matrix in der Dokument-Termcluster-Matrix verschwindet und daß letztere deutlich weniger dünn besetzt ist, was den sonst durch Smoothing beabsichtigen Effekten gleichkommt.[17].

Dieses Zweiwegeverfahren kann zu Mehrwegeverfahren verallgemeinert werden, wobei die Clusterung der einzelnen Merkmale gleichzeitig erfolgt [3]. Das ermöglicht, nicht nur die Kookkurrenzen von Dokumenten und Termen, sondern auch weitere möglicherweise vorliegende Merkmale wie z. B. Autor oder Titel und die paarweisen Kookkurrenzen der Merkmale in Form von z. B. Autor-Term-Matrizen zu verarbeiten. Maximiert wird nun die gewichtete Summe der paarweisen Transinformation der Merkmale X_i, X_j

$$\sum_{i,j} w_{ij} I(\tilde{X}_i;\tilde{X}_j) \;, \tag{3.16}$$

wobei w_{ij} das Gewicht der Beziehung zwischen den Merkmalen X_i und X_j darstellt.[18] Es entstehen scharfe Clusterungen.

Vergleiche von Clusterverfahren und Schlußfolgerungen

Die ausgefeilten Verfahren, wie die Mehrwegeverfahren, Graph-factorization-Clustering und auch Chinese Whispers, erreichen bei ihren Evaluierungen

[16] Zu den informationstheoretischen Begriffen siehe Abschn. 5.1.2.

[17] aber eben nicht willkürlich ist

[18] Falls die X_i-X_j-Matrix nicht vorliegt oder kein Zusammenhang zwischen X_i und X_j erwartet wird, ist $w_{ij} = 0$.

vielfach bessere Ergbenisse als die einfacheren Ansätze des HAC, HDC oder k-means.

Allerdings ist damit auch der Aufwand einer komplexeren Implementierung und möglicherweise auch eine geringere Nachvollziehbarkeit der Ergebnisse verbunden (siehe auch [8] und Abschn. 3.2.4).

Da es in dieser Arbeit um eine Machbarkeitsstudie unter Benutzung von möglichst wenig sprachspezifischem Vorwissen geht, liegt als Ansatz eher etwas Einfacheres und Nachvollziehbareres nahe. Deswegen fiel für diese Arbeit die Wahl auf das HAC als Clusteralgorithmus mit dem Cosinus als Ähnlichkeitsmaß. Falls die Idee dieser Arbeit funktioniert, können die Ergebnisse durch die Verwendung eines der ausgefeilteren Verfahren immer noch verbessert werden.[19] Der Einsatz komplexer Verfahren und die mögliche detaillierte Abstimmung und Optimierung der in Kapitel 4 beschriebenen, zusammenarbeitenden Teile für die Strukturierung der Dokumentenkollektion ist erst lohnenswert und mit Gewinn möglich, wenn klar ist, daß das Vorhaben überhaupt funktioniert und welche kritischen Stellen es gibt.

3.1.3 Latente Konzepte

Wie das Clustern ist auch die Extraktion latenter Konzepte geeignet, um Zusammenhänge und Ähnlichkeiten, die zwischen Dokumenten bestehen, zu finden. Auch die formale Herangehensweise gleicht dem Vorgehen beim Clustern: Auf Basis der Dokument-Term-Matrix mit den beobachteten Termgewichten werden die Gewichte berechnet, mit denen ein Dokument bzw. ein Term zu einem Konzept gehören. Die extrahierten Konzepte spannen einen Vektorraum auf, indem sowohl die Terme als auch die Dokumente lokalisiert sind. Diese Zuordnung der Dokumente und Terme zu den latenten Konzepten ist auch als gleichzeitiges unscharfes Clustern der Dokumente und Terme, wie es sich beispielsweise beim Mehrwegeclustern ergibt, interpretierbar, wobei ein Konzept einem Cluster entspricht.

Das erste Verfahren dieser Art war LSA (latent semantic analysis) von *Deerwester* et al. [11]. LSA benutzt die Singulärwertzerlegung der Dokument-Term-Matrix, um die latenten Konzepte als Singulärwerte der Matrix zu extrahieren, die dann auf die vorgegebene Anzahl Konzepte reduziert werden.

[19] Die Implementierung ist dahingehend modular gestalten worden (siehe Abschn. 4.4).

Die Idee der latenten Konzepte weiterführend wurde von *Hofmann* PLSA (probabilistic latent semantic analysis) entwickelt [15]. Sie basiert im Gegensatz zum LSA auf einem stochastischen Aspekt-Modell der Dokumentgenerierung. Damit ist ein realistischeres Modell für die Verteilung der Dokumente und Terme in der Matrix als beim LSA gegeben, was zu besseren Ergebnissen führt.

Diese stochastischen Modelle können weiter verfeinert werden, wie z. B. beim LDA (latent dirichlet allocation) von *Blei* et al. [7] (siehe auch Abschn. 3.2.3).

PLSA

Da das Ziel dieser Arbeit eine Machbarkeitsstudie ist (siehe Kap. 2), lag das Augenmerk nicht auf möglichst spezialisierten oder optimierten Verfahren, sondern im Gegenteil auf allgemeinen, breit einsetzbaren und bereits eingesetzte Verfahren, für die es auch Vergleichsergebnisse und möglicherweise bestehende und bewährte Implementierungen gibt. Im Fall der latenten Konzepte fiel damit die Wahl auf PLSA, bei den Experimenten wurde das PennAspect-Paket als Implementierung von PLSA benutzt [29].

Die Extraktion latenter Konzepte beruht beim PLSA auf einem Aspekt-Modell, welches die Generierung der Dokumente beschreibt. Die Aspekte modellieren als latente Variablen $z_k \in \{z_1, \ldots, z_K\}$ die semantischen Konzepte, die nicht direkt beobachtbar sind. Jedes Dokument und jeder Term kann prinzipiell zu mehreren Konzepten gehören, aber jeder Dokument-Term-Kookkurrenz, also jedem Auftreten (d_i, t_j) wird genau ein Konzept zu geordnet.

Die Generierung der Dokumente mit ihren Termen, also der Dokument-Term-Kookkurrenzen erfolgt so:

1. Ziehe ein Dokument d_i mit $p(d_i)$.
2. Ziehe ein Konzept z_k mit $p(z_k|d_i)$.
3. Ziehe einen Term t_j mit $p(t_j|z_k)$.

mit den Parametern

$p(d_i) =$ Wahrscheinlichkeit für die Beobachtung eines beliebigen Terms im Dokument d_i

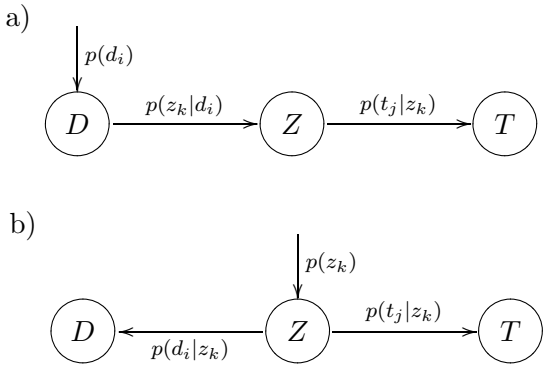

Abbildung 3.1: Die asymmetrische (a) und symmetrische (b) Variante des Aspekt-Modells als Bayessches Netz

$p(z_k|d_i) =$ Wahrscheinlichkeitsverteilung der latenten Konzepte spezifisch für das Dokument d_i

$p(t_j|z_k) =$ Wahrscheinlichkeitsverteilung der Terme in Abhängigkeit vom Konzept z_k

Somit ergibt sich als Wahrscheinlichkeit für eine Kookkurrenz, also das Beobachten des Auftretens eines Terms in einem Dokument (siehe auch Abb. 3.1)

$$p(d_i, t_j) = p(d_i) \sum_{k=1}^{K} p(t_j|z_k) p(z_k|d_i) \; . \tag{3.17}$$

Aus den beobachteten Daten $f(d_i, t_j)$ in der Dokument-Term-Matrix können mithilfe des EM-Algorithmus (expectation maximization algorithm) die wahrscheinlichsten Werte für die Paramter $p(z_k|d_i)$ und $p(t_j|z_k)$ [20], die unter Annahme des beschriebenen Modells zu eben den beobachteten Daten führen, berechnet werden. Dafür ist die folgende Maximum-likelihood-Funktion zu maximieren

$$\mathcal{L} = \sum_{i,j} f(d_i, t_j) \log p(d_i, t_j) \tag{3.18}$$

Für den E-Schritt, indem die Wahrscheinlichkeiten für die Konzepte z_k

[20] $p(d_i)$ muß nicht miteinbezogen werden, da die Dokumentlänge $n(d_i) = \sum_j f(d_i, t_j)$ beobachtbar ist.

aufgrund der aktuellen Parameter berechnet werden, gilt

$$p(z_k|d_i, t_j) = \frac{p(t_j|z_k)p(z_k|d_i)}{\sum_{l=1}^{K} p(t_j|z_l)p(z_l|d_i)} \tag{3.19}$$

Für den M-Schritt der Parameteranpassung für die Maximierung von 3.18 gilt

$$p(t_j|z_k) = \frac{\sum_i f(d_i, t_j)p(z_k|d_i, t_j)}{\sum_{i,m} f(d_i, t_m)p(z_k|d_i, t_m)} , \tag{3.20}$$

$$p(z_k|d_i) = \frac{\sum_j f(d_i, t_j)p(z_k|d_i, t_j)}{n(d_i)} . \tag{3.21}$$

Der E- und der M-Schritt werden abwechselnd ausgeführt, bis die Abbruchbedingung der Iteration erfüllt ist. Das kann entweder ein Konvergenzkriterium sein oder auch eine festgelegte Anzahl von Iterationen.

Die asymmetrische Formulierung des Aspekt-Modells aus (3.17) kann mithilfe des Satzes von Bayes in eine symmetrische mit den Parametern $p(z_k)$, $p(d_i|z_k)$, $p(t_j|z_k)$ überführt werden (siehe auch Abb. 3.1):

$$p(d_i, t_j) = p(d_i) \sum_{k=1}^{K} p(z_k)p(t_j|z_k)p(d_i|z_k) . \tag{3.22}$$

Diese Parametrisierung wird vom eingesetzten PennAspect-Paket verwendet.

3.1.4 Klassifikation

Wie beim Clustern ist die Aufgabe bei der Klassifikation, Elemente zu Clustern/Klassen zuzuordnen. Der Unterschied besteht darin, daß bei der Klassifikation die Klassen schon vorgegeben sind und feststehen. Die Vorgabe besteht nicht nur aus der Anzahl der Klassen, sondern auch aus den Klasseneigenschaften.

Diese Klasseneigenschaften können entweder direkt gegeben sein, z. B. als Merkmalsvektor im selben Vektorraum wie die zu klassifizierenden Elemente, mit dem direkt über ein Ähnlichkeitsmaß das Element verglichen werden kann, oder als Beispiele schon vorklassifizierter Elemente, die mit den zu klassifizierenden Elementen verglichen werden oder als Trainingsmengen benutzt werden.

Der Fall, daß nur vorklassifizierte Beispielelemente gegeben sind, tritt im Rahmen dieser Arbeit nicht ein. Zu den möglichen Vorgehensweisen in diesem Fall, wie mithilfe des direkten Vergleichs der zu klassifizierende Elemente mit den Beispielen, z. B. beim k-Nearest-Neighbour-Verfahren (kNN), oder die Verwendung der Beispiele als Trainingsmenge für z. B. Support-Vector-Machines (SVM), Neuronale Netze oder die Erstellung von Entscheidungsbäumen, siehe [17, 20].

In dieser Arbeit sind die Klassen immer durch einen eigenen Merkmalsvektor gegeben ist. Eine Klassifikation erfolgt, wenn aus der extrahierten Termhierarchie die Ergebnisdokumentenhierarchie erstellt werden soll (siehe Kap. 4 und Abschn. 4.5), indem die Dokumente den Knoten zugeordnet werden. Dabei sind die Knoteneigenschaften als Vektor der Terme des jeweiligen Knotens gegeben. Somit können alle Erkenntnisse über die Messung von Ähnlichkeiten zwischen Elementen und Klassen sowie Klassen und Klassen aus dem entsprechenden Bereich beim Clustern übernommen werden (siehe Abschn. 3.1.2).

3.2 Bisherige Ansätze

Zur Extraktion von Ontologien gibt es natürlich schon Ansätze genauso, wie die Strukturierung von Dokumentenkollektionen und die Klassifikation von Dokumenten wie im vorigen Abschnitt beschrieben nicht neu sind. In diesem Abschnitt werden einige bisherige Ansätze vorgestellt.

Der Hauptunterschied zum in dieser Arbeit verfolgten Ziel ist, daß bei diesen Ansätzen die erhaltenen Ontologien selber das Ziel sind, genauso wie bei den im vorigen Abschnitt vorgestellten Verfahren nur die Strukturierung bzw. Klassifikation der Dokumente das Ziel ist. Das Ziel dieser Arbeit ist im Gegensatz dazu die hierarchische Strukturierung einer vorher unstrukturierten Dokumentenkollektion mithilfe der extrahierten Termhierarchie, die zu einer Ontologie die beschriebenen Unterschiede aufweisen wird (siehe Kap. 2).

Abgesehen von diesem externen Unterschied gibt es verschiedene Möglichkeiten, Ontologien zu lernen, die andere Ansätze als die in dieser Arbeit betrachteten verfolgen.

Ontologien können aus Daten aus verschiedenen Quellen extrahiert werden: Datenbanken, Wörterbüchern und Thesauri sowie strukturierten und

unstrukturierten Dokumenten und Dokumentenmengen. Bei der Verarbeitung von unstrukturierten Dokumentenmengen können Wortfolgemuster (Hearst-Patterns) gesucht und daraus die gefragten Relationen extrahiert werden. Das setzt allerdings das sprachspezifische Wissen über die entsprechenden Muster voraus. Die zweite Möglichkeit, die auch in dieser Arbeit angewendet wird, ist Verarbeitung der vorgefundenen Verteilung der Terme über die Dokumente (siehe Abschn. 3.1). Wegen der in Kapitel 2 beschriebenen Einschränkung auf möglichst wenig Vorwissen werden im folgenden nur derartige Ansätze betrachtet. Eine weitergehende Betrachtung von Ansätzen, die auf unstrukturiertem Text basieren, stellt [4] dar.

3.2.1 Ontologien aus hierarchischen Kollektionen

Einen Ansatz zur Ontologieextraktion, der auf hierarchisch kategorisierten Dokumentenkollektionen arbeitet, verwenden *Makagonov* et al. [18]. Die verwendete Dokumentenkollektion besteht aus Abstracts von Papern, die Zeitschriften, Büchern und Konferenzen untergeordnet sind (Abb. 3.2).

Dieses Vorgehen basiert auf der Beobachtung, daß die Terme der Hauptäste, also der oberen Knoten der Ontologie, die die allgemeinen Themengebiete des Sachgebietes beschreiben, mit den Termen in Titeln von Zeitschriften, Büchern und Konferenzen übereinstimmen, wohingegen spezifischere Themen und Konzepte, die in der Ontologie die feineren Verzweigungen, also die unteren Knoten darstellen, mit den Titeln von Veröffentlichungen/Papers bzw. Kapiteln, die die Bestandteile ebenjener Zeitschriften, Konferenzen und Bücher darstellen, einhergehen.

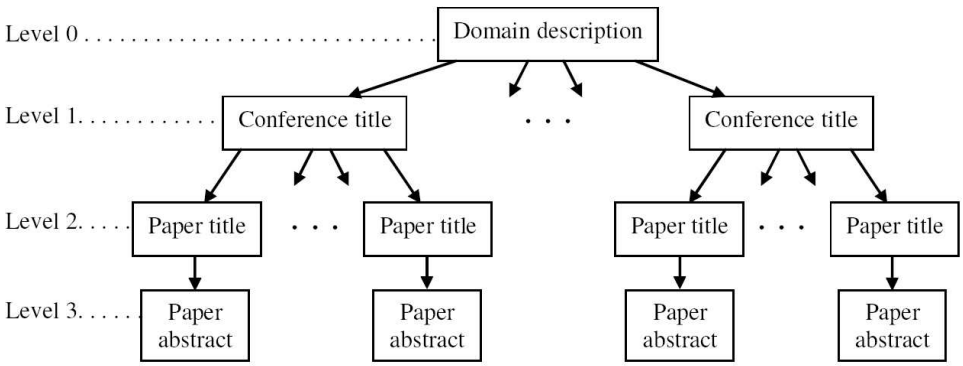

Abbildung 3.2: Struktur der Dokumentenkollektion (Quelle [18])

Eine Ontologie wird hier als eine Polyhierarchie von Termmengen betrachtet. Es wird nicht ausgeschlossen, daß der Algorithmus Knoten mit mehreren Vorgängern produziert. Die angestrebte Hierarchie der Themen und Unterthemen wird direkt aus der Hierarchie des betrachteten Textkorpus' extrahiert (Abb. 3.2), sodaß die Über- und Unterordnung von Wörtern in der Ontologie den Beziehungen der Textsegmente in der Texthierarchie, in denen die Wörter vorkommen, entspricht. Damit ergibt sich, daß die Ebenen der Hierarchie der Textkollektion sich direkt in den Ebenen der Ontologie widerspiegeln.

Für die Erstellung der Ontologie werden die gefundenen Terme in drei Vokabulare eingeteilt: allgemeines, technisches und sachgebietsspezifisches Vokabular, wobei nur Terme aus dem letzteren von Interesse sind. Um Terme der ersten beiden herauszufiltern, wird ein allgemeinsprachliches und ein technisches Referenzkorpus benutzt. Aus den erkannten Termen werden Konzepte gebildet, die Mengen von in einem Dokument signifikant kookkurrienden Termen sind und die Blätter der Ontologie bilden. Die Superkonzepte an den inneren Knoten der Ontologie werden analog aus Konzeptmengen gebildet.

Für den beschriebenen Algorithmus existiert noch keine Evaluierung [18]. Deswegen, und weil das verwendete HTE, was ebenso aus einer hierarchischen Dokumentenkollektion eine Termhierarchie extrahiert (siehe Abschn. 4.2), inhaltlich besser mit den anderen Verfahren vergleichbar ist, wurde dieses Verfahren nicht als inhaltliche obere Baseline verwendet.

3.2.2 Erweiterungen von Ontologien

Es gibt verschiedene Ansätze, existierende Ontologien zu erweitern. Generell ist es nicht ausreichend, nur die Struktur der Ontologie vorgegeben zu haben, sondern es müssen den Knoten auch schon Terme, Konzepte o.ä. zugeordnet sein.

Eine verbreitete Vorgehensweise ist, die Erweiterung als Klassifikation der neuen Terme in die Knoten der Ontologie aufzufassen. Dabei gibt es mehrere Punkte, bei denen sich die Optimierung der Klassifikation in Bezug auf das Aufgabengebiet als gewinnbringend erwiesen hat.

Zum einen können die Klassenbeschreibungen, also die Auswahl und die Kombination der Merkmale der Knoten verbessert werden [24, 25].

Zum anderen können die verwendeten Ähnlichkeitsmaße z. B. durch

Kombination verteilungsbasierter und semantischer Maße verbessert werden [2, 27].

Der Vorgang der reinen Klassifikation kann aber auch z. B. durch die Hinzunahme der Informationen aus der Struktur der Ontologie verbessert werden. Dafür können z. B. für einen Knoten nicht nur seine eigenen Eigenschaften, sondern auch die der ihm untergeordneten Knoten benutzt werden [26]. Das entspricht dem Vorgehen des in dieser Arbeit verwendeten Verfahrens der hierarchischen Termextraktion (HTE), bei dem jeder Knoten auch die Information aller seiner Subknoten enthält und das mit der Benutzung von Informationen, die den zu evaluierenden Verfahren nicht zur Verfügung stehen, als inhaltliche obere Baseline dient (siehe Abschn. 4.2). Daneben können auch die Ähnlichkeitsmaße dahingehend erweitert werden, daß sie auch die Eigenschaften und Ähnlichkeiten benachbarter Knoten miteinbeziehen [27]. Das entspricht der in dieser Arbeit vorgenommenen Anpassung von Evaluationsmaßen an die hierarchische Struktur (siehe Abschn. 5.1.4).

Abgesehen von den beschriebenen Analogien bei der Anpassung des Vorgehens an die hierarchischen Strukturen war der Einsatz eines ontologieerweiternden Verfahrens in dieser Arbeit war nicht möglich, da für die von den verwendeten Dokumentenkollektionen abgedeckten Gebiete keine in struktureller und inhaltlicher Hinsicht passenden Ontologien vorlagen.

Allerdings sollte man diese Verfahren trotzdem nicht generell außer acht lassen, da ein unterstützender Einsatz bei den ontologieextrahierenden Verfahren zu besseren Ergebnissen führen könnte. Beispielsweise ist es denkbar, sich bei der Extraktion der Ontologie auf die relativ sicher einordenbaren Terme zu beschränken, und für die anderen die Vorschläge der Erweiterungsverfahren zu berücksichtigen.

3.2.3 Latente Konzepte

Die in Abschnitt 3.1.3 beschriebenen Verfahren zur Extraktion latenter Konzepte können wie gesagt weiter verfeinert werden, als das beim PLSA der Fall ist. Ein neuer Ansatz besteht darin, den hierarchischen Zusammenhang zwischen den Konzepten in das generative Modell zu integrieren [6]. Damit wird die in Abschnitt 4.3 beschriebene Iteration des Verfahrens, um die einzelnen Ebenen der Hierarchie zu extrahieren, unnötig.

Dieses hierarchische Modell basiert auf dem LDA [7]. Dabei werden die dokumentspezifischen Konzeptverteilungen $p(z|d)$ und die konzeptspezifi-

schen Termverteilungen $p(t|z)$ selbst als Zufallsvariablen aufgefaßt, die in einem vorgeschalteten Schritt erst gezogen, also festgelegt werden müssen. Diese Ziehung folgt einer Dirichletverteilung, die über den Dirichletprozeß erhalten werden kann. Diese Verteilung kann aber auch über den Chinese Restaurant Process (CRP) erhalten werden. Der CRP ist nun hierarchisch erweiterbar, was das Gesamtmodell liefert [6].

Die Erprobung dieses Ansatzes steht noch am Anfang, und es existiert keine zum Vergleich benutzbare Evaluierung sowie keine verfügbare Implementierung. Eine eigene Implementierung hätte den Rahmen dieser Arbeit gesprengt.

3.2.4 Weitere Ansätze

Natürlich gibt es noch viele weitere Arbeiten auf diesem Gebiet. So gibt es Ansätze, die viel syntaktische Information benutzen, um z. B. semantische Relationen zu extrahieren und daraus formale Konzepte, die die Terme beschreiben, zu konstruieren [9]. Diese formalen Konzepte können algebraisch über die Relationen zu einer Ontologie zusammengeführt werden. Dieser Ansatz auf Basis der FCA (formal concept analysis, dt. formale Begriffsanalyse) hat eine Komplexität von $O(2^n)$, aber die damit erreichten Verbesserungen sind gering. In einem Vergleich der FCA mit hierarchisch-agglomerativem und mit Bisection-KMeans-Clustern war die FCA als bestes Verfahren nur 9% besser als die Baseline, bei der alle Terme dem Wurzelknoten zugeordnet wurden [8]. Schon die für die Evaluierung benutzten handgemachten Ontologien wiesen untereinander nur 47%–87% Gemeinsamkeit[21] auf. Die Ergebnisse lassen sich somit nur sehr schwerlich zum Vergleich in dieser Arbeit heranziehen, zumal die verwendeten Maße in dieser Arbeit nicht verwendbar sind, da sie die Identifizierung von Knoten aus den zu vergleichenden Hierarchien miteinander erforden.[22]

Ebenso werden andere Informationen zur möglichen Verbesserung der Ergebnisse herangezogen. *Cimiano* et al. benutzen in [10] Hyperonyme, um das hierarchisch-agglomerative Clustern der Terme auf Terme, die mindestens ein gemeinsames Hyperonym haben, einzuschränken. Die Hyperonyme erhalten sie zum einem aus WordNet, zum anderen extrahieren sie sie

[21] genauer: taxonomic overlap, siehe [8]

[22] Auf die Problematik, geeignete Maße zur Evaluierung zu finden, wird in Abschnitt 5.1 ausführlich eingegangen.

mithilfe von Wortfolgemustern aus einer Dokumentenkollektion. Daneben wurden auch hier synktatische Informationen benutzt, um die Termvektoren auf Basis in Relation stehender Terme zu bilden. Die Evaluierung wurde gegenüber handgemachten Ontologien vorgenommen bzw. wurde von Hand evaluiert. Auch diese Ergebnisse lassen sich somit nur sehr schlecht mit anderen vergleichen.

Kapitel 4

Eingesetzte Verfahren

In dieser Arbeit sollten verschiedene Verfahren zur Strukturierung von Dokumentenkollektionen mithilfe der Extraktion von Termhierarchien erprobt werden. Dafür wurde ein einheitlicher Rahmen geschaffen, in dem die vorangehende Verarbeitung der Rohtexte und die nachfolgende Strukturierung der Dokumente und die Bewertung erfolgt. Grundsätzlich wurde mit einer hierarchisch strukturierten Testkollektion gearbeitet, wobei die Ergebnisstruktur der Dokumente gegen die vorgegebene evaluiert wurde (siehe Kap. 5.1).[23] Im einzelnen wurden die folgenden Verfahren verglichen:

- Hierarchische Termextraktion (HTE)
- iteriertes PLSA
- iteriertes HAC

Dabei dient die Hierarchische Termextraktion als inhaltliche obere Baseline. Sie benutzt als Vorinformation die hierarchische Struktur der Testkollektion, die von den anderen beiden Verfahren nicht verwendet wird. Diese betrachten die Dokumentenkollektion ohne die Strukturinformation und sollen diese erst mithilfe der zu extrahierenden Termhierarchie finden.

Als Testkollektionen wurden zwei echte Dokumentenhierarchien benutzt, sowie eine Hierarchie von synthetischen Dokumenten, die einen angenommenen Idealfall darstellen (siehe Kap. 5.2). Die mit diesen synthetischen Dokumenten gewonnen Ergebnisse stellen eine weitere inhatliche obere Baseline

[23] Eine Term- bzw. Dokumenthierarchie im Sinne dieser Arbeit ist ein graphentheoretischer Baum, dessen Knoten Mengen von Termen bzw. Dokumenten zugeordnet sind, wobei jeder Term bzw. jedes Dokument genau einem Knoten zugeordnet ist (siehe Kap. 2).

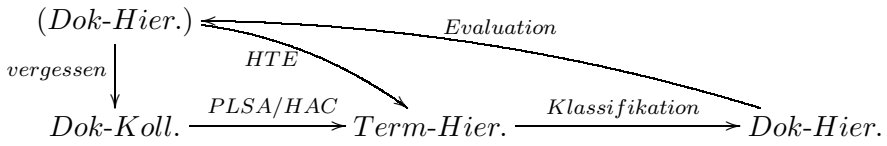

Abbildung 4.1: Der Versuchaufbau

dar. Als untere Baseline wurden eine Zufallshierarchie, in der die Dokumente zufällig auf die Knoten verteilt wurden, und eine „Wurzel“hierarchie, in der alle Dokumente dem Wurzelknoten zugeordnet wurden, verwendet. Der gesamte Versuchaufbau ist in Abb. 4.1 schematisch dargestellt.

Für die zu evaluierenden Verfahren PLSA und HAC wurde die Strukturinformation der vorliegenden, nur zur Evaluation dienenden Dokumentenhierarchie „vergessen“, sodaß diese beiden nur auf einer Dokumentenkollektion als Datenbasis arbeiteten. Daraus extrahierte jedes Verfahren eine Termhierarchie, mithilfe der dann die Dokumente in eine Hierarchie gleicher Struktur wie die extrahierte Termhierarchie einsortiert wurden, was die zu bewertende Dokumentenhierarchie ergab. Die HTE extrahierte die Termhierarchie aus der ursprünglich vorliegenden Dokumentenhierarchie. Die Konstruktion der Ergebnisdokumentenhierarchie erfolgte analog zum Vorgehen bei den beiden anderen Verfahren, ebenso wie die Bewertung.

Für die Versuche mit synthetischen Dokumenten wurde die Ausgangsdokumentenhierarchie auf Basis einer künstlich generierten Termhierarchie konstriert (siehe Abschn. 5.2.3).

4.1 Vorverarbeitung

Für die einzelnen Verfahren müssen die Dokumente als Dokumentvektoren bzw. als Dokument-Term-Matrix vorliegen (siehe 3.1.1). Die Werte in den Vektoren bzw. Matrizen können entweder die Frequenz, die Signifikanz oder die tf-idf der Terme sein. Um diese Daten und Datenstrukturen zu erhalten, werden die Rohtexte vorverarbeitet.

Um die Dokumentenvektoren mit den Frequenzen und den Signifikanzen zu erhalten, wird die Terminologieextraktion von *Witschel* benutzt [33]. Dabei werden alle auftretenden Tokens grundformreduziert. Außerdem findet eine Filterung mithilfe einer Stopwortliste statt. Dabei werden extrem häufige Terme, die in jedem Text vorkommen und somit keinerlei Bezug zum

konkreten Thema eines Dokuments haben, aussortiert und im weiteren nicht berücksichtigt (siehe Abschn. 2.2.3). Keine Berücksichtigung finden außerdem sprachliche Phänomene wie Synonymie, Polysemie und Mehrwortbegriffe. Somit wird angenommen, daß jeder Term eine und nur eine, genau abgegrenzte, zu denen der anderen Terme orthogonale, kontextfreie Bedeutung hat. Das sind natürlich sehr starke Vereinfachungen, die sehr weit von der Realität natürlicher Sprache abstrahieren und nicht erwarten lassen, daß die Ergebnisse davon besser werden. Aber das Ziel dieser Arbeit ist, die prinzipielle Machbarkeit der hierarchischen Strukturierung von Dokumentenkollektionen zu überprüfen, und nicht, die Verfahren für bestimmte Umstände und Modelle zu optimieren. Außerdem wird so ermöglicht, daß die gezeigten Verfahren auch für Dokumentenkollektionen fremder Sprachen eingesetzt werden können, wenn über diese Sprachen fast kein Wissen vorliegt. Experimente, bei denen für die Erstellung der Dokumentvektoren[24] keine Grundformreduktion stattfindet, konnten im Rahmen dieser Arbeit aus Zeitgründen nicht durchgeführt werden, da ohne Grundformreduktion die Anzahl der Types und damit der Berechnunsaufwand deutlich steigt.

Die Signifikanz für das Auftreten eines Terms in einem Dokument wird von der Terminologieextraktion mithilfe des Log-Likelihood-Maßes und eines Referenzkorpus' berechnet. Das Referenkorpus ist im allgemeinen ein allgemeinsprachliches Korpus mit den beobachteten Frequenzen für die darin enthaltenen Terme. Daneben wurde auch die Dokumentenkollektion mit den dortigen Gesamtfrequenzen der Terme als Referenzkorpus benutzt (siehe Abschn. 5.2.4). Für die Erstellung der Dokumentvektoren kommt ein Signifikanzschwellwert zum Einsatz, sodaß nur Terme in den Dokumentvektor aufgenommen werden (auch in den Frequenz- und tf-idf-Vektor), die mit einer bestimmten Mindestsignifikanz im Dokument vorkommen. Diese Mindestsignifikanz wurde auch variiert (siehe Abschn. 5.2.4).

Auf Basis der von der Terminologieextraktion erhaltenen Frequenzvektoren und den Gesamtfrequenzen der Terme in der Dokumentenkollektion wurden dann die tf-idf-Vektoren berechnet,

[24] Der Terminus „Dokumentvektor" bezeichnet dasselbe, wie der „Termvektor", der im vorigen Kapitel eingeführt wurde. Allerdings ist die Perspektive eine andere: Der hier verwendete Termvektor *besteht* aus Termen; der Dokumentvektor *beschreibt* ein Dokument. Da in dieser Arbeit keine termbeschreibenden oder aus Dokumenten bestehenden Vektoren betrachtet werden, ist keine Mehrdeutigkeit zu befürchten und kann die Wortbildungsmächtigkeit des Deutschen für den Ausdruck der Perspektive und des zu betonenden Aspekts benutzt werden.

4.2 Hierarchische Termextraktion (HTE)

Die HTE ist als inhaltliche obere Baseline zum Vergleich für die anderen Verfahren vorgesehen und benutzt auch die Struktur der vorliegenden Dokumentenhierarchie. Die entstehende Termhierarchie hat dieselbe Struktur wie die originale Dokumentenhierarchie, nur sind den Knoten dann Terme statt Dokumente zugeordnet. Das Verfahren arbeitet top-down.

Für jeden Term wird von der Wurzel der Hierarchie ausgehend überprüft, ob er in den Subknoten des aktuellen Knoten hinreichend gleichoft[25] vorkommt, wobei in diesem Sinne ein Knoten die Zusammenfassung all seiner eigenen Dokumente und der Dokumente der Subknoten ist.[26] Ist das der Fall, kommt der Term also in den Subknoten des aktuellen Knotens hinreichend gleichoft vor, wird er dem aktuellen Knoten zugeordnet, andernfalls wird der Subknoten zum aktuellen Knoten, in dem der Term am häufigsten vorkommt. Als Maß, ob der Term hinreichend gleichoft vorkommt, wurde der Variationskoeffizient (relative Standardabweichung) benutzt, da bei einer vorliegenden Gleichverteilung die Standardabweichung gegen 0 gehen sollte:[27]

$$\text{gleichverteilt}(f_1,\ldots,f_k) = \begin{cases} wahr & , \text{VarK}(f_1,...,f_k) < s \\ falsch & , sonst \end{cases} \tag{4.1}$$

$$\text{mit VarK}(X) = \frac{\text{D}(X)}{\text{E}(X)} = \frac{\sqrt{\text{Var}(X)}}{\text{E}(X)} \quad \text{Variationskoeffizient ,}$$

$$X = (f_1,\ldots,f_k) \text{ ,}$$

$$f_i = \text{Frequenz des aktuellen Terms im Knoten } i \text{ ,}$$

$$s = \text{Entscheidungsschwellwert .}$$

Der Entscheidungsschwellwert s ist bei den Experimenten variiert worden, um seinen Einfluß zu ermitteln (siehe Kap. 5.2.2). Wird in den Dokumentvektoren mit einem anderen Termgewicht als der Frequenz gearbeitet, stehen

[25] „Hinreichend gleichoft“ meint hier allgemein „mit hinreichend gleichem Termgewicht“, was nur bei der Frequenz als Maß für das Termgewicht dem üblichen „(hinreichend) gleichoft“ entspricht (siehe Abschn. 3.1.1).

[26] Das ist mit der Zusammenfassung von Subknoten beim tree ascending algorithm vergleichbar [26]. Da hier aber ein anderes Ähnlichkeitsmaß als in [26] verwendet wird, ist die Aufsummierung der Termgewichte unproblematisch.

[27] Wenn als Aussage auch ein Signifkanzniveau nötig ist (oder eine andere Verteilung als die Gleichverteilung erwartet wird), muß ein richtiger statistischer Test wie z. B. der χ^2-Test erfolgen [16, S. 42ff u. S. 61].

die f_i für die entsprechenden Termgewichte des Terms in den Knoten (siehe Abschn. 3.1.1).

Die Idee, die diesem Vorgehen zugrundeliegt, ist, daß ein Term, der in mehreren Dokumenten hinreichend häufig vorkommt, allgemeiner ist als das konkrete Thema des einzelnen Dokumentes und er eher auf die Ebene des die Dokumente zusammenfassenden Knotens gehört. Das gilt auch auf höherer Ebene und tritt somit der Term auch noch in anderen Knoten hinreichend häufig auf, so ist er noch allgemeiner und dementsprechend weiter oben in der Hierarchie anzusiedeln. Tritt der Term jedoch in anderen Knoten nicht so häufig auf, tritt er also in Dokumenten, die anderen Knoten zu- oder untergeordnet sind, nicht so häufig auf, so ist er nicht einem dieser Knoten oder einem gemeinsamen übergeordneten Knoten zuzuordnen, da er nicht für alle Dokumente relevant ist, da er eben nicht in allen Dokumente entsprechend häufig auftritt.

Wenn die Dokumentvektoren nicht Termfrequenzen, sondern -signifikanzen bzw. tf-idf-Werte enthalten, so findet derselbe Test statt, und der eventuell ausgewählte Subknoten ist der mit der höchsten Signifikanz bzw. mit dem höchsten tf-idf-Wert.

4.3 Iteriertes PLSA

Dieses Verfahren wendet das in Abschn. 3.1.3 beschriebene PLSA mehrfach an, um aus der Dokumentenkollektion die einzelnen Ebenen der Hierarchie zu extrahieren. Dafür muß im Vorfeld bekannt sein, wieviele Knoten auf jeder Ebene der Hierarchie existieren sollen, und damit auch, wieviele Ebenen die Hierarchie haben soll.

Das iterierte PLSA arbeitet bottom-up und extrahiert jeweils aus den vorangegangenen Ergebnissen die nächste Ebene. Aus den Ergebnissen der einzelnen PLSA-Läufe wird dann die Hierarchie konstruiert und in ihr die Terme dem richtigen Knoten zugeordnet:

- Iterierte PLSA-Läufe ergeben $\text{Konzept}_{\text{Ebene } i}$-Term-Matrizen und $\text{Konzept}_{\text{Ebene } i}$-$\text{Konzept}_{\text{Ebene } i+1}$-Matrizen.
- Aus den $\text{Konzept}_{\text{Ebene } i}$-$\text{Konzept}_{\text{Ebene } i+1}$-Matrizen wird die Hierarchiestruktur extrahiert.

- In dieser Struktur werden mithilfe der Konzept$_{\text{Ebene } i}$-Term-Matrizen die Terme plaziert.

Ausgehend von der Dokumentenkollektion, konkret der Dokument-Term-Matrix, wird der erste PLSA-Lauf gestartet. Zur Veranschaulichung dient das folgende Beispiel aus 4 Termen, 4 Dokumenten und 2 zu extrahierenden Konzepten. Im allgemeinen Fall entsprechen während der Iteration die Dokumente den Konzepten der Ebene $i+1$, aus denen die Konzepte der Ebene i extrahiert werden[28].

$$\text{Input: } \left(\begin{array}{cccc} f(d_1,t_1) & f(d_1,t_2) & f(d_1,t_3) & f(d_1,t_4) \\ f(d_2,t_1) & f(d_2,t_2) & f(d_2,t_3) & f(d_2,t_4) \\ f(d_3,t_1) & f(d_3,t_2) & f(d_3,t_3) & f(d_3,t_4) \\ f(d_4,t_1) & f(d_4,t_2) & f(d_4,t_3) & f(d_4,t_4) \end{array} \right) \begin{array}{l} \text{Dokument-} \\ \text{Term-Matrix} \end{array} ,$$

$$f(d_i,t_j) = \text{Frequenz der Terms } j \text{ im Dokument } i \text{ .}$$

Das Ergebnis dieses Laufs sind der Konzeptvektor mit den Wahrscheinlichkeiten $p(k)$, daß das Konzept k generiert wird, sowie eine Dokument-Konzept-Matrix und eine Term-Konzept-Matrix, die jeweils die bedingten Wahrscheinlichkeiten $p(d|k)$ und $p(t|k)$, daß ausgehend von einem Konzept k das Dokument d bzw. der Term t generiert wird, enthalten (siehe Abschn. 3.1.3).

$$\text{Output: } \left(\begin{array}{cc} p(d_1|k_1) & p(d_1|k_2) \\ p(d_2|k_1) & p(d_2|k_2) \\ p(d_3|k_1) & p(d_3|k_2) \\ p(d_4|k_1) & p(d_4|k_2) \end{array} \right) \text{ Dokument-Konzept-Matrix ,}$$

$$\left(\begin{array}{cc} p(t_1|k_1) & p(t_1|k_2) \\ p(t_2|k_1) & p(t_2|k_2) \\ p(t_3|k_1) & p(t_3|k_2) \\ p(t_4|k_1) & p(t_4|k_2) \end{array} \right) \text{ Term-Konzept-Matrix ,}$$

$$\left(\begin{array}{c} p(k_1) \\ p(k_2) \end{array} \right) \text{ Konzeptvektor .}$$

Nach der Bayesschen Regel werden jetzt beide Matrizen umgeformt, indem jeder Wert in der Spalte des Konzepts k mit $p(k)$ multipliziert wird, sodaß

[28] Ebene 0 ist die Wurzel.

die Werte nun absolute Wahrscheinlichkeiten $p(d,k)$ und $p(t,k)$ sind:

$$\begin{pmatrix} p(d_1,k_1) & p(d_1,k_2) \\ p(d_2,k_1) & p(d_2,k_2) \\ p(d_3,k_1) & p(d_3,k_2) \\ p(d_4,k_1) & p(d_4,k_2) \end{pmatrix}, \quad \begin{pmatrix} p(t_1,k_1) & p(t_1,k_2) \\ p(t_2,k_1) & p(t_2,k_2) \\ p(t_3,k_1) & p(t_3,k_2) \\ p(t_4,k_1) & p(t_4,k_2) \end{pmatrix}.$$

Außerdem werden beide Matrizen transponiert. Die Werte der Konzept-Term-Matrix werden nun noch mit der Gesamtanzahl Terme in der Dokumentenkollektion multipliziert, um aus den Wahrscheinlichkeiten die Frequenzen zu erhalten, mit denen die Terme in den jeweiligen Konzepten vorkommen. In der Konzept-Subkonzept-Matrix werden jetzt noch die Spalten normalisiert, d.h. jeder Wert in einer Spalte wird mit der Summe der Werte dieser Spalte multipliziert, sodaß diese Matrix jetzt die bedingten Wahrscheinlichkeiten $p(k|d)$, mit der ein Dokument einem Konzept zugehörig ist, enthält:

$$\begin{pmatrix} p(k_1|d_1) & p(k_1|d_2) & p(k_1|d_3) & p(k_1|d_4) \\ p(k_2|d_1) & p(k_2|d_2) & p(k_2|d_3) & p(k_2|d_4) \end{pmatrix} \text{Konzept-Dokument-Matrix},$$

$$\begin{pmatrix} f(k_1,t_1) & f(k_1,t_2) & f(k_1,t_3) & f(k_1,t_4) \\ f(k_2,t_1) & f(k_2,t_2) & f(k_2,t_3) & f(k_2,t_4) \end{pmatrix} \text{Konzept-Term-Matrix}.$$

Diese Konzept-Term-Matrix mit den Frequenzen des Auftretens jedes Terms in einem Konzept bildet analog zur Dokument-Term-Matrix vorher den Input für den nächsten PLSA-Lauf. Aus diesem Lauf erhält man inhaltlich nun eine Term-Superkonzept-Matrix und eine Konzept-Superkonzept-Matrix, mit denen genauso verfahren wird, wie mit den Ergebnissen des ersten Laufs, woraus man die Superkonzept-Term-Matirx und die Superkonzept-Konzept-Matrix, also die Konzept-Subkonzept-Matrix erhält. Diese Iteration läuft über alle Ebenen der Hierarchie, bis die Ebene unter der Wurzel erreicht ist.[29]

Das Ergebnis dieser PLSA-Läufe sind mehrere Konzept-Subkonzept- und Konzept-Term-Zuordnungen, für jede Ebene der zu extrahierenden Hierar-

[29] Die Werte für die Term-Wurzel-Matrix und die Konzept-Wurzel-Matrix sind trivial und werden auch nicht benötigt. Die Term-Wurzel-Matrix ist der Vektor mit den Gesamtfrequenzen der Terme, die Konzept-Wurzel-Matrix ist der Einsvektor. Diese beiden Matrizen werden nicht benötigt, weil der erste Schritt beim Einfügen der Terme in die Hierarchie gleich deren Verteilung in den Subknoten der Wurzel bewertet (siehe übernächster Absatz und Abschn. 4.2).

chie jeweils eine. Aus den Konzept-Subkonzept-Wahrscheinlichkeiten wird die Struktur der Hierarchie konstruiert. Das erfolgt top-down, indem jedes Subkonzept sk einen Knoten bildet und dem Knoten des Konzepts k eine Ebene über ihm zugeordnet wird, für den der Wert $p(k|sk)$[30] maximal ist.

In diese Struktur werden nun die Terme eingefügt. Das erfolgt top-down wie bei der HTE, bloß daß nun die Häufigkeit der Terme in den Subknoten aus der entsprechenden Konzept-Term-Matrix entnommen wird. Der variable Parameter ist wieder der Schwellwert für den Variationskoeffizienten.

Bei diesen Experimenten wurde das PennAspect-Paket als Implementierung von PLSA benutzt [29].

Mittelung mehrerer Läufe

Da PLSA wegen der Initialisierung mit Zufallszahlen nichtdeterministisch ist und bei der Optimierung der Wahrscheinlichkeiten in lokalen Extrema hängen bleiben kann, ist nicht genau abschätzbar, wie gut das Ergebnis eines Laufes ist.

Um das zu kompensieren, kann man mehrere Läufe mit denselben Ausgangsdaten machen, und deren Ergebnisse, also die erhaltenen Matrizen, mitteln.

Da allerdings die Zuordnung der Matrixindizes zu den extrahierten Konzepten auch variiert bzw. in verschiedenen Läufen gar nicht deckungsgleiche Konzepte extrahiert werden, können nicht einfach die verschiedenen Matrizen addiert werden[31], sondern vorher muß man die Zeilen, die wahrscheinlich das gleiche Konzept repräsentieren, identifizieren. „Wahrscheinlich“, weil, wenn unterschiedliche Ergebnisse, deren Defizite kompensiert werden sollen, erwartet werden, die Konzepte sich nicht genau entsprechen werden.

Um die beste oder eine gute Zuordnung zu finden, muß der Permutationsraum durchsucht werden. Bei n Zeilen sind das $n!$ viele Permutation, also definitiv zuviele, um jede zu probieren.

Deswegen wurde auf einen genetischen Algorithmus zurückgegriffen [23]. Die initiale Menge an Permutationen für zwei zu mittelnde Matrizen ergibt sich, wenn man für jede Spalte der zweiten Matrix die Elemente in dieselbe Rangfolge bringt, wie die in den ensprechenden Spalten der ersten Matrix.

[30] entspricht dem $p(k|d)$ nach den Verarbeitungen nach dem ersten Lauf

[31] sonst konvergieren die Konzept-Term- und Konzept-Dokument-Zuordnungen zu Gleichverteilungen

D.h. man stellt den größten Wahrscheinlichkeitswert in der aktuellen Spalte der zweiten Matrix an die Stelle, wo in der ersten Matrix der größte Wert in dieser Spalte steht. Diese Bewegung ergibt einen Teil der aktuellen Permutation für die intiale Permutationsmenge. Der nächste Teil wird von der Bewegung des zweitgrößten Elementes bestimmt. Hat man alle Elemente bewegt, hat man eine Permutation.

$$Spalte_1 = \begin{pmatrix} 0.1 \\ 0.5 \\ 0.3 \\ 0.2 \end{pmatrix} \begin{matrix} 4. \\ 1. \\ 2. \\ 3. \end{matrix}, \; Spalte_2 = \begin{pmatrix} 0.3 \\ 0.1 \\ 0.2 \\ 0.5 \end{pmatrix} \begin{matrix} 2. \\ 4. \\ 3. \\ 1. \end{matrix} \rightarrow \begin{matrix} 1 \text{ Permutation} \\ 3 \text{ um } Spalte_2 \\ 0 \text{ wie } Spalte_1 \\ 2 \text{ zu sortieren} \end{matrix}$$

Für jede Spalte durchgeführt ergibt das die Startmenge der Permutationen.

Die für den Selektionsprozeß nötige Güte einer Permutation ergibt sich aus der Summe der Quadrate aller Elemente der Differenzmatrix aus der ersten Matrix und der zweiten Matrix, nachdem die Zeilen der zweiten Matrix mit der zu bewertenden Permutation permutiert wurden. Von jeder Generation werden nur die n besten betrachtet, wobei n die Anzahl der Initialpermutationen ist. Jeder dieser besten Permutationen rekombiniert sich mit jeder anderen dieser besten zu zwei neuen Permutationen. Für die Rekombination wird zuerst ein Cross-over-Bereich ausgewürfelt. In diesem werden für Kind_1 alle Positionen aus Elter_1 übernommen. Die restlichen Positionen werden mit den restlichen Elementen aus Elter_1 in der Reihenfolge ihres Auftretens in Elter_2 gefüllt. Für Kind_2 wird genauso bloß mit vertauschten Rollen vorgegangen:

Permutation_1	3	1	4	7	5	0	2	6
Permutation_2	7	6	5	4	3	2	1	0
Cross-over		x	x			x	x	
Kind_1	–	1	4	–	–	0	2	–
Kind_1	7	1	4	6	5	0	2	3
Kind_2	7	–	–	4	3	–	–	0
Kind_2	7	1	5	4	3	2	6	0

Es wurden immer 10 Generationen durchlaufen, Mutation erfolgte nicht.

4.4 Iteriertes HAC

Hierarchisch agglomeratives Clustern ergibt von sich aus schon eine Hierarchie von Elementen (siehe Abschn. 3.1.2). Das hier verwendete Verfahren läßt sich auch als ein vollständiger HAC-Lauf bis zu einem Allcluster interpretieren, wobei an bestimmten Stellen Schnappschüsse der momentanen Clusterung gemacht werden bzw. man am Ende nicht das ganze Dendrogramm betrachtet, sondern nur bestimmte Schnitte durch den Clusterbaum. Die Perspektive mehrerer iterierter Läufe ist methodisch begründet. Jeder Lauf des Clusterverfahrens geht nicht bis zum Allcluster, sondern bei einem vorgegebenen Ähnlichkeitsschwellwert oder einer erreichten Clusteranzahl, wie beim PLSA, wird abgebrochen. Die als Ergebnis dieses Laufes erhaltene Clusterung stellt eine Ebene in der zu schaffenden Hierarchie dar und diese Cluster sind gleichzeitig die zu clusternden Elemente des nächsten Laufes, der die nächste Ebene der Hierarchie extrahiert. Außerdem ist bei der Implementierung gleich eine Schnittstelle geschaffen worden, mithilfe derer auch beliebige andere Clusterverfahren verwendet werden können.

Das iterierte HAC arbeitet bottom-up ausgehend von der Gesamtemenge der Dokumente und nimmt jeweils die erhaltenen Cluster des letzten Laufes als neu zu clusternde Menge für den nächsten Lauf:

- iterierte HAC-Läufe ergeben hierarchische Clusterungen von Dokumentmengen
- aus den geschichteten Clusterungen wird die Termhierarchie extrahiert

Für jeden Lauf muß entweder ein Ähnlichkeitsschwellwert für den Abbruch oder wie beim iterierten PLSA die zu erhaltende Anzahl an Ergebnisclustern angegeben werden.

In einem Lauf werden Dokumente/Cluster vereinigt, indem ihre Vektoren addiert werden. Dadurch erhält man im Clustervektor immer die reale Frequenz eines Termes in den Dokumenten dieses Clusters und der Cluster ist sozusagen die Zusammenfassung seiner Dokumente. Damit große Cluster nicht gegenüber kleinen bevorzugt werden, wird als Ähnlichkeitsmaß der Cosinus benutzt (siehe Abschn. 3.1.2).

Da direkt nach den HAC-Läufen schon eine hierarchische Clusterung von Dokumenten vorliegt, wäre man fertig, wenn als Ziel nur die Clusterung von Dokumentenmengen vorgegeben wäre. Allerdings sind in dieser Hierar-

chie alle Dokumente den Blattknoten zugeordnet, d.h. es wird noch nicht zwischen spezielleren und allgemeineren Dokumenten unterschieden. Diese Hierarchie wird ab jetzt Prähierarchie genannt. Um die Dokumente den Knoten in der Hierarchie zuordnen zu können, wird aus der Prähierarchie die Termhierarchie extrahiert.

Die Extraktion der Termhierarchie aus der Dokumentenhierarchie erfolgt wie oben bei der HTE beschrieben. Der variable Parameter ist wieder der Schwellwert für den Variationskoeffizienten.

4.5 Klassifikation der Dokumente

Um aus den von den oben beschriebenen Verfahren erhaltenen Termhierarchien Dokumentenhierarchien zu machen, werden die Dokumente klassifiziert. D.h. die Struktur der Termhierarchie wird direkt übernommen und die Dokumente werden den Knoten zugeordnet. Die Zuordnung erfolgt durch Messung der Ähnlichkeit zwischen dem zu klassifizerenden Dokument und den Knoten, beide repräsentiert durch ihre Termvektoren. Der Termvektor für den Knoten ergibt sich aus der Menge der Terme, die dem Knoten zugeordnet sind, wobei jeder Term das Gewicht 1 hat. Als Ähnlichkeitsmaß dient auch hier der Cosinus (siehe Abschn. 3.1.4).

Kapitel 5

Evaluierung: Experimente und Ergebnisse

In diesem Kapitel wird die Evaluierung der eingesetzten Verfahren beschrieben. Dazu werden im Abschnitt 5.1 die Vorgehensweise bei der Evaluation und die verwendeten Maße erklärt. Im Abschnitt 5.2 werden die Ergebnisse der Experimente und die Wirkung der variierten Parameter behandelt.

5.1 Evaluationsart und -maße

Um etwas zu evaluieren gibt es prinzipiell die folgenden Möglichkeiten:

- direkt
  - absolut
  - relativ
- indirekt

Etwas indirekt zu evaluieren, heißt, die Ergebnisse werden als Grundlage für eine Anwendung benutzt, die sonst auf anderen Daten basiert (oder ganz anders vorgeht). Die Bewertung der Ergebnisse der Anwendung mit den zu evaluierenden Daten, entweder relativ im Vergleich zu den Ergebnissen mit den üblichen Daten bzw. bei der üblichen Vorgehensweise oder absolut, wird als Bewertung der zu evaluierenden Daten genommen.

Eine direkte relative Evaluierung basiert auf dem Vergleich der zu evaluierenden Ergebnisse mit einer Referenz, wenn möglich mit einem allgemein

akzeptierten Goldstandard oder einer anderen hinreichenden „Lösung der Aufgabe“, indem man die Unterschiede zur Referenz bestimmt und mißt.

Eine direkte absolute Evaluierung benutzt nur die Eigenschaften der zu evaluierenden Ergebnisse selber, um daraus deren Güte zu berechnen.

Eine solche direkte absolute Evaluierung der Dokumentenhierarchie könnte z. B. so aussehen, daß, unter Berücksichtigung der Struktur der Hierarchie, eine Art mittlere Homogenität der einzelnen Knoten bezüglich ihrer Dokumente gemessen wird, z. B. über das Ähnlichkeitsmaß zwischen den Dokumenten. Allerdings wäre ein solches Maß schlecht benutzbar, da unbekannt ist, wie die erhaltenen Zahlen einzuschätzen sind. Auch muß das nicht die gewünschten Eigenschaften treffend erfassen, da z. B. Algorithmen wie das hierarchisch-divise Clustern schon bei der Konstruktion der Ergebnisse mit derartigen Maßen arbeiten und trotzdem nicht zwingend immer optimale Ergebnisse liefern. Auch sonst sind absolute Maße für komplexe semantische Probleme und deren Lösungen selten. So gibt es z. B. kein Maß, das angibt, wie gut WordNet ist.

Eine direkte relative Evaluierung der Dokumentenhierarchie ist z. B. möglich, indem die Ergebnishierarchie mit der den Dokumentenkollektionen für die Experimente ursprünglich zugrundeliegenden Hierarchie verglichen wird. Dieser Ansatz bildet die Grundlage für die in dieser Arbeit durchgeführten Evaluierungen. Dabei ist natürlich zu berücksichtigen, daß die als Vergleich herangezogene ursprüngliche Hierarchie keineswegs eine perfekte Vorgabe sein muß, da sie z. B. sehr wahrscheinlich nach anderen Kriterien zustandekam als die hier eingesetzten Verfahren vorgehen.

Schon das Kriterium der semantischen Ähnlichkeit auf Basis der Messung der Ähnlichkeit der Dokumentvektoren deckt sich nicht mit den Entscheidungskriterien, die ein menschlicher Sortierer (unbewußt) benutzt, wenn er Dokumente nach semantischer Ähnlichkeit gruppieren soll. Hierbei spielt auch eine Rolle, daß natürlichsprachliche Phänomene wie Synonymie, Polysemie und Mehrwortbegriffe bei den evaluierten Verfahren nicht berücksichtigt werden (siehe Abschn. 4.1). Die Testkollektion aus den Abschnitten des Buches *Wissensrohstoff Text* [14] (siehe Abschn. 5.2.3) ist z. B. natürlich von den Autoren auch nach semantischen Kriterien strukturiert worden, aber andere Aspekte, wie z. B. eine didaktische Zielstellung, werden die Anordnung der einzelnen Abschnitte und Kapitel ebenso beeinflußt haben.

Eine indirekte Evaluierung der Dokumentenhierarchien wurde nicht in Betracht gezogen, da zum einen die Verzerrungen der Ergebnisse durch die eingesetzte Anwendung nur schwer einzuschätzen sind und zum anderen die möglichen Anwendungen so unterschiedlich sind (siehe Abschn. 2.4), daß die Aussagen einer Evaluation mit nur einer Anwendung nur schwer verallgemeinerbar gewesen wären. Der Aufwand, viele Anwendungen zu benutzen, um verallgemeinerbare Aussagen zu erhalten, ist im Gegensatz zur gewählten direkten relativen Evaluierung nicht vetretbar und auch nur schwerlich abschließend und vollständig möglich.

Die hier durchgeführte direkte relative Evaluierung der Dokumentenhierarchie ist auch als indirekte Evaluierung der Termhierarchie interpretierbar. Die Anwendung ist in diesem Fall die Klassifikation/das semantische Gruppieren der Dokumente.

Eine direkte Evaluierung der im Zwischenschritt entstehenden Termhierarchien erfolgt nicht, da hierfür keine verwendbare Referenz, gegen die evaluiert werden könnte, zur Verfügung steht[32], und ein Maß, was aus der Hierarchie selbst ihre Güte berechnet, ebenso fehlt. Deswegen gibt es wie bereits erwähnt auch kein Maß, welches sagt, wie gut WordNet ist, und somit bietet sich Wordnet auch nicht als Goldstandard an.

Für die beschlossene direkte relative Evaluierung fehlt nun nur noch das Maß, das die Ergebnisse mit der Referenz vergleicht. Wie in Abschnitt 3.2 bei der Vorstellung bisheriger Arbeiten beschrieben, gibt es kein allgemein eingesetztes und für vergleichbare Ergebnisse benutzbares Maß. Deswegen wurden die erhaltenen Hierarchien mit verschiedenen Maßen evaluiert, wobei natürlich die Motivation für die hiesige Verwendung bei jedem Maß eine andere ist, was sich auch in den Ergebnissen und deren Interpretation niederschlägt (siehe Kap. 5.2). Trotz aller Unterschiede der Maße und daraus folgender Vergleichsschwierigkeiten können auf jeden Fall qualitative Urteile über die Ergebnisse durch übereinstimmende Aussagen der verschiedenen Maße besser fundiert werden.

[32] abgesehen von den Vergleichsexperimenten mit synthetischen Dokumenten, wo eine zugrundeliegende Termhierarchie gegeben war

5.1.1 Precision, Recall und F-Wert

Die Maße Precision und Recall sind die üblichen Maße im Information Retrieval, um die Qualität von Antwortmengen auf Suchanfragen zu messen.

$$\text{Precision } P = \frac{|A \cap Z|}{|A|} \,, \tag{5.1}$$

$$\text{Recall } R = \frac{|A \cap Z|}{|Z|} \,, \tag{5.2}$$

$$\text{mit } A = \text{gefundene Items, Anwortmenge} \,,$$
$$Z = \text{relevante Items, Zielmenge} \,.$$

Dabei gibt die Precision die Genauigkeit, der Recall die Vollständigkeit der Antwort an.

Der Evaluationsansatz in dieser Arbeit ist, die erhaltene Dokumentenhierarchie mit der ursprünglichen zu vergleichen, d.h. deren Unterschiede festzustellen und zu messen. Die Unterschiede zwischen beiden Hierarchien bestehen in der unterschiedlichen Zuordnung von Dokumenten zu Knoten. Die Knoten sind allerdings nicht miteinander identifizierbar, d.h. es kann nicht gesagt werden, dieser Knoten in der Ergebnishierarchie entspricht jenem in der ursprünglichen und nun werden die jeweiligen Dokumente an den beiden Knoten verglichen. Sondern das einzige Bekannte, was in der Ergbenishierarchie wiederzufinden ist, sind die Dokumente selbst und eben deren Zusammenfassung zu Mengen an den Knoten. Somit ist von jedem Dokument auszugehen und zu messen, wieviele seiner ursprünglichen Nachbarn in Ausgangsknoten sind jetzt wieder mit ihm im selben Knoten. Damit ergibt sich für die oben eingeführten zu betrachtenden Mengen folgende Festlegung:

$$A = \text{Nachbarn im Knoten der Ergebnishierarchie} \,,$$
$$Z = \text{Nachbarn im Knoten der Ursprungshierarchie} \,. \tag{5.3}$$

Damit wird für jedes Dokumente Precision und Recall berechnet. Das arithmitische Mittel der Precision- und Recallwerte aller Dokumente ergibt Precision und Recall für die Gesamthierarchie.

Diese Maße berücksichtigen allerdings nicht die Struktur der beiden zu vergleichenden Hierarchien. Um das zu erreichen, wurden Anpassungen der beiden Maße vorgenommen (siehe Abschn. 5.1.4).

Da es in vielen Anwendungsbereichen dieser beiden Maße einen Precision-Recall-Zielkonflikt gibt, d.h. die Verbesserung des einen Wertes geht zu Lasten des anderen und umgekehrt, wird oft der F-Wert aus beiden betrachtet:

$$F_\beta = \frac{(1+\beta^2) * P * R}{\beta^2 P + R} \ . \tag{5.4}$$

Der F-Wert gibt die Gesamtgüte der Antwort an und wird nur groß, wenn sowohl die Precision als auch der Recall groß sind. β ist ein Wichtungsfaktor zwischen Precision und Recall. $\beta = 2$ bedeutet, daß der Recall doppelt so wichtig ist wie die Precision, bei $\beta = 0.5$ ist es umgekehrt. Da es keinen Grund gibt, eines der beiden stärker zu gewichten, wurde bei diesen Experimenten immer mit $\beta = 1$ gearbeitet:

$$F = F_1 = \frac{2 * P * R}{P + R} \ . \tag{5.5}$$

5.1.2 Informationstheoretische Maße

Die relative Transinformation und die Varianz der Information sind beides informationstheoretische Maße. Sie basieren auf der Entropiedefinition von Shannon, die für eine gegebene Zufallsgröße X die enthaltene Information in Anzahl Bits, die für die Darstellung mindestens nötig sind, mißt:

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x) \ . \tag{5.6}$$

Für den Logarithmus kann auch eine andere Basis als 2 genommen werden. Qualititiv ändert das die Ergebnisse nicht, nur ist dann die Interpretation mit der Darstellung zur Basis 2 nicht mehr möglich.

Um zu bestimmen, wieviel Information einer Zufallsgröße X in einer anderen Zufallsgröße Y enthalten ist, wird die Transinformation (*mutual information*) benutzt:

$$\begin{aligned} I(X;Y) &= \sum_{x \in X, y \in Y} p(x,y) \log \frac{p(x,y)}{p(y)p(x)} \\ &= \sum_{x \in X, y \in Y} p(x)p(y|x) \log \frac{p(y|x)}{p(y)} \ . \end{aligned} \tag{5.7}$$

Diese gibt an, wieviel Information in X und Y gemeinsam vorliegt, also wie stark die beiden Zufallsgrößen voneinander abhängig sind. Sie steht in direktem Zusammenhang mit der informationstheoretischen Entropie

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) \; . \tag{5.8}$$

Dabei bedeutet

$$I(X;Y) = 0 \; , \tag{5.9}$$

daß X und Y unabhängig sind, und

$$I(X;Y) = H(X) = H(Y) \tag{5.10}$$

bedeutet, daß die gesamte Information von X auch in Y enthalten ist und umgekehrt und somit daß die beiden Zufallsgrößen völlig voneinander abhängig sind.

Für die Berechnung der Entropie einer Hierarchie und der Transinformation zwischen zwei Hierarchien sind die Zufallsgrößen X und Y durch die Anzahl der Elemente (Dokumente bzw. Terme) in den Knoten der Hierarchien gegeben:

$$\begin{aligned} X &= \text{Verteilung der Elemente über die Knoten der Hierarchie} \; , \\ p(x) &= \frac{\text{Anzahl der Elemente im Knoten } x}{\text{Gesamtzahl der Terme}} \\ &= \frac{|x|}{|T|} \; , \end{aligned} \tag{5.11}$$

$$\begin{aligned} p(x,y) &= \frac{\text{Anzahl der gemeinsamen Elemente der Knoten } x \text{ und } y}{\text{Gesamtzahl der Terme}} \\ &= \frac{|x \cap y|}{|T|} \; . \end{aligned} \tag{5.12}$$

Relative Transinformation

Für die Transinformation gilt

$$I(X;Y) \leq \min\{H(X), H(Y)\} \; . \tag{5.13}$$

Um einzuschätzen, wie hoch der relative Anteil der Information, der von Y in X enthalten ist, bildet man die relative Transinformation (entropy purity)

$$rT(X,Y) = \frac{I(X;Y)}{H(Y)} \; . \tag{5.14}$$

Mit (5.13) gilt $0 \leq rT(X,Y) \leq 1$. Nach (5.9) bedeutet $rT(X,Y) = 1$, daß die gesamte Information von Y in X vorliegt, und nach (5.10) bedeutet $rT(X,Y) = 0$, daß gar keine Information von Y in X vorliegt.

Variance of Information

Die Varianz der Information basiert auch auf der Entropie und der Transinformation. Aber sie mißt nicht die Gemeinsamkeit sondern den informationellen Unterschied zwischen X und Y, und sie ist nicht auf den Wertebereich [0|1] normiert, sondern ein echtes Abstandsmaß auf Zufallsgrößen [21, 22]:

$$VI(X,Y) = H(X) + H(Y) - 2I(X;Y) \; . \tag{5.15}$$

5.1.3 Learning Accuracy

Die Learning Accuracy ist ein Maß, um den Abstand zweier Knoten in einem Baum zu messen[33], und wird für die Evaluierung von Ontologien und ähnlichen Hierarchien benutzt [13, 1].

Die Learning Accuracy eines Knotens f bzgl. eines Referenzknotens s wird mithilfe des nächsten Knotens c, der gemeinsamer Oberknoten von f und s ist, berechnet. Dabei sind FP, SP und CP die Pfadlängen von der Wurzel zum entsprechenden Knoten[34] und DP der Abstand zwischen f und c.

$$LA = \begin{cases} 1 & , \quad f = s \\ \frac{CP}{SP} & , \quad f = c \\ \frac{CP}{FP+DP} & , \quad sonst \end{cases} \; . \tag{5.16}$$

Beim Vergleich der Ergebnisdokumentenhierarchie mit der ursprünglichen Dokumentenhierarchie sind die Knoten nicht direkt miteinander identifizierbar. Analog zur Berechnung von Precision und Recall erfolgt deswegen

[33] bzw. sind über Minimumbildung über die verschiedenen möglichen Pfade auch Polyhierarchien möglich, in denen ein Knoten mehrere Oberknoten hat.

[34] als Anzahl besuchter Knoten

die Berechnung der LA für jedes Dokument basierend auf seinen Nachbarn. Man erhält also eine Precision-Learning-Accuracy, indem für jeden Nachbarn des Dokumentes in der Ergebnishierarchie die Learning Accuracy zwischen dem Knoten des Nachbarn in der Referenzhierarchie und dem Knoten des Dokumentes in der Referenzhierarchie berechnet wird. Dasselbe erfolgt in der Ergebnishierarchie für die Knoten der Nachbarn des Dokumentes in der Referenzhierarchie, was eine Recall-Learning-Accuracy ergibt. Der Mittelwert über alle Nachbarn bildet die Precision- bzw. Recall-Learning-Accuracy des Dokumentes, und der Mittelwert über alle Dokumente die Precision- bzw. Recall-Learning-Accuracy der zu evaluierenden Hierarchie. Aus diesen beiden Werten wird die Gesamt-Learning-Accuracy als deren F-Wert analog (5.5) berechnet.

5.1.4 Anpassung der Maße an die hierarchische Struktur

Bis auf die Learning Accuracy berücksichtigen die betrachteten Maße nicht die Struktur der zu vergleichenden Hierarchien. Sowohl Precision und Recall als auch die informationstheoretischen Maße betrachten nur die Dokumente in den jeweils am Vergleich beteiligten Knoten, die in beiden gemeinsam vorhanden sind, und zählen diese. Sie kategorisieren die im (Ziel-)Knoten nichtvorhandenen Dokumente des anderen(/Ausgangs-)Knotens einfach als völlig falsch und zählen sie gar nicht. Dabei spielt es keine Rolle, ob ein Dokument nur leicht falsch dem Nachbarknoten zugeordnet wurde, oder ob der Abstand des Dokumentknotens vom Zielknoten viel größer ist.

Um dieses Manko zu beheben und damit die Maße für die hier betrachteten Hierarchien aussagekräftiger zu machen, wurde für jedes Maß auch eine modifizierte Variante betrachtet.

Die Modifikation erfolgte dahingehend, daß ein Dokument, welches nicht im richtigen Knoten ist, nicht gar keinen Beitrag zum Ergebnis liefert (also 0), sondern sein Beitrag, abhängig von der Position seines (Dokument)Knotens und des Zielknotens, zwischen 0 und 1 liegt. Die hierfür verwendete Gewichtungsfunktion hat folgende Eigenschaften:

- Wenn der Pfad zwischen beiden Knoten über die Wurzel geht, ist das Gewicht 0.
- Ist der Dokumentknoten im Teilbaum unter dem Zielknoten, so ist das Gewicht 1.

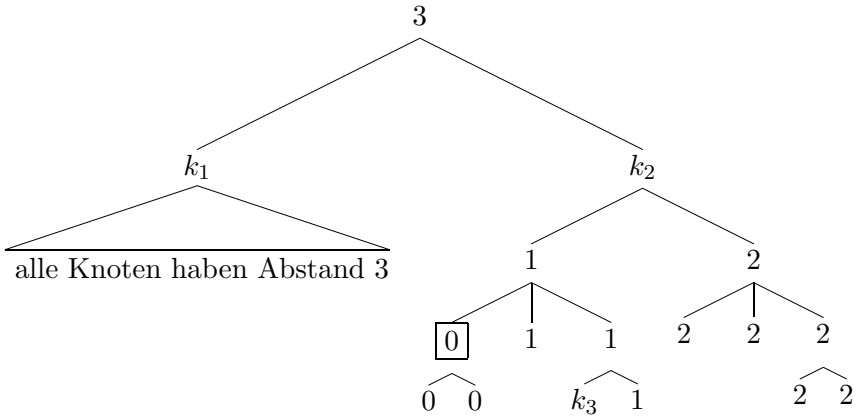

Abbildung 5.1: Beispielwerte der Abstandsfunktion d

Diese beiden Anforderungen sollen sicherstellen, daß die Modifikation möglichst gut die semantischen Eigenschaften abbildet und nicht nur irgendwie die Evaluationszahlen verbessert.

Die erste Anforderung ergibt sich daraus, daß die Wurzel der Hierarchie die sich am stärksten unterscheidenden Teilbäume zusammenfaßt. Wenn ein Dokument in einem anderen Teilbaum unter der Wurzel landet, so ist es maximalfalsch kategorisiert und das schlägt sich im Gewicht 0 nieder.

Ist hingegen das Dokument im Teilbaum unter dem Zielknoten einsortiert worden, so ist ja das richtige Gebiet des Dokumentes gefunden worden, es ist bloß als zu speziell eingestuft worden. Das kann unter anderem daran liegen, daß für das Dokument sehr signifikante Terme in anderen Dokumente nicht oft genug vorkommen und damit diese Terme in der extrahierten Termhierarchie einen eigenen Cluster bzw. Konzept bilden, d. h. einem Knoten sehr weit unten in der Hierarchie zugeordnet werden, dem nun das Dokument zugeordnet wird. Dieser Effekt wird dadurch, daß Synonyme und andere Zusammenhänge zwischen den Termen nicht berücksichtigt, sondern alle Terme als orthogonal angesehen werden, verstärkt (siehe Abschn. 2.2.4). Deshalb ist an der Aussage „Das Dokument beschäftigt sich mit einem Teil des Gebietes des Zielknotens“ nichts falsch, was durch die zweite Anforderung und dem darin geforderten Gewicht 1 repräsentiert wird.

Um das zweite Kriterium zu erfüllen, wird eine entsprechende asymmetrische Abstandsfunktion zwischen den Knoten benutzt: Der Abstand ist die Pfadlänge, wobei nur Kanten nach oben gezählt werden. Abbildung 5.1 veranschaulicht das. Angegeben ist der Abstand des jeweiligen Knotens

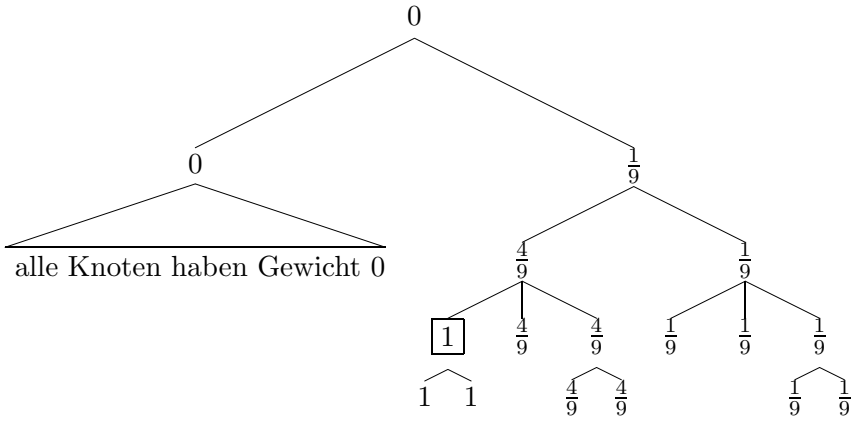

Abbildung 5.2: Beispielwerte der Gewichtsfunktion g

vom Zielknoten $\square$. Für die markierten Beispiele gilt also $d(k_1, \square) = 3$, $d(k_2, \square) = 2$, $d(k_3, \square) = 1$.[35]

Um das erste Kriterium zu erfüllen, berechnet sich nun das Gewicht des Knotens k relativ zum Abstand der Wurzel r vom Zielknoten z:

$$g_z(k) = \left(1 - \frac{d(k,z)}{d(r,z)}\right)^2 . \tag{5.17}$$

Abbildung 5.2 stellt die Gewichte zu dem Beispiel aus Abb. 5.1 dar.

Damit verändern sich die Formeln für Precision und Recall aus (5.1) und (5.2) wie folgt:

$$\text{Precision } P = \frac{1}{|A|} \sum_{d_i \in A} g_{\text{uk}(d)}\big(\text{uk}(d_i)\big) , \tag{5.18}$$

$$\text{Recall } R = \frac{1}{|Z|} \sum_{d_j \in Z} g_{\text{ek}(d)}\big(\text{ek}(d_j)\big) , \tag{5.19}$$

$$\begin{aligned} \text{mit } A &= \text{Nachbarn des Dokument } d \text{ im Knoten} \\ &\quad \text{der Ergebnishierarchie} , \\ Z &= \text{Nachbarn des Dokument } d \text{ im Knoten} \\ &\quad \text{der Ursprungshierarchie} , \\ \text{uk}(d) &= \text{Knoten von } d \text{ in der Ursprungshierarchie} , \\ \text{ek}(d) &= \text{Knoten von } d \text{ in der Ergebnishierarchie} . \end{aligned}$$

[35] Diese Funktion ist asymmetrisch. Z. B.gilt in Abb. 5.1 $d(\square, k_1) = 1$, $d(\square, k_2) = 0$ und $d(\square, k_3) = 2$.

Im Gegensatz zur Learning Accuracy, die mißt, wieweit im Baum die Pfade zu den beiden Knoten übereinstimmen, mißt die hier vorgenommene Modifikation die Abweichung vom als richtig erachteten Knoten. Die modifizierte Precision mißt also eine Art Reinheit, indem sie angibt, aus wieviel Nähe die Dokumente im Zielknoten zusammenkamen. Der modifizierte Recall mißt die Treffgenauigkeit, mit der die Dokumente des Ursprungsknotens dem Zielknoten zugeordnet wurden.

Für den modifizierten F-Wert gilt die übliche Formel (5.4) bzw. (5.5) unter Verwendung der modifizierten Varianten von Precision und Recall.

Bei der Berechnung der informationtheoretischen Maße wird die Formel für die Transinformation (5.7, 5.11) angepaßt. Die Transinformation ist die gewichtete Summe der Transinformationen $\log(p(x,y)/[p(x)p(y)])$ der Knotenpaare. Das Gewicht ist jeweils $p(x,y)$ und wird nicht modifiziert. Modifiziert wird nur die Transinformation der Knotenpaare, indem, statt für $p(x,y)$ nur die gemeinsamen Knotenelemente zu zählen, auch nichtgemeinsame Elemente nach (5.17) analog zu (5.18) etwas beitragen:

$$I(X;Y) = \sum_{x\in X, y\in Y} p(x,y) \log \frac{p_{\text{ext}}(x,y)}{p(y)p(x)} \tag{5.20}$$

$$p_{\text{ext}}(x,y) = \frac{1}{2|T|}\left(\frac{1}{|x|}\sum_{d_i\in x} g_y\big(\mathrm{k}_Y(d_i)\big) + \frac{1}{|y|}\sum_{d_j\in y} g_x\big(\mathrm{k}_X(d_j)\big)\right) ,$$

$$\text{mit } \mathrm{k}_Z(d) = \text{Knoten von } d \text{ in der Hierarchie gegeben durch } Z\ .$$

Der Unterschied der hier gemachten Modifikation zur Learning Accuracy besteht darin, daß die Learning Accuracy den richtigen Anteil des Pfades zum zu bewertetenden Knoten mißt, wohingegen die Modifkation den (gesamten) Abstand der beiden Knoten erfaßt.

5.2 Variierte Parameter und Ergebnisse

Bei den Experimenten zur Evaluation der ausgewählten Verfahren wurden mehrere Parameter, von denen anzunehmen war, daß ihre Werte deutlichen Einfluß auf das Ergebnis haben, variiert.

Die Parameter, die sich aus dem allgemeinen Versuchsaufbau (Abb. 4.1) ergeben, sind in Abbildung 5.3 dargestellt.

Daneben wurde noch die Textsorte der eingesetzten Dokumentenkollek-

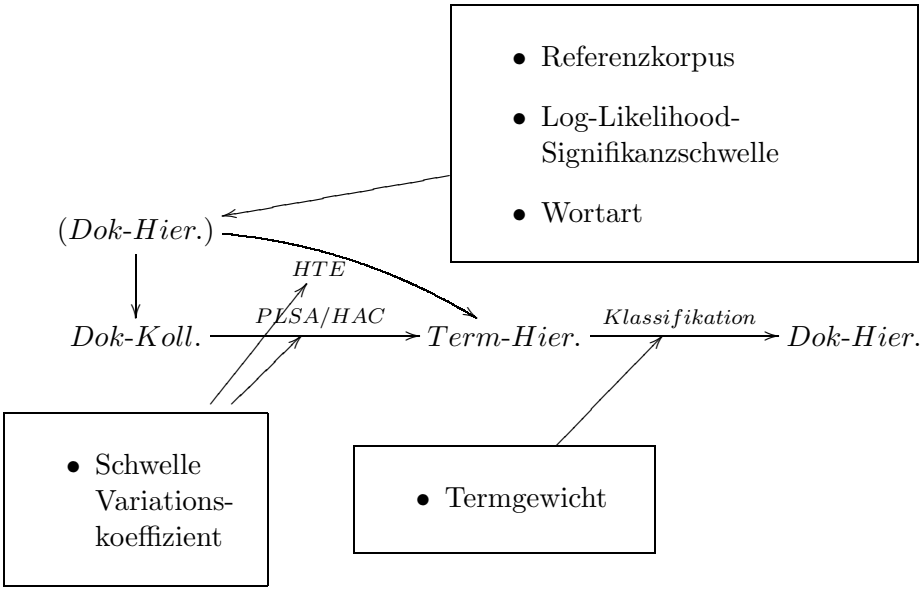

Abbildung 5.3: Variierte Parameter

tion variiert, sowie der Anteil lokaler Terme und die Länge der Dokumente bei den synthetischen Dokumente (siehe Abschn. 5.2.3) und die Anzahl der gemittelten Läufe beim PLSA.

Außerdem kann die Wahl des Maßes als ein Parameter angesehen werden, da im Vorfeld nicht feststand, welches Maß für die hiesige Problemstellung aussagekräftig sein würde (siehe Abschn. 5.1).

Neben dem HTE als oberer Baseline wurden als untere Baselines die Zufallshierarchie und die Wurzelhierarchie verwendet. Bei der Zufallshierarchie wurde für jedes Dokumente der Knoten gleichverteilt ausgewürfelt. Die Wurzelhierarchie besteht nur aus dem Wurzelknoten, dem alle Dokumente zugeordnet sind.

Aufgrund der großen Anzahl und der mehrfachen Referenzierung sind die Ergebnisdiagramme im Anhang A zusammengefaßt dargestellt.

5.2.1 Maß

Da wie gesagt nicht nur ein Maß in Frage kommt, wurden die verschiedenen, im vorigen Abschnitt beschriebenen Maße zur Bewertung eingesetzt. Die Zielstellung dabei war, festzustellen welches Maß sich eignen könnte

- Precision
- Recall
- F-Wert
- relative Transinformation
- Variance of Information
- ext. Precision
- ext. Recall
- ext. F-Wert
- ext. relative Transinformation
- ext. Variance of Information
- Learning Accuracy

Abbildung 5.4: Die eingesetzten Maße

und qualitative Aussagen die von mehreren Maßen untermauert werden als gesicherter ansehen zu können als Aussagen, die nur aus den Ergebnissen eines Maßes ableitbar sind.

Es kamen die in Abbildung 5.4 aufgelisteten Maße zum Einsatz. Die Maße in der rechten Spalte sind die üblichen Maße, die für unabhängigen Mengen gedacht sind. Deswegen wurden sie wie in Abschnitt 5.1.4 beschrieben in Bezug auf die inneren Zusammenhänge zwischen den Knoten der Hierarchie erweitert, was durch das vorgestellte „ext.“, bei den Varianten in der linken Spalte deutlich gemacht wird. Dazu kommt noch die Learning Accuracy als direkt für Hierarchien gedachtes Maß.

Die (ext.) Variance of Information ist ein Abstandsmaß mit dem Wertebereich $[0|\infty[$, wobei kleiner Werte für bessere Ergebnisse stehen. Die anderen Maße sind (Ähnlichkeits)Maße mit Werten in $[0|1]$, wobei größere Werte für bessere Ergebnisse mit größerer Ähnlichkeit zur Referenzhierarchie stehen.

Die Wirkung der Anpassung der Maße an die Hierarchie besteht hauptsächlich darin, daß die evaluierten Verfahren sich je nach Maß mit den ext-Maßen deutlicher von der Zufalls- und der Wurzelhierarchie abheben. Gut sichtbar ist das in den Abbildungen A.13 und A.14 sowie A.35 und A.36, die jeweils die beiden Varianten der Variance of Information zeigen. Bei anderen Maßen ist kein qualitativer Effekt zu beobachten, wie beim F-Wert (z. B. Abb. A.3 und A.4), der Precision (z. B. Abb. A.7 und A.8), dem Recall (z. B. Abb. A.9 und A.10) und der relativen Transinformation (z. B. Abb. A.5 und A.6). Da die Anpassung der Maße an die hierarchische Struktur somit nicht zu unzulässigen Verfälschungen führt, werden im folgenden nur noch die angepaßten Maße betrachtet.

Unter den Maßen qualitativ direkt miteinander vergleichbar sind der (ext.) F-Wert, die (ext.) relative Transinformation, die (ext.) Variance of Information und die Learning Accuracy. Die (ext.) Precision und der (ext.) Recall messen nicht wie die anderen die Gesamtgüte, sondern nur jeweils einen speziellen Aspekt. Der (ext.) F-Wert führt diese beiden zusammen (siehe Abschn. 5.1.1), weswegen sie nur in bestimmten Situationen herangezogen werden müssen. Die anderen Maße stimmen in ihrer qualitativen Aussagen weitgehend überein, was für die Aussagekräftigkeit der Ergebnisse spricht.

Bei der (ext.) relativen Transinformation allerdings wird die Zufallshierarchie überdurchschnittlich gut und meistens sogar besser als die inhaltlichen Verfahren bewertet, was ihre Aussagen schwächt (z. B. Abb. A.6). Auch bei der Learning Accuracy erhalten die Zufalls- und die Wurzelhierarchie relativ hohe Zahlenwerte, allerdings liegen bei ihr die anderen Verfahren darüber[36], wie auch bei der (ext.) Variance of Information und dem (ext.) F-Wert. Diese drei Evaluationsmaße sind somit für die Interpretion der Ergebnisse als maßgebend zu betrachten.

5.2.2 Schwellwert für den Variationskoeffizienten

Der Schwellwert für den Variationskoeffizienten (VarKoeff) bestimmt das Ergebnis maßgeblich, da mit ihm entschieden wird, ob ein Term einem Knoten oder einem seiner Subknoten zugeordnet wird (siehe Abschn. 4.2), und er bei allen Verfahren eine Rolle spielt (siehe Absch. 4.2–4.4).

Ein kleiner Schwellwert für den Variationskoeffizienten bedeutet, bei der Entscheidung, ob ein Term in den Subknoten eines Knotens hinreichend gleichverteilt ist, sehr streng zu sein. Damit werden die Terme eher als nichtgleichverteilt, sondern als speziell angesehen und einem Subknoten zugewiesen, bei dem dieselbe Entscheidung zu treffen ist. Ein sehr kleiner Schwellwert führt somit dazu, daß die Terme sehr wahrscheinlich bis zu den Knoten auf Blattebene der Hierarchie durchgereicht werden.

Da die Dokumente aufgrund der Ähnlichkkeit ihrer Termvektoren zu den Termvektoren der Knoten dem entsprechenden Knoten zugewiesen werden, führt ein sehr kleiner Schwellwert dazu, daß fast alle oder gar alle Dokumente den Blattknoten zugewiesen werden. Damit wird im Ergebnis keine

[36] außer bei den Experimenten mit der Spiegelkollektion, siehe Abschn. 5.2.3

Dokumenthierarchie mehr erhalten, sondern nur noch eine Clusterung der Dokumente. Bei noch kleineren Schwellwerten ändert sich dann nichts mehr.

Ein großer Schwellwert für den Variationskoeffizienten hingegen bedeutet, bei der Entscheidung, ob ein Term in den Subknoten eines Knotens hinreichend gleichverteilt ist, großzügiger zu sein und stärkere Schwankungen in den extrahierten Häufigkeiten eines Termes in den verschiedenen Subknoten eines Knotens zuzulassen. Damit werden die Terme tendenziell den Knoten weiter oben in der Hierarchie zugeordnet. Das gleiche gilt damit auch für die Dokumente, die bei einem sehr großen Schwellwert alle dem Wurzelknoten zugeordnet werden, womit überhaupt keine Strukturierung der Dokumentenkollektion erhalten wird.

Diese beiden Effekte sind daran ablesbar, daß bei allen Maßen für große Schwellwerte die Ergebnisse der verschiedenen Verfahren mit den Ergebnissen der Wurzelhierarchie übereinstimmen. Für kleiner werdende Schwellwerte pendeln sich die Ergebnisse aller Maße auf einem bestimmten Niveau ein, da sich mit immer kleiner werdendem Schwellwert immer weniger an der Ergebnishierarchie ändert (Abb. A.3–A.37).

Ganz deutlich ist die Wirkung dieses Schwellwertes bei den Maßen Precision und Recall zu sehen. Je größer der Schwellwert wird und je weiter oben die Dokumente in der Hierarchie landen, desto größer wird der Recall, da immer mehr der ursprünglichen Nachbarn aus der Referenzhierarchie als Nachbarn in der Ergebnishierarchie wiedergefunden werden (Abb. A.10, A.23 und A.34). Die Precision sinkt währenddessen deutlich (Abb. A.8, A.21 und A.32). Bei kleiner werdendem Schwellwert hingegen sinkt der Recall, während die Precision ihre Maximalwerte erreicht (ebenda).

Als guter Wert, wenn keine weiteren Informationen verfügbar sind, hat sich 1 herausgestellt (Abb. A.3–A.37).

5.2.3 Textsorte

Wie in Abschnitt 2.2.1 schon angesprochen, hat die Textsorte der zu strukturierenden Dokumentenkollektion Einfluß auf die Güte des Ergebnisses. Dokumentmengen mit einem hohen Anteil an eindeutigem, themenspezifischen, anderweitig nicht vorkommenden Vokabular in den Dokumenten, also vorallem Fachtexte mit fachsprachlicher Terminologie, lassen sich leichter strukturieren als solche, wo die Dokumente nicht durch derartige Terme ausgezeichnet sind.

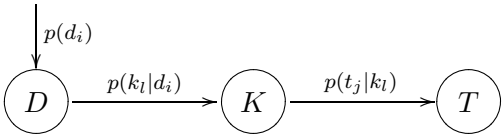

Abbildung 5.5: Das Modell der synthetischen Dokumente als Bayessches Netz

Die beiden für diese Experimente verwendeten natürlichsprachlichen Dokumentenkollektionen waren das Buch *Wissensrohstoff Text* [14] als Fachtext und eine Sammlung von Spiegel-Online-Artikeln, die eher als allgemeinsprachlicher Text einzustufen sind.

Dazu kamen vorher noch Experimente mit synthetische Dokumenten als grundlegender Machbarkeitstest.

Synthetische Dokumente

Als erstes wurden Experimente mit synthetischen Dokumenten gemacht, um festzustellen, wie gut die Verfahren unter Idealbedingung arbeiten. Die synthetischen Dokumente wurden mithilfe eines Modells generiert, was dem Aspekt-Modell des PLSA ähnlich ist.

Modell der synthetischen Dokumente Der Generierung der synthetischen Dokumente liegt folgendes Modell zugrunde. Zuerst wird eine Termhierarchie generiert, die dieselbe graphentheoretische Struktur wie die zu generierende Dokumentehierarchie aufweist. Darauf aufbauend wird die Dokumentenhierarchie erzeugt, indem jedem Dokumenten ein Knoten zugeordnet wird und in Abhängigkeit des Knotens die Terme ausgewählt werden. Die Dokumente werden direkt als Termvektoren erzeugt.

Dieses Vorgehen ist in Abbildung 5.5 als Bayessches Netz dargestellt und entspricht dem Modell, das dem PLSA zugrunde liegt (siehe Abschn. 3.1.3, vgl. Abb. 3.1). Die Knoten der Hierarchioe hier entsprechen den Konzepten beim PLSA. Allerdings sind die Knoten hier nicht unabhängig, sondern stehen über die Hierarchie in einem thematischem Zusammenhang, der benutzt werden kann, um die Wahrscheinlichkeiten $p(t|k)$ für die Terme in Abhängigkeit des Dokumentknotens festzulegen (s. u.).

Als Vorgaben für die Parameter Struktur, Anzahl der Dokumente $|D|$, Anzahl der Terme $|T|$ in der Hierarchie und (durchschnittliche) Anzahl der

Tokens pro Dokument ld wurde die Parameter des WRT-Korpus genommen, um eine möglichst gute Vergleichbarkeit der Ergebnisse zu erreichen.

Da keine Annahmen über die Verteilung von Termen über Konzepten vorlagen, wurden die Terme gleichmäßig verteilt, d. h. jeder Knoten bekam gleichviele Terme und jeder Term wurde nur genau einem Knoten zugewiesen. Dafür wurden die Terme blockweise verteilt:

$$\mathrm{k}(t_j) = j \text{ div } |K| \; . \tag{5.21}$$

Da ebenso keine Annahmen über die Verteilung von Dokumenten über Konzepten vorlagen, wurde die Dokumente ebenso gleichmäßig generiert, d. h. zu jedem Knoten wurde dieselbe Anzahl Dokumente generiert. Jedes Dokument wurde genau einem Knoten zugeordnet. Dieser Knoten ist das Thema des Dokuments.[37]

$$p(k_l|d_i) = \begin{cases} 1 & , \quad l = i \text{ div } |K| \\ 0 & , \quad sonst \end{cases} \; . \tag{5.22}$$

Weil auch kein Wissen über die allgemeine Verteilung der Dokumentlänge über Dokumente vorlag, wurde jedes Dokument gleichlang mit der vorgegebenen durchschnittlichen Anzahl Tokens ld erzeugt.

Bei der Erzeugung der Terme eines Dokumentes wurde ein weiterer Parameter eingeführt: der Anteil der lokalen Terme lt. Dieser Parameter gibt an, welcher Anteil der Tokens nur aus dem Knoten des Dokumentes gezogen wird. Für jedes Dokument werden die ersten $lt * ld$ Tokens gleichverteilt aus den Termen des Knotens des Dokumentes gezogen. Die restlichen Tokens werden wie folgt aus allen Knoten gezogen. Jedem Term wird eine Wahrscheinlichkeit zugeordnet, die vom Knoten des Terms bzw. genauer vom Abstand zwischen diesem Knoten und dem des Dokumentes, für das die Terme gezogen werden, abhängt.

$$\begin{aligned} & p_d(t) = \frac{g_d(t)}{\sum_{t_j \in T} g_d(t_j)} \; , \\ \text{mit} \quad & g_d(t) = \frac{1}{2^{d_d(t)}} \; , \\ & d_d(t) = \text{Anzahl Kanten auf dem Pfad von k}(d) \text{ zu k}(t) \; . \end{aligned} \tag{5.23}$$

[37] Das entspricht der blockweisen Zuteilung der Terme zu den Knoten: $\mathrm{k}(d_i) = i \text{ div } |K|$

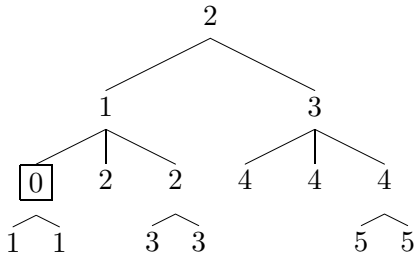

Abbildung 5.6: Beispielwerte der Abstandsfunktion d

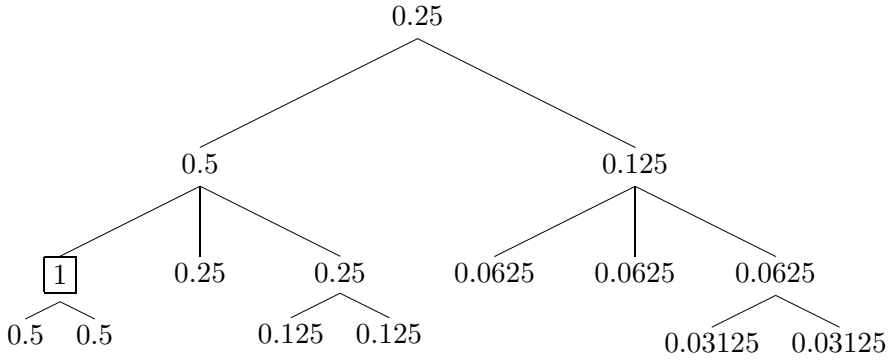

Abbildung 5.7: Beispielwerte der Gewichtsfunktion g

Die Funktionen d und g sind in den Abbildungen 5.6 und 5.7 an einem Beispiel dargestellt.

Der Parameter lt gibt somit an, wie rein bzw. ideal die generierten Dokumente sind, da er beschreibt, wieviele Tokens eines Dokuments garantiert zum Themengebiet des Dokuments gehören. Aber selbst ein $lt = 0$ bedeutet nicht, daß die Dokumente keine Terme aus ihrem Knoten enthalten, sondern diese Terme sind immer noch die wahrscheinlichsten.

WRT

Die Dokumentenkollektion WRT basiert auf dem Buch *Wissensrohstoff Text* [14]. Das Buch ist in 8 Kapitel gegliedert, mit jeweils 3 bis 15 Abschnitten. Als Dokumente wurden die einzelnen Unterabschnitte benutzt, das sind zwischen 2 und 8, im Schnitt 6 pro Abschnitt. Insgesamt bestand die Kollektion aus 210 Dokumenten.

Spiegel

Die Kollektion von Spiegel-Online-Artikeln bestand aus 1000 Artikeln, die jeweils als ein Dokument betrachtet wurden. Die Dokumente verteilten sich auf 10 Kategorien mit je 100 Dokumenten, die gleichmäßig auf mögliche Unterkategorien verteilt waren. Die Kategorien sind in Abbildung A.2 dargestellt.

Ergebnisse

Die Experimente mit den synthetischen Dokumenten verliefen sehr vielversprechend. Schon bei einem vorgegebenen Anteil lokaler Terme von $lt = 0.3$ erreicht das HAC diesselbe Güte wie das HTE als obere Baseline und ab $ld = 0.4$ erreicht auch das PLSA Werte nur wenig unter dem HTE (Abb. A.40, A.42, A.44, A.46, A.48, A.50). Bei kleineren Mindestanteilen lokaler Terme, also bei verrauschteren Dokumente, bricht das HAC stark ein, das PLSA kann damit besser umgehen und bei passendem Variationskoeffizientschwellwert $vk = 1.0$ sein Niveau über den unteren Baselines halten (Abb. A.40 und A.42). Dabei ist gut erkennbar, daß gleichzeitig auch das HTE einbricht, was zeigt, daß die einzelnen Dokumentenmengen, also Themenbereiche, anhand der Terme bei immer verrauschteren Dokumenten bedeutend schwerer zu trennen sind.

Bei größeren Schwellwerten für den Variationskoeffizienten übertrifft das HAC sogar das HTE (Abb. A.44, A.46 und A.27–A.37). Wie das erklärbar ist, wird im nächsten Abschnitt beschrieben.

Da die synthetischen Dokumente mithilfe einer zugrundeliegenden Termhierarchie generiert wurden, konnten auch die im Zwischenschritt der Verfahren extrahierten Termhierarchie direkt gegen diese zugrundeliegende Hierarchie evaluiert werden. Die Ergebnisse spiegeln die Ergebnisse der Dokumenthierarchien wider (Abb. A.41, A.43, A.45, A.47, A.49, A.51).

Die Experimente mit den natürlichsprachlichen Dokumentenkollektionen brachten (qualitativ) die erwarteten Ergebnisse (siehe Abschn. 2.2.1). Die Aufgabe ist deutlich schwieriger als mit idealisierten synthetischen Dokumenten, womit die getesteten Verfahren HAC und PLSA deutlich hinter die oberere Baseline vom HTE, die weiter bei einem F-Wert von rund 0.8 liegt, zurückfallen (Abb. A.4 und A.17). Bei der fachsprachlicheren Dokumentenkollektion (WRT) werden deutlich bessere Ergebnisse erzielt als bei

der eher allgemeinsprachlicheren Dokumentenkollektion (Spiegel).

Wie kann ein uninformiertes Verfahren besser als ein informiertes sein?

Bei der Generierung der Dokumente und der Dokumentenhierarchie aus der Termhierarchie findet eine „Verschleierung“ der Zugehörigkeiten der Terme zu den Knoten statt. Es ist z. B. theoretisch möglich, daß ein Dokument, das einem Knoten („blau“ genannt) zugeordnet ist, übermäßig viele Terme aus einem Knoten oder einer Menge von Knoten in einem Teilbaum („grün“ genannt) bekommt. Bei der Clusterung mittels HAC wird dieses Dokument dann mit den anderen grünen zusammengeclustert. Genauso wie das PLSA, was ja genau für diese Art Mischverteilungen konstruiert ist, dieses Dokument hauptsächlich dem grünen Konzept zuordnen wird. Wenn daraus die Termhierarchie extrahiert wird, werden dessen Terme klar als grün erkannt. Bei der HTE hingegen ist ja das Dokument als blau bekannt und damit werden auch seine Terme als möglicherweise blau eingestuft. So sind aufgrund der oben beschriebenen Effekte die Knoten in der Termhierarchie aus dem HAC bzw. dem PLSA tendenziell reiner als die Knoten in der Termhierarchie aus dem HTE. D. h. die Terme sind in der HAC/PLSA-Termhierarchie weniger gleichverteilt und der Variationskoeffizient muß größer werden als beim HTE, um den gleichen Effekt, nämlich, daß die Terme zuweit oben in der Hierarchie eingeordnet werden, zu erreichen. Damit können bei der Evaluierung der Termhierarchien die aus dem HAC und die dem PLSA besser abschneiden als die aus dem HTE. Da beim HAC die Dokumente scharf geclustert werden, wohingegen das PLSA eine unscharfe Zuordnung vornimmt, ist der Effekt beim HAC deutlicher sichtbar, da das hier Dokument als 100% grün eingestuft wird und alle restlichen blauen Anteile auf 0 gesetzt werden.

Wenn mithilfe der Termhierarchie nun die Dokumente in die Dokumentenhierarchie klassifiziert werden, wird natürlich das eigentlich blaue, zufällig hauptsächlich grün betermte Dokument falsch klassifizert. Da aber die Entstehung eines solchen Dokumentes eher unwahrscheinlich ist, wird es ein Einzelfall sein und profitiert die Mehrheit der Dokumente bei ihrer Klassifikation beim HAC und beim PLSA von den reineren Knoten in deren Termhierarchien, womit die Wahrscheinlichkeit, daß diese richtig klassifiziert werden, gegenüber der HTE-Termhierarchie steigt. Damit ist es auch bei der Evaluation der entstehenden Dokumentenhierarchien nicht auszu-

schließen, daß das HAC und das PLSA besser als das HTE abschneiden könnten. Insbesondere trifft das auch wieder für die Entwicklung mit steigendem Variationskoeffizienten zu (Abb. A.28, A.38 und A.39).

5.2.4 Log-Likelihood-Signifikanzschwelle, Referenzkorpus und Wortart

Diese drei Parameter beeinflussen, welche, und damit auch wieviele, Terme aus dem Dokument für den Dokumentvektor extrahiert werden (siehe Abschn. 3.1.1).

Die Log-Likelihood-Signifikanzschwelle gibt die Mindestsignifikanz an, die ein Term haben muß, um in den Dokumentvektor aufgenommen zu werden. Das gilt auch, wenn die Termgewichte nicht Signifikanzwerte, sondern Frequenzen oder tf-idf-Werte sind (siehe Abschn. 4.1).

Der Referenzkorpus dient zur Schätzung der erwarteten Häufigkeiten der Terme, die benutzt werden, um die Signifikanz der gemessenen tatsächlichen Häufigkeiten zu berechnen (siehe Abschn. 3.1.1). Da für die Strukturierung der Dokumentenkollektion die Dokumente nur gegeneinander ausgezeichnet werden müssen, lag es nahe, die Dokumentkollektion selber zur Grundlage eines Referenzkorpus' zu machen.

Die Wortart der extrahierten Terme wurde variiert, da üblicherweise die Substantive die Terme sind, die die meiste dokumentspezifische Information tragen. Für die Bestimmung der Wortart ist aber sprachspezifisches Wissen nötig, und die Signifikanz stellt ein wortartunabhängiges Maß für die Bedeutung eines Term für ein Dokument dar. Deswegen wurden Experimente mit einer Beschränkung nur auf Substantive[38] und Experimente, bei denen alle signifikanten Terme oberhalb der Mindestsignifikanz in den Dokumentvektor aufgenommen wurden, gemacht. Außerdem wurden zum Vergleich noch Experimente mit Dokumentvektoren aus allen Types des Dokumentes außer Stopwörtern gemacht.

Im vorhinein ist festzustellen, daß die Beschränkung auf signifikante Terme bzw. signifikante Substantive eine deutliche Reduktion der Berechnungskomplexität bewirkt (s. u.). Aufgrund der Größe der Spiegelkollektion und zeitlicher Rahmenbedingungen wurden diese Experimente nur mit der WRT-Kollektion gemacht.

[38] genauer: Terme, deren POS-Tag mit „N“ beginnt

Die Abbildungen A.52 und A.53 geben an, wieviele verschiedene Terme, also Types, bei den Experimenten zu verarbeiten waren. Abbildung A.52 zeigt die Terme, die mit dem allgemeinsprachlichen Korpus als Referenzkorpus (bezeichnet als de-Korpus) extrahiert wurden, Abbildung A.53 die Terme, die unter Verwendung der Dokumentenkollektion als Referenzkorpus (bezeichnet als lokaler Korpus) extrahiert wurden. Diese Anzahl stellt z. B. die Breite der Dokument-Term-Matrix dar, womit die Reduktion des Berechnungsaufwandes deutlich wird. Außerdem ist ersichtlich, daß die Terme, mit den höheren Signifikanzwerten hauptsächlich bis fast ausschließlich Substantive sind. Bei der Verwendung des lokalen Korpus' gibt es mehr Terme mit höherer Signifikanz, die also die Dokumente deutlicher voneinander unterscheiden. Das schlägt sich auch in den Ergebnissen der Experimente nieder. Bei der Beschränkung auf höchstsignifikante Terme in den Dokumentvektoren werden die Ergebnisse für das HAC und HTE bei der Verwendung des allgemeinsprachlichen Korpus' schlechter (Abbildung A.54). Diesen Einbruch gibt es bei Verwendung des lokalen Korpus' nicht (Abbildung A.55). Die Abbildungen A.56 und A.60, zeigen dasselbe aus einer anderen Perspektive. Beim PLSA sind die sichtbaren Schwankungen wahrscheinlich hauptsächlich darauf zurückzuführen, daß der Algorithmus nichtdeterministisch ist (Abb. A.54, A.55 und A.58).

Die Ergebnisse zeigen deutlich (auch Abb. A.57, A.59 und A.61) , daß kein externer Referenzkorpus nötig ist, und daß die Auswahl aller signifikanten Terme über einer gewissen Mindestsignifikanz die besten Ergebnisse liefert. Das hat zwei deutliche Vorteile: Durch die Beschränkung auf die signifikanten Terme wird der Berechnungsaufwand erheblich reduziert, und durch den Verzicht auf ein allgemeinsprachliches Korpus und eine Filterung der Terme nach Wortarten mit einem POS-Tagger ist nur sehr wenig sprachspezifisches Wissen nötig, um die Dokumentenkollektion zu strukturieren. Das läßt die Eignung der Verfahren für die Handhabbarmachung von Textquellen in bisher wenig erforschten Sprachen wahrscheinlich erscheinen.

5.2.5 Termgewicht

Die ursprüngliche Idee war, die Dokumentvektoren mit den verschiedenen Termgewichten als Ausgangsdaten für die Verfahren zu benutzen (siehe Abschn. 3.1.1). Aus Zeitgründen konnten die Verfahren aber leider nur mit den Frequenzvektoren erprobt werden. Lediglich für den Teil der Klassifi-

kation der Dokumente mithilfe der Termhierarchie konnten auch die Signifikanz und die *tf-idf* als Termgewicht exemplarisch getestet werden. Hierbei zeigten sich keine signifikanten Unterschiede zwischen den verschiedenen Termgewichten (Abb. A.62 und A.63). Dieses Ergebnis ist aber ohne weitere Untersuchungen nicht auf die Verwendung der anderen Termgewichte als Ausgangsdaten für die Verfahren übertragbar.

5.2.6 PLSA-Läufe

Da das PLSA ein nichtdeterministisches Verfahren ist, ist nicht zu erwarten, daß zwei verschiedene Läufe mit denselben Ausgangsdaten dasselbe Ergebnis liefern (siehe Abschn. 3.1.3). Vielmehr ist zu erwarten, daß der EM-Algorithmus jedesmal in einem anderen lokalen Maximum endet.

Um ermessen zu können, wie verläßlich die Ergebnisse sind, wurde ein Experiment zehnmal durchgeführt. Dabei zeigten sich keine großen Abweichungen zwischen den einzelnen Läufen des PLSA (Abb. A.64 und A.65).

Außerdem wurde noch untersucht, inweit es möglich ist, die möglichen Defizite einzelner Ergebnisse durch Mittelung mehrerer Ergebnisse zu kompensieren (siehe Abschn. 4.3). Dabei zeigte sich aber keine grundsätzliche Verbesserung der Ergebnisse. Bei den Experimenten mit der Spiegelkollektion zeigten sich leichte Verbesserungen (Abb. A.17). Bei der WRT-Kollektion führte dieser Ansatz zu deutlichen Verschlechterungen der Ergebnisse (Abb. A.4).

Dieser Effekt läßt sich dadurch erklären, daß die extrahierten Zuordnungen der Terme zu den latenten Konzepten durch die Mittelung verwischt werden. Damit nähern sich die Werte der Terme für die Konzepte mehr an die Gleichverteilung an, wodurch bei der Zuordnung der Terme zu den Knoten der Termhierarchie tendenziell weiter oben in die Hierarchie eingeordnet werden, und das auch bei tendenziell kleinerem Variationskoeffizienten (siehe auch Abschn. 5.2.2). Damit steigt der Recall gegenüber nur einem PLSA-Lauf (Abb. A.10 und A.23). Bei der WRT-Kollektion fällt die Precision deutlich (Abb. A.8), wohingegen sie bei der Spiegelkollektion nahezu unverändert bleibt (Abb. A.21).

Auf die Mittelung mehrerer PLSA-Läufe kann somit verzichtet werden.

Kapitel 6

Zusammenfassung und Ausblick

Die vorliegende Arbeit beschäftigt sich mit der hierarchischen Strukturierung von Dokumentenmengen. Dabei werden für die Verarbeitung der natürlichsprachlichen Texte hauptsächlich statistische Methoden verwendet und bestehende Verfahren der Strukturfindung bzw. -bildung neu kombiniert, um die gegebene Zielstellung der Extraktion einer Termhierarchie zu erreichen. Diese Termhierarchie dient dabei sowohl als Grundlage für die Hierarchisierung der Dokumentenkollektion als auch der Beschreibung der gefundenen Dokumententeilmengen.

Die Experimente mit synthetischen Dokumente, die mithilfe eines statistischen Modells generiert wurden, bewiesen die Funktionsfähigkeit der kombinierten Verfahren.

Die Anwendung auf reale Dokumentenkollektionen zeigte die prinzipielle Machbarkeit des Vorhabens, auch wenn nur ein geringer Teil der Information einer zum Vergleich vorgegebenen Strukturierung der Daten rekonstruiert werden konnte. Das deckt sich mit den Ergebnissen vorangegangener Arbeiten auf dem Gebiet der Extraktion von Taxonomien [8].

Dabei zeigte sich auch eine deutliche Abhängigkeit der Güte der Ergebnisse von der Textsorte der zu verarbeitenden Dokumente. Weitere Untersuchungen oder auch Optimierungen auf bestimmte Aufgabenstellungen sollten diesen Aspekt entsprechend berücksichtigen. Sehr wahrscheinlich werden die Verfahren generell für wissenschaftliche Texte wie z. B. Papers und Abstracts oder allgemein stark themenspezifische Dokumente wie z. B. aus dem

(e-)learning-Bereich besser geeignet und optimierbar sein als für allgemeinsprachliche Dokumente mit nur wenig signifikanten und diskriminierenden Terme wie z. B. Zeitungstexte.

Daneben sind auch ganz andere Einsatzmöglichkeiten denkbar. Wenn z. B. aus einem großen Korpus berechnete Kookkurrenzvektoren für Terme vorliegen, also die die signifikanten Kookkurrenzen eines Terms als Vektor dargestellt, können diese Vektoren als „Dokumente“ betrachtet und mithilfe der beschriebenen Verfahren strukturiert werden. Möglicherweise wäre das ein vielversprechender Ansatz für das Ontology Learning.

Außerdem gibt es, wie in den einzelnen Abschnitten beschrieben, viele Punkte, an denen möglicherweise einflußreiche Veränderungen der bisherigen Versuchsanordnung vorgenommen werden können. Das reicht vom Einsatz anderer strukturbildender Verfahren, die z. B. auch die Anzahl der Knoten und Ebenen in der entstehenden Hierarchie selbst bestimmen, bis zur Verallgemeinerung der Definition einer Termhierarchie, z. B.in dem Sinne, daß Terme mit verschiedenen Gewichten oder zu mehreren Knoten gehören können, und den verwendeten Maßen für die Berechnung der Ähnlichkeiten, sowohl zwischen Dokumenten und Termhierarchieknoten als auch zwischen Dokumenten und Dokumenten. So könnten die Berücksichtung linguistischer Effekte wie der Synonymie oder die Einbeziehung der Terme von Subknoten bei der Berechnung der Ähnlichkeiten von Dokumenten mit Knoten Verbesserungen mitsichbringen. Vielversprechend könnte auch sein, bei dem Schritt der Extraktion der Termhierarchie Sätze (oder Absätze) statt ganze Dokumente oder Kapitel zu betrachten, da die Mischung (von Termen) verschiedener Themen in kleineren Einheiten wie Sätzen weitaus seltener auftritt als in großen wie ganzen Dokumenten.

Anhang A

Abbildungen

- Wissen und Text
  - Text Mining
  - Aufbau & Struktur von Text
  - Wissensverarbeitung gestern & heute
- Grundlagen der Bedeutungsanalyse
  - Syntagmatische Relationen
  - Linguistische Ebenen
  - Paradigmatische Relationen
  - Semantische Relationen
  - Logische Relationen
  - Logische Relationen – Formal
  - Fach- & Allgemeinsprache
- Textdatenbanken
  - Textressourcen
  - Aufbau von Textdatenbanken
  - Segmentierung von Text
  - Datenstrukturen für Wörter
  - Abfragemöglichkeiten in Textdatenbanken
- Sprachstatistik
  - Zipfsche Gesetze
  - Differenzanalyse
  - Probabilistisches Sprachmodell 1
  - Probabilistisches Sprachmodell 2
  - Hidden Markov-Modelle
  - Tagging – als Anwendung HMM
  - Signifikante Kookkurrenzen
  - Signifikante Kookkurrenzen – Beispiele und Anwendungen
  - Visualisierung von signifikanten Kookkurrenzen
  - Anwendung auf andere Sprachen
  - Kookkurrenzen höherer Ordnung
  - Netze von Kookkurrenten
  - Small Worlds und skalenfreie Netze
  - Disambiguierung
  - Communities
- Clustering
  - Clustering-Verfahren
  - Dokumentenähnlichkeit
  - Clustern von Wortformen
  - Merkmalsextraktion
- Musteranalyse
  - Reguläre Ausdrücke
  - Syntaktische Muster
  - Morphemmuster
- Hybride Verfahren
  - Lexikalische Filter
  - Kombination verschiedener Wissensquellen
  - Bootstrapping
- Beispielanwendungen
  - Fachterminologie
  - Erkennung von Fachterminologie
  - Beispielanalysen von zwei Textsammlungen
  - Aufbau einer Wissenslandkarte

Abbildung A.1: Struktur der WRT-Kollektion

- auto
  - fahrberichte
  - werkstatt
- geld
- jobundberuf
- kultur
  - kino
  - literatur
  - musik
- politik
  - ausland
  - deutschland
- reise
- sport
  - formel1
  - fussball
- studium
- technologie
- wissenschaft
  - erde
  - mensch
  - weltraum

Abbildung A.2: Struktur der Spiegelkollektion

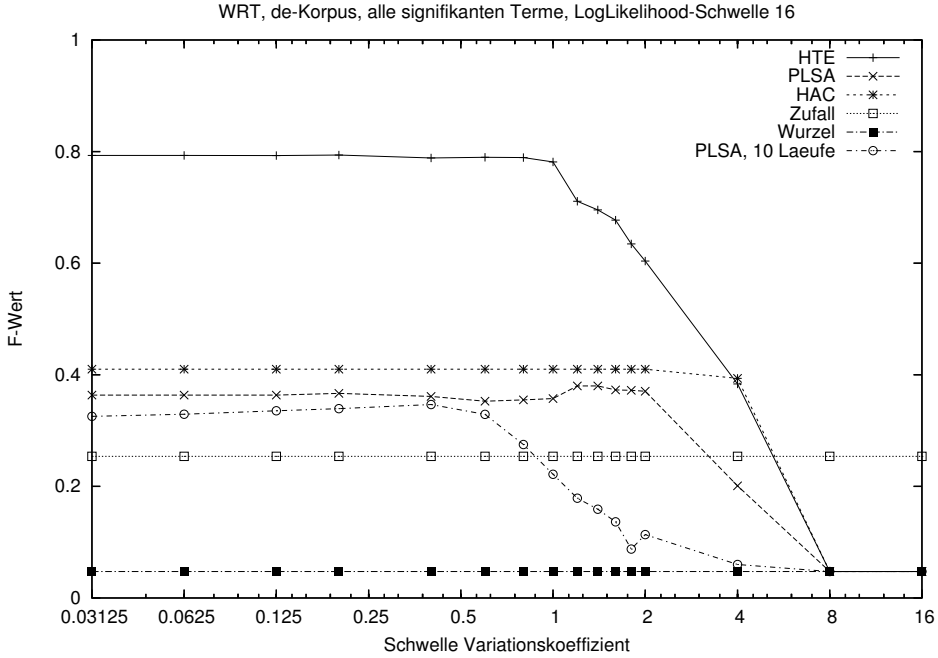

Abbildung A.3: Der F-Wert ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

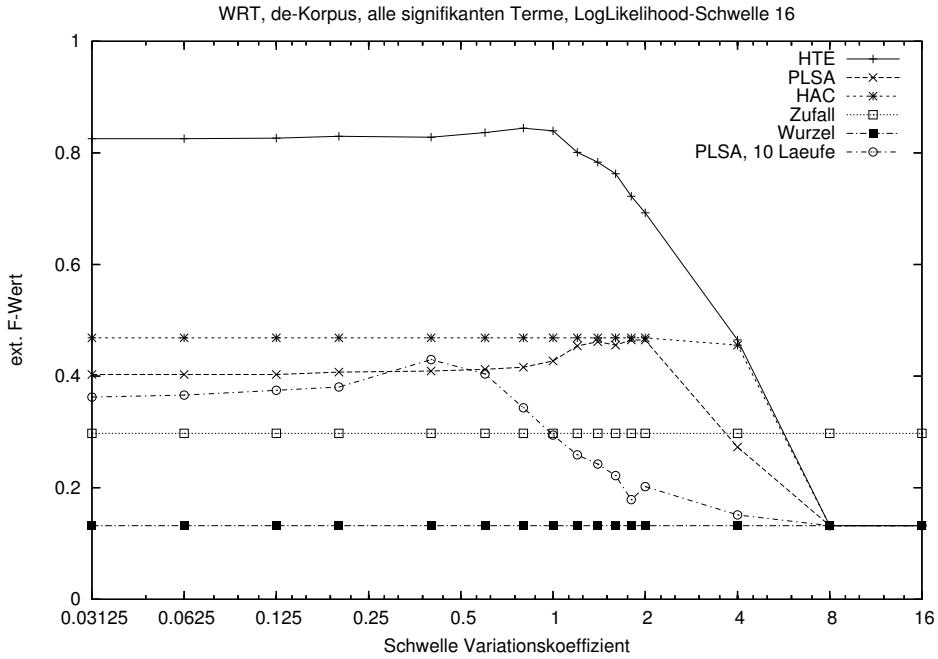

Abbildung A.4: Der ext. F-Wert ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

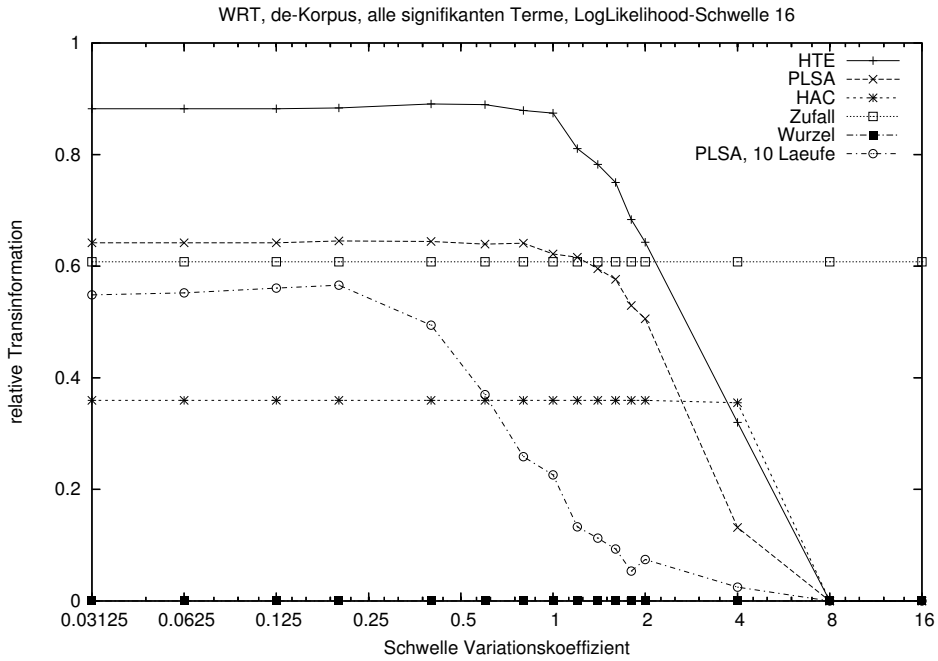

Abbildung A.5: Die relative Transinformation ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

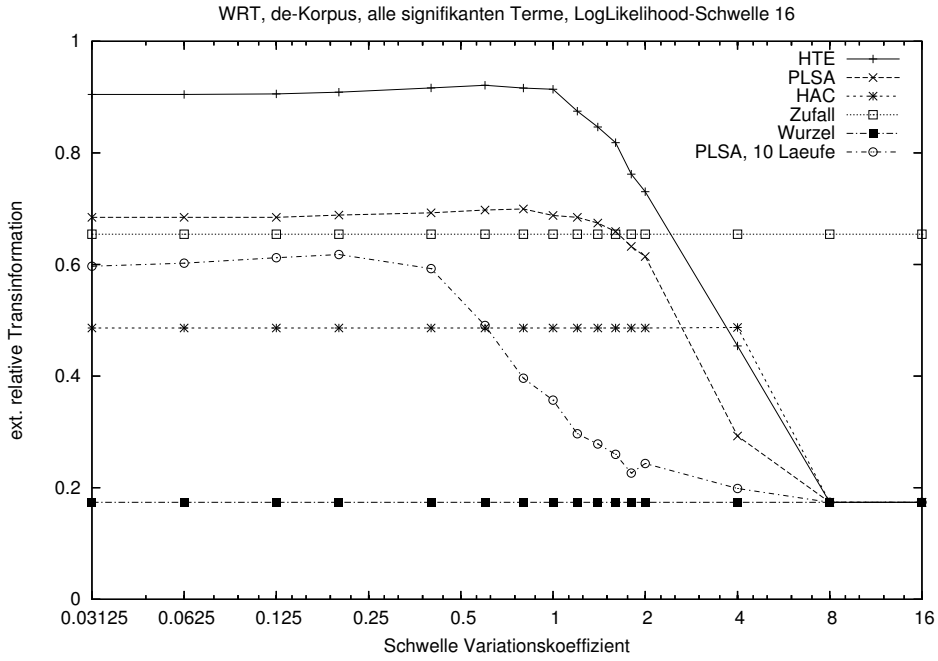

Abbildung A.6: Die ext. relative Transinformation ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

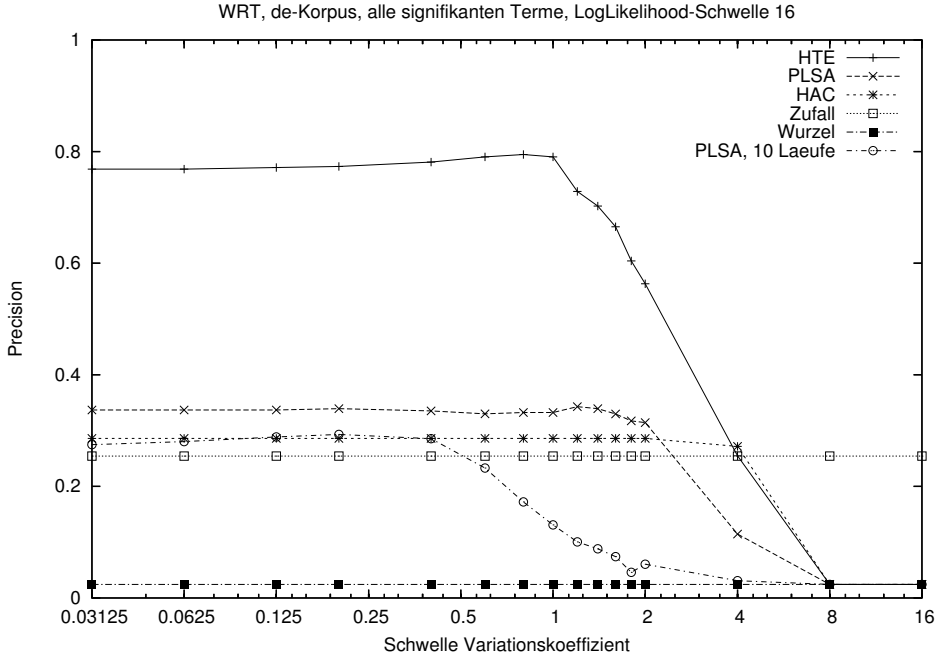

Abbildung A.7: Die Precision ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

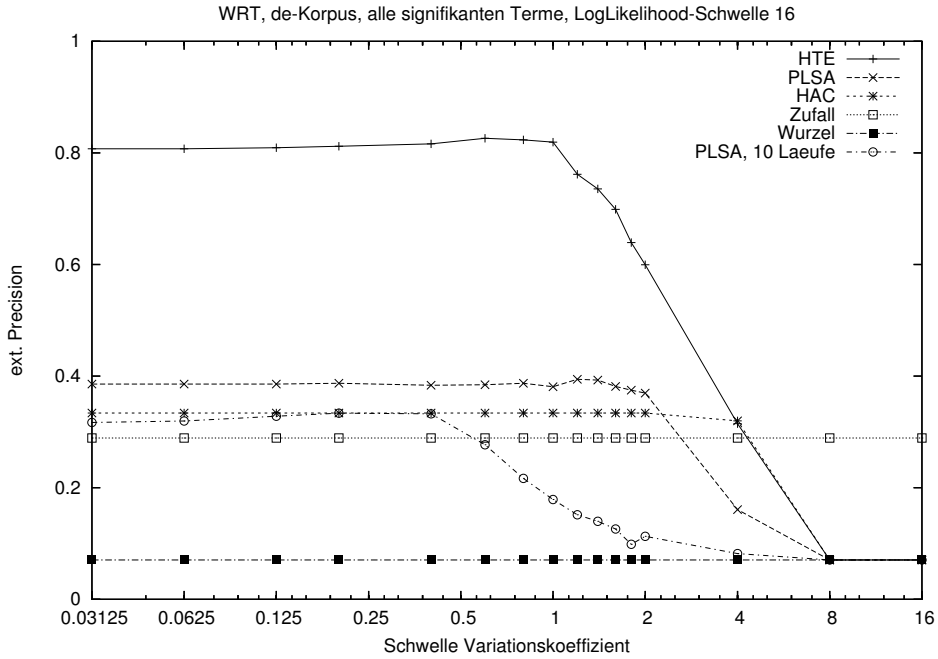

Abbildung A.8: Die ext. Precision ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

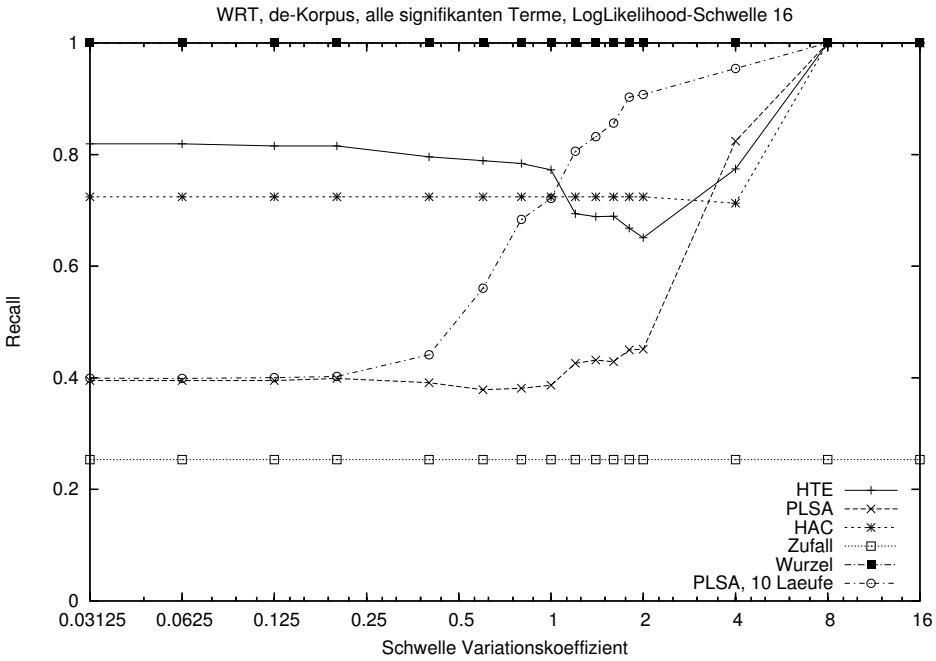

Abbildung A.9: Der Recall ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

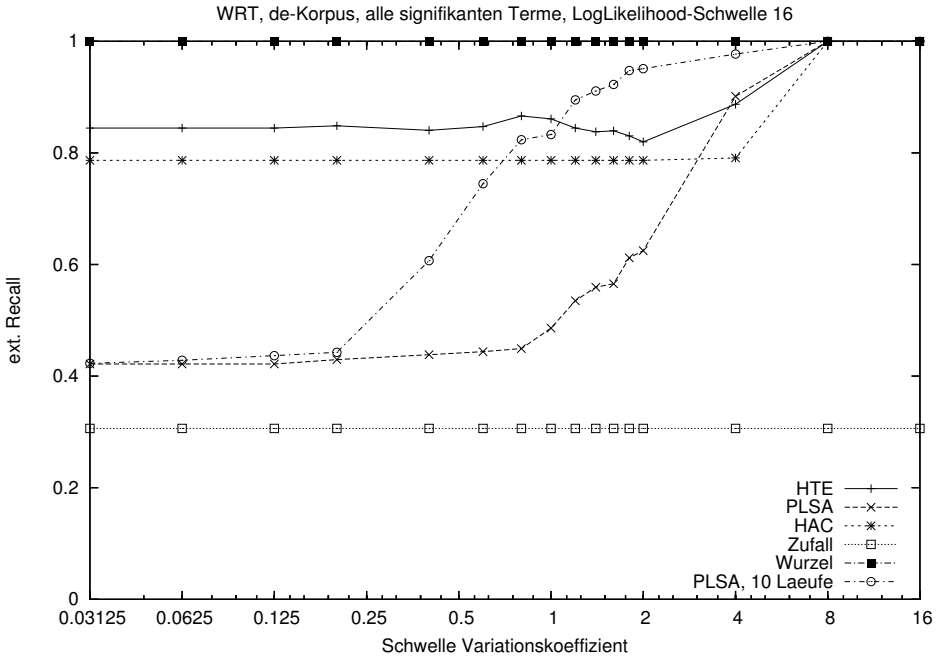

Abbildung A.10: Der ext. Recall ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

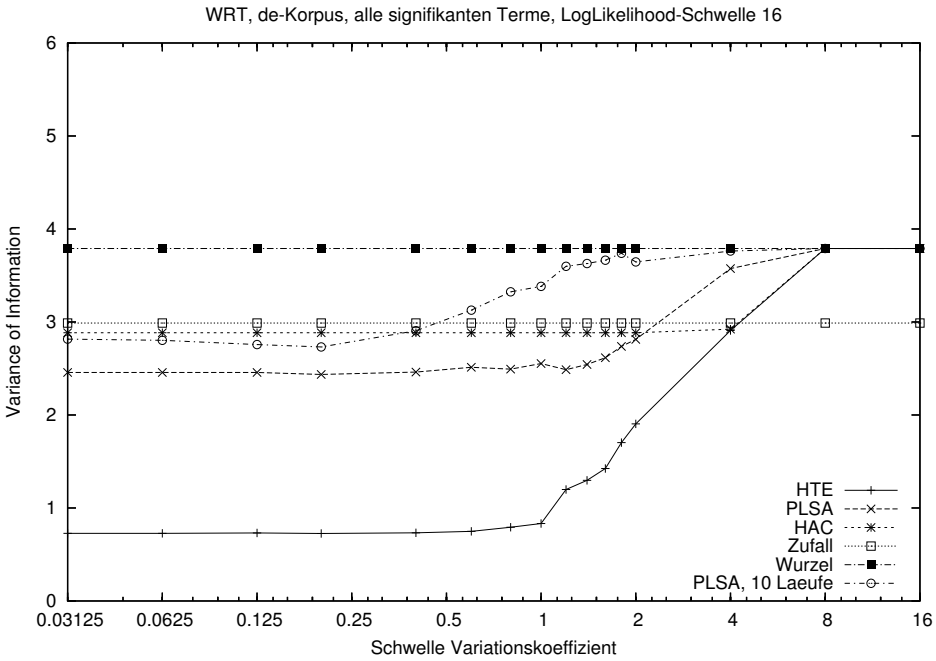

Abbildung A.11: Die Variance of Information ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

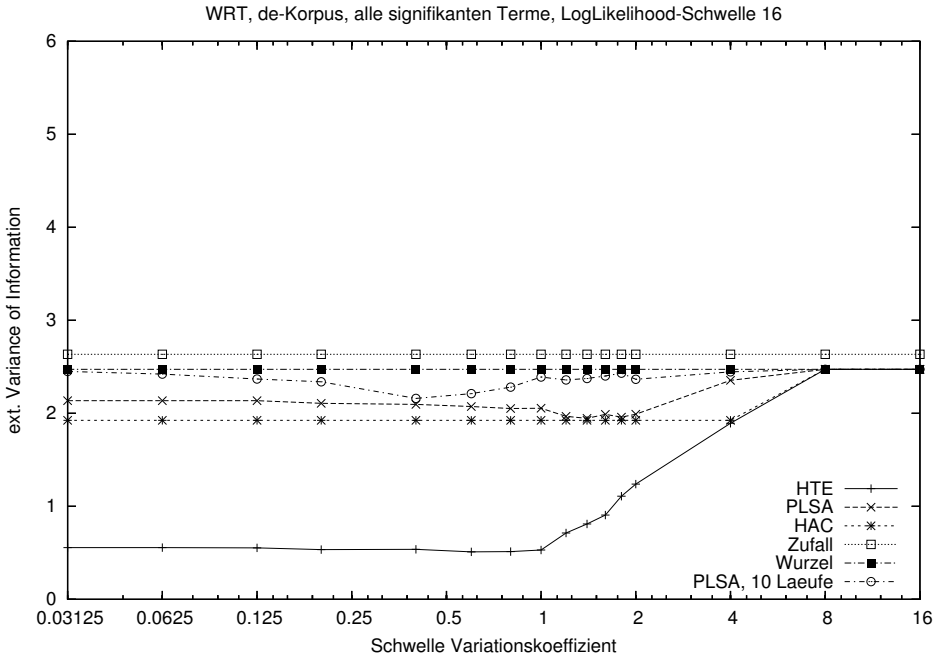

Abbildung A.12: Die ext. Variance of Information ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

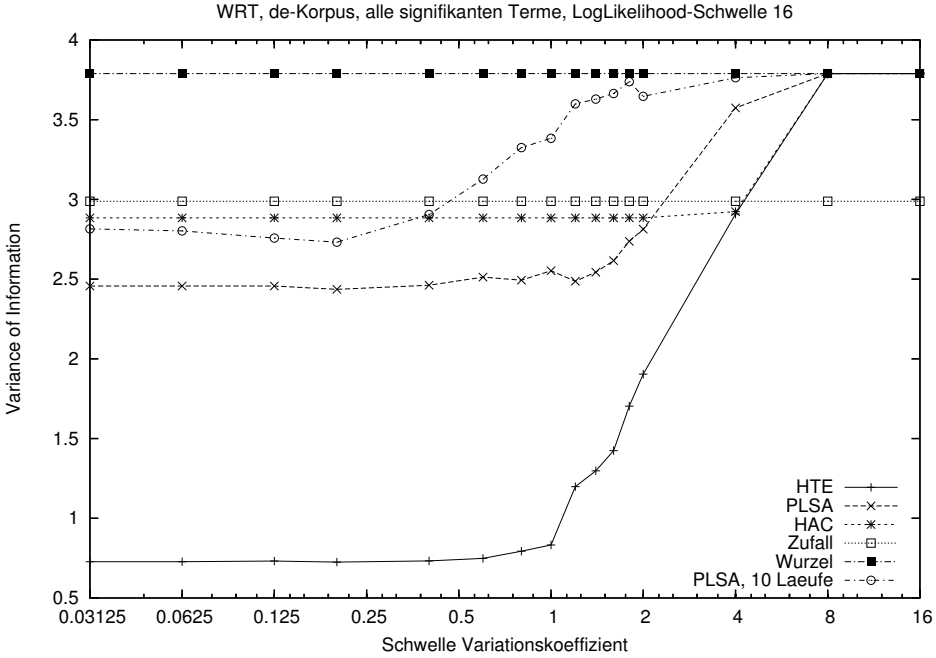

Abbildung A.13: Die Variance of Information ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

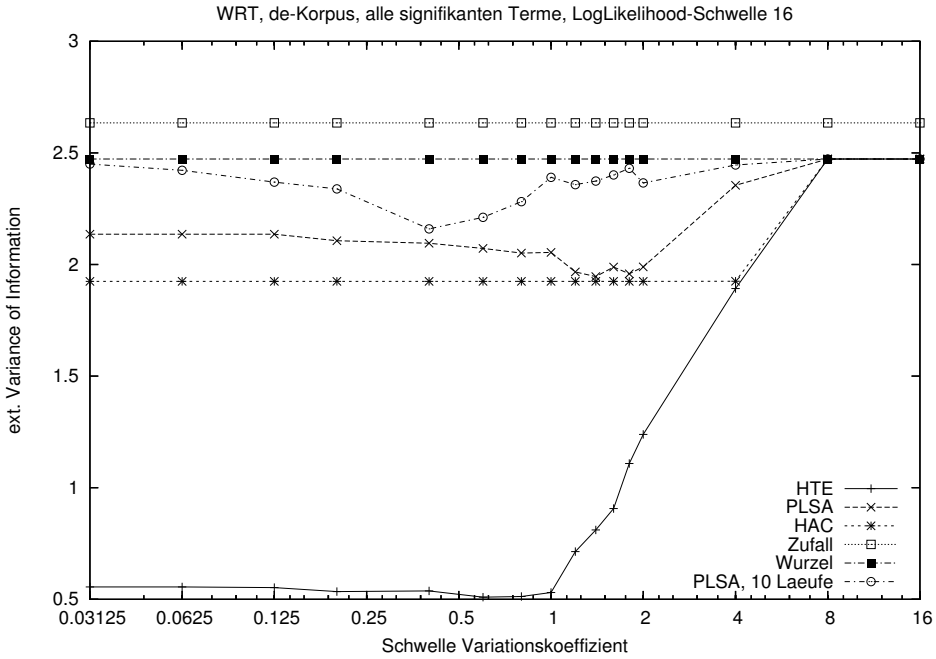

Abbildung A.14: Die ext. Variance of Information ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

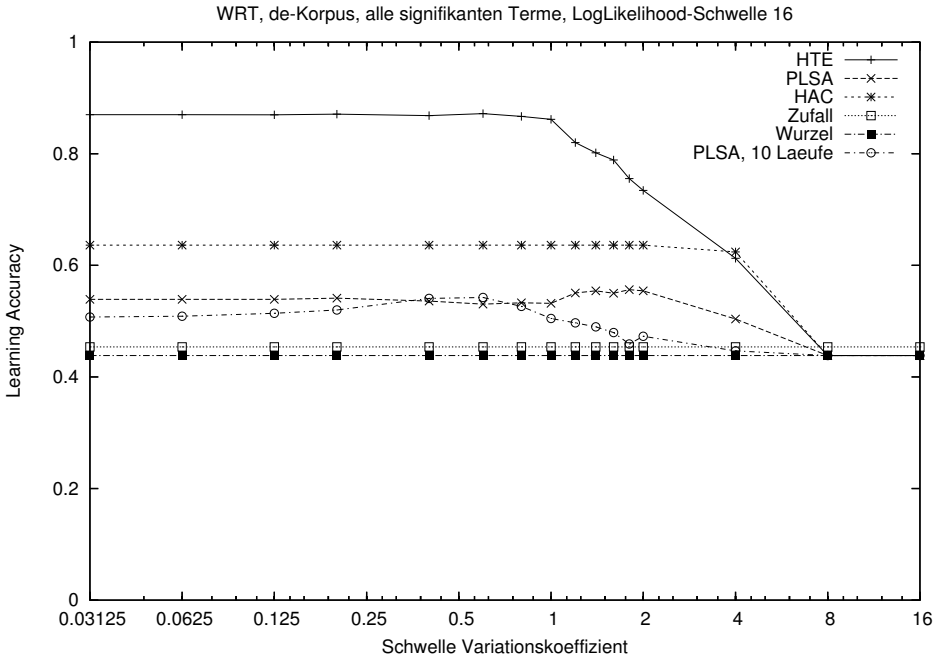

Abbildung A.15: Die Learning Accuracy ggü. dem Varianzkoeffizienten (WRT, alle Terme mit Signifikanz ≥ 16 bzgl. des de-Korpus')

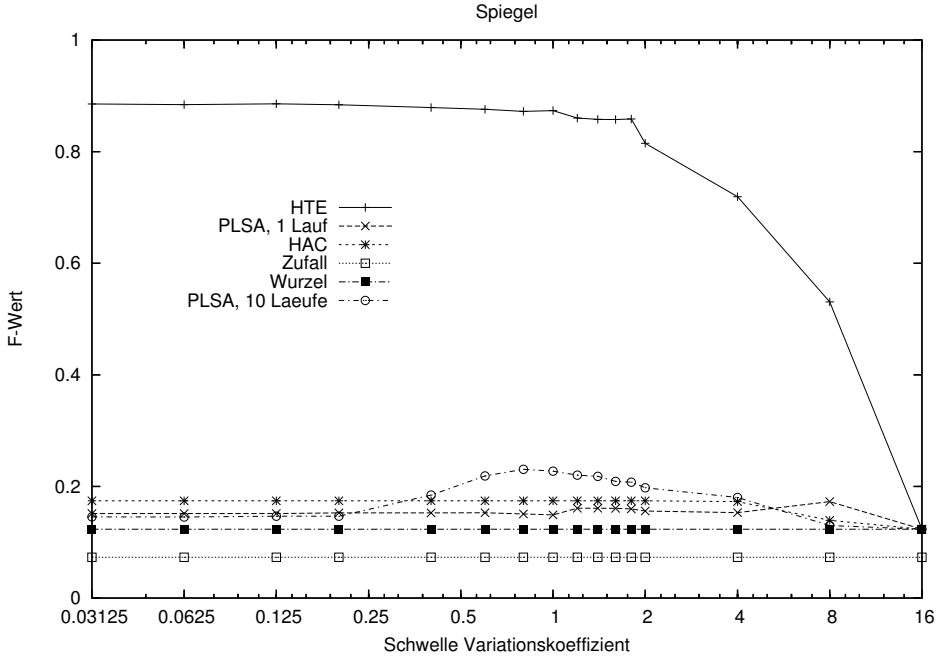

Abbildung A.16: Der F-Wert ggü. dem Varianzkoeffizienten (Spiegel)

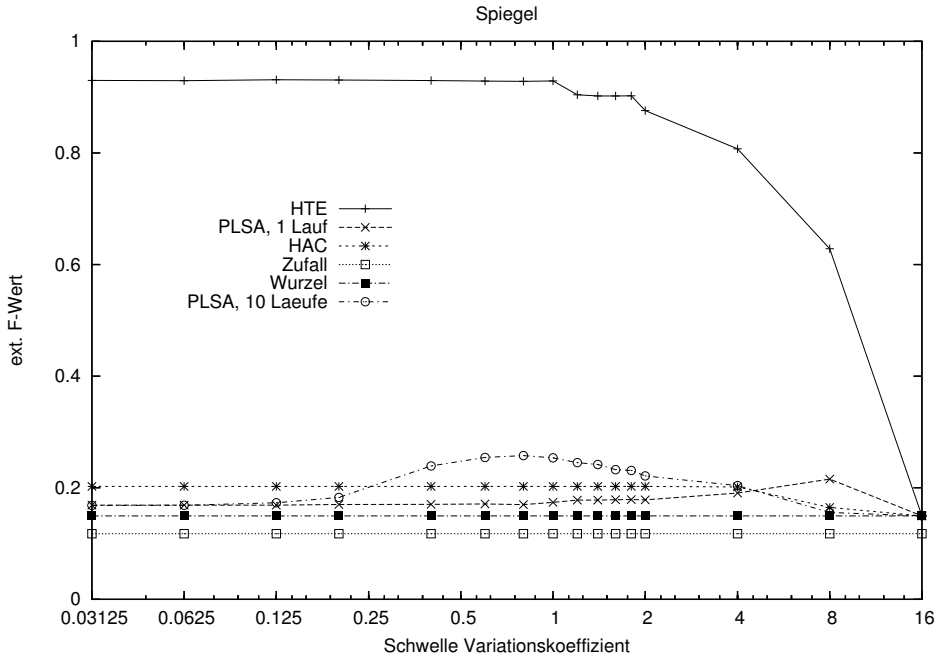

Abbildung A.17: Der ext. F-Wert ggü. dem Varianzkoeffizienten (Spiegel)

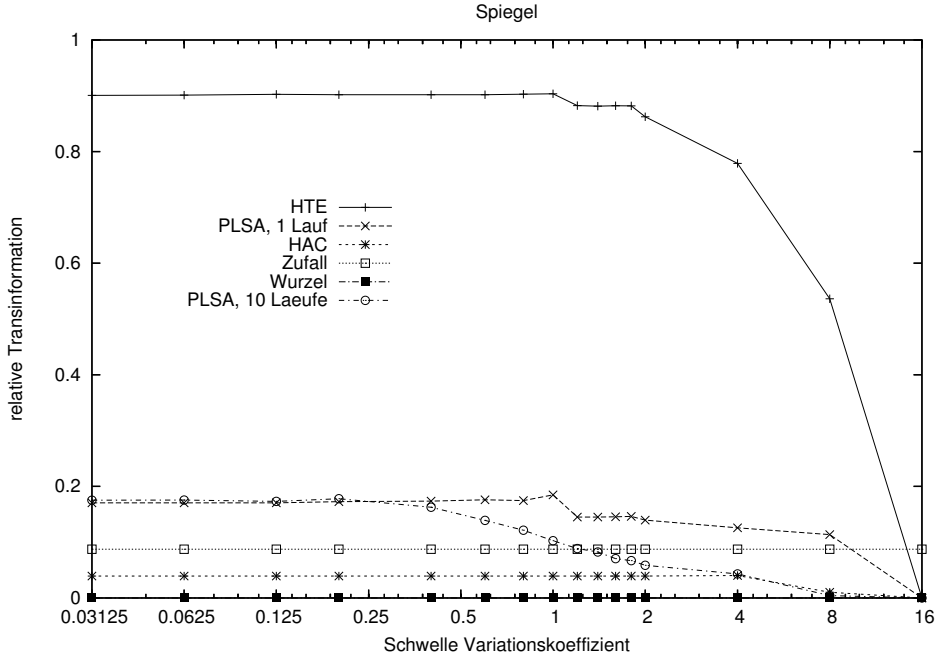

Abbildung A.18: Die relative Transinformation ggü. dem Varianzkoeffizienten (Spiegel)

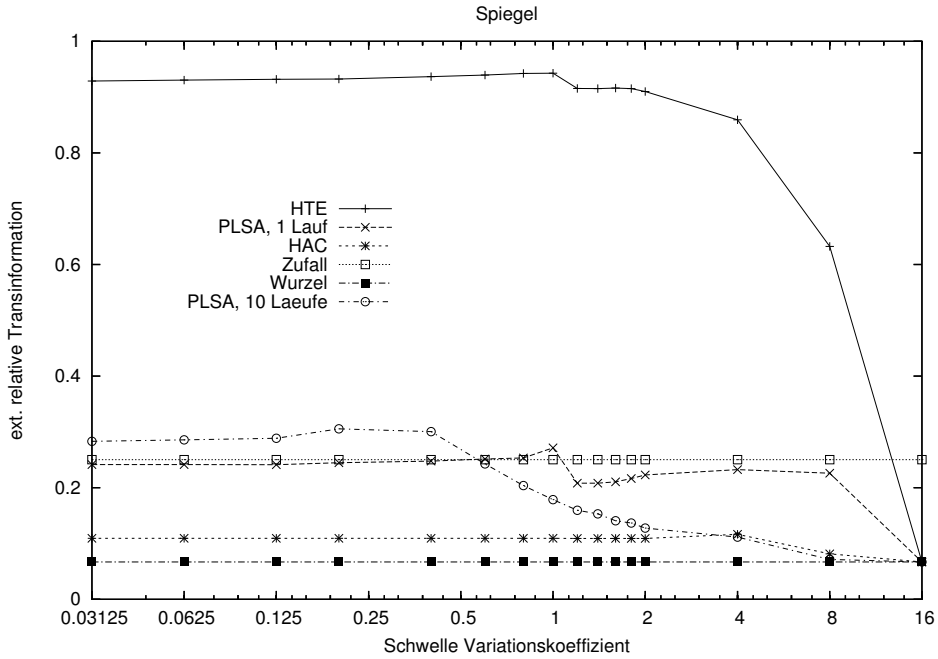

Abbildung A.19: Die ext. relative Transinformation ggü. dem Varianzkoeffizienten (Spiegel)

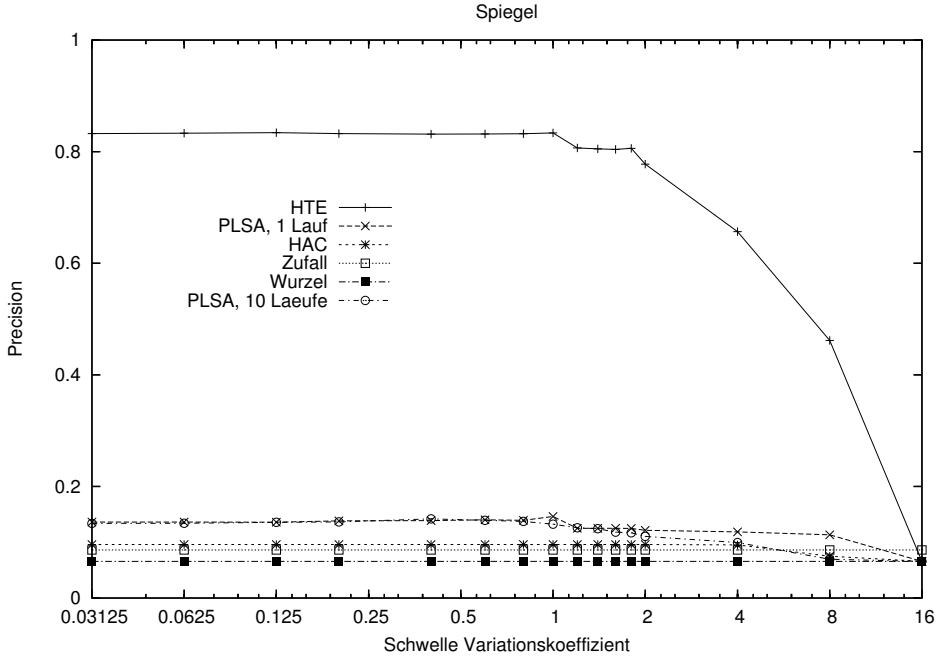

Abbildung A.20: Die Precision ggü. dem Varianzkoeffizienten (Spiegel)

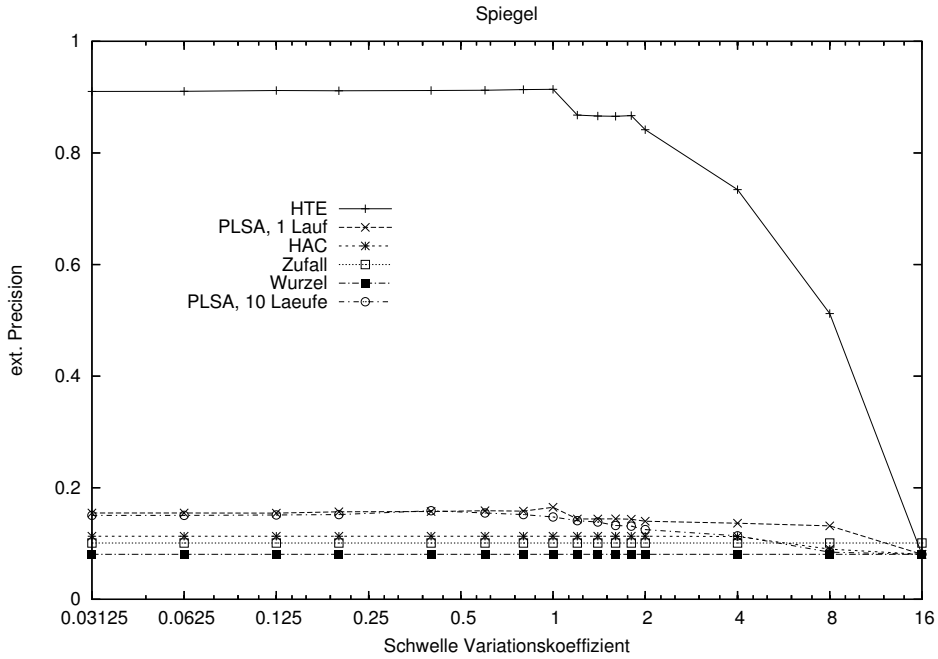

Abbildung A.21: Die ext. Precision ggü. dem Varianzkoeffizienten (Spiegel)

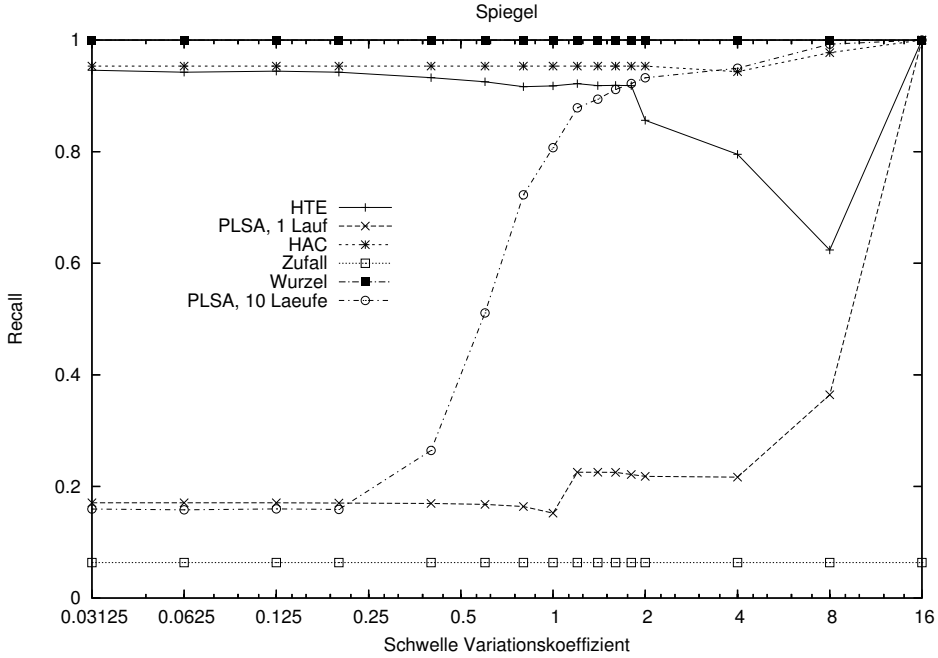

Abbildung A.22: Der Recall ggü. dem Varianzkoeffizienten (Spiegel)

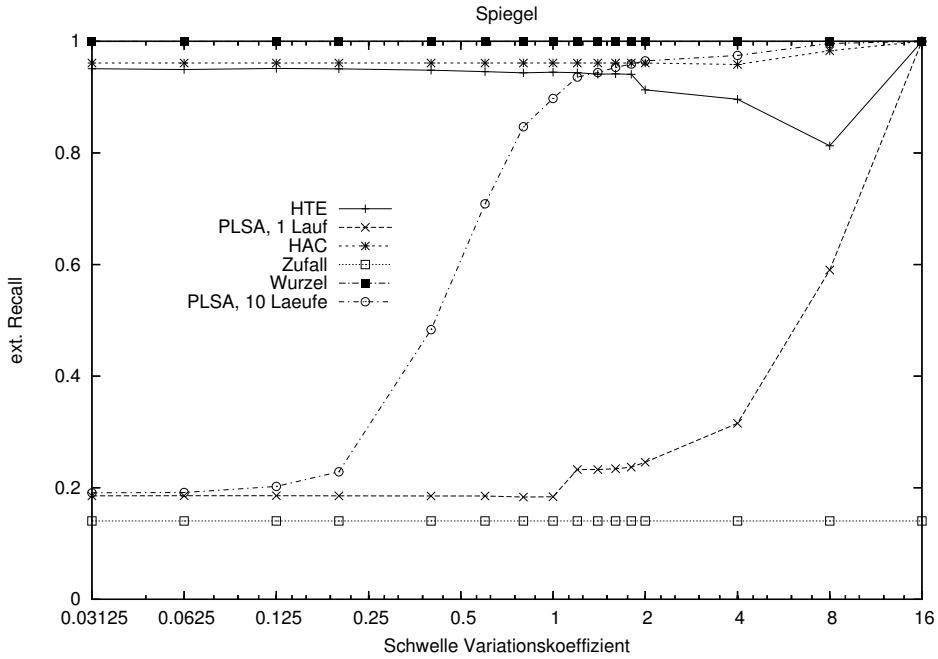

Abbildung A.23: Der ext. Recall ggü. dem Varianzkoeffizienten (Spiegel)

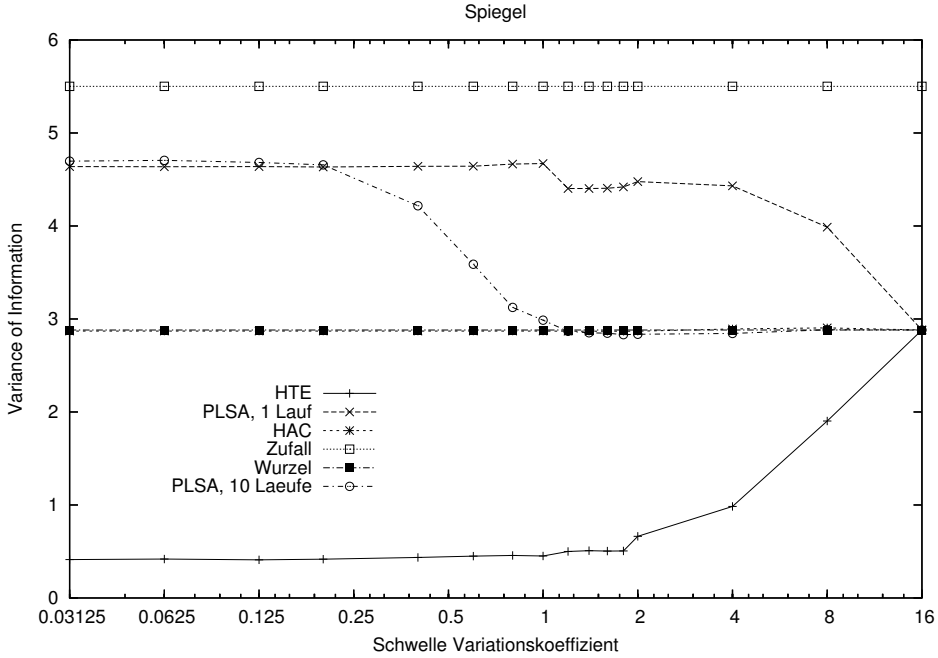

Abbildung A.24: Die Variance of Information ggü. dem Varianzkoeffizienten (Spiegel)

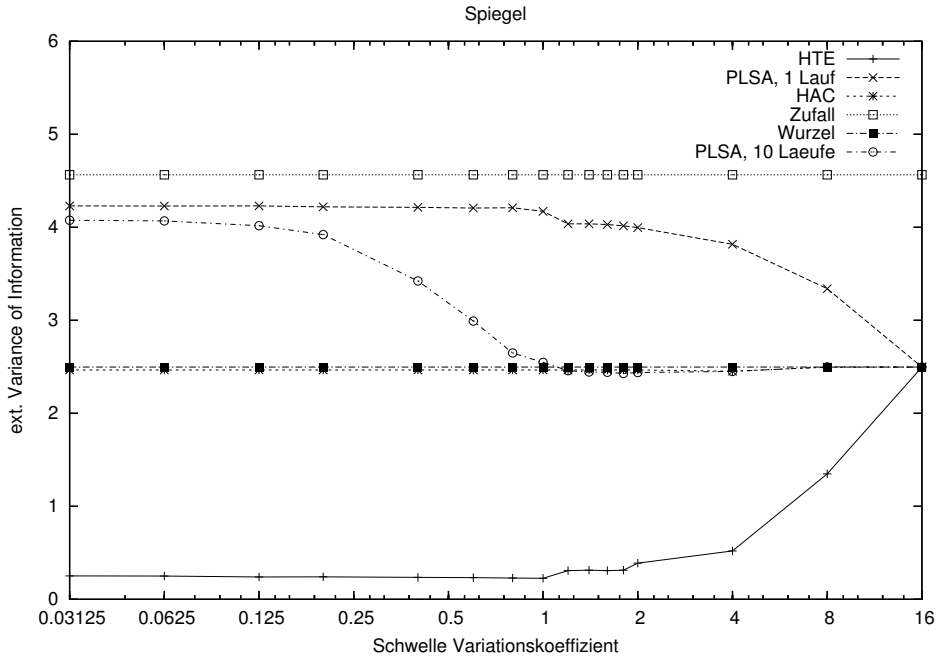

Abbildung A.25: Die ext. Variance of Information ggü. dem Varianzkoeffizienten (Spiegel)

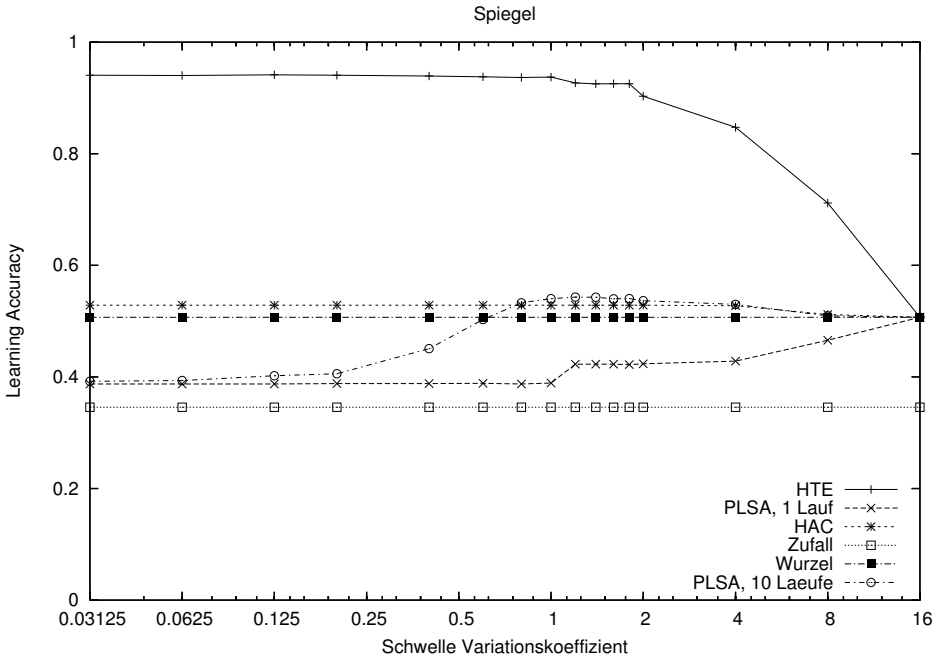

Abbildung A.26: Die Learning Accuracy ggü. dem Varianzkoeffizienten (Spiegel)

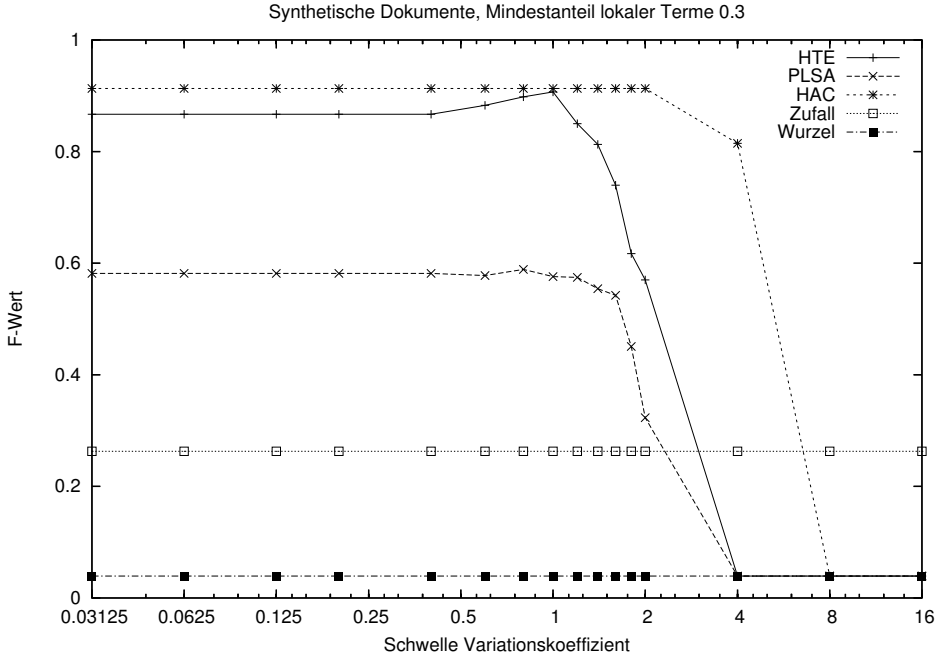

Abbildung A.27: Der F-Wert ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

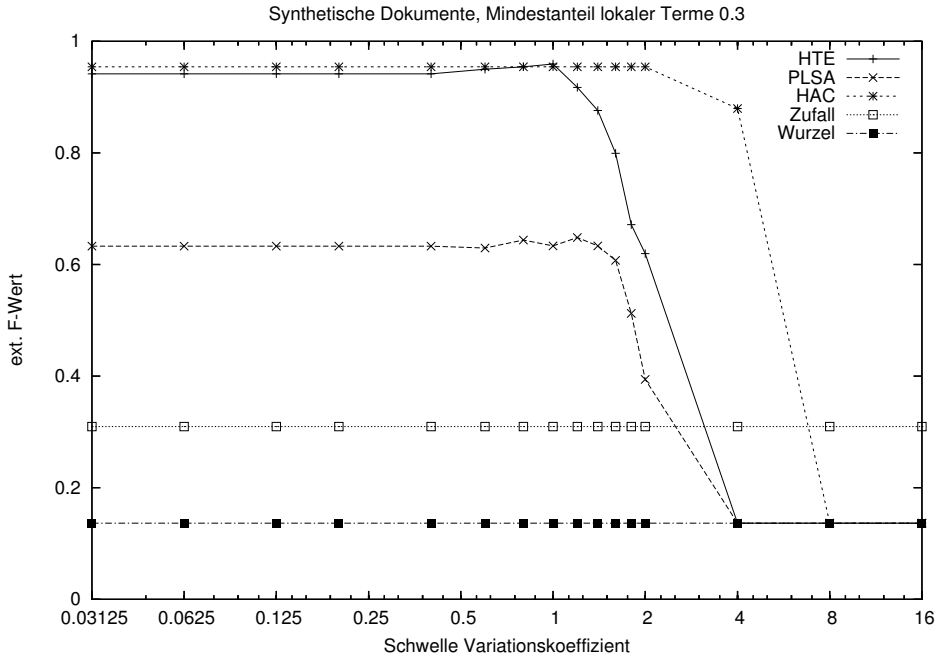

Abbildung A.28: Der ext. F-Wert ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

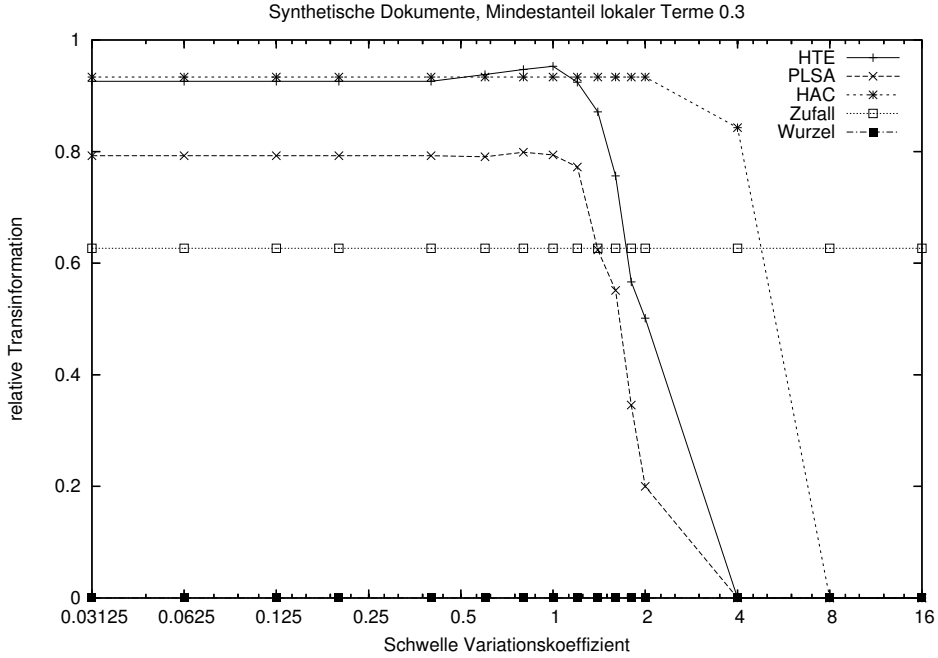

Abbildung A.29: Die relative Transinformation ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

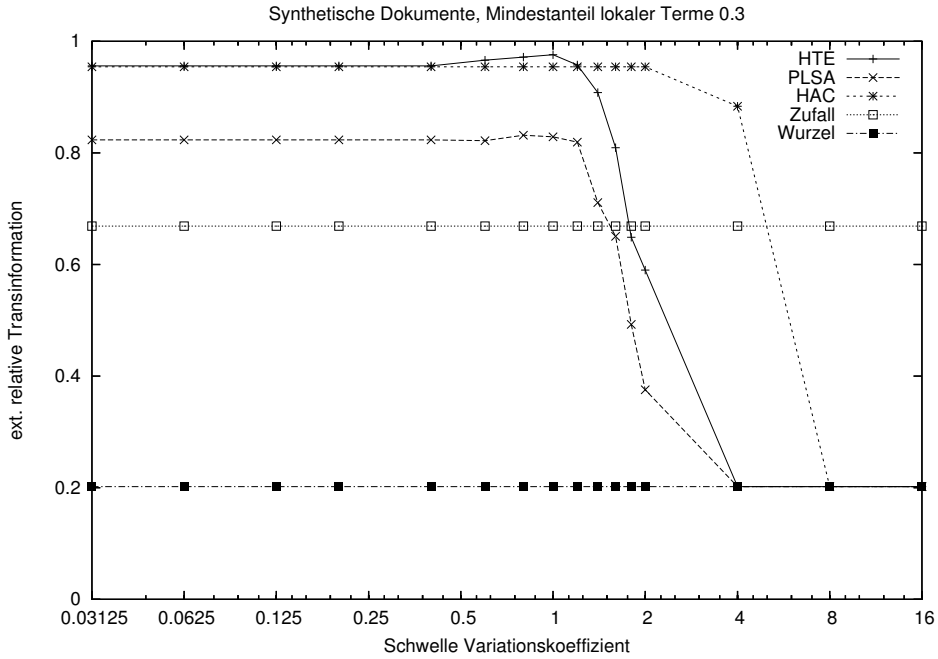

Abbildung A.30: Die ext. relative Transinformation ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

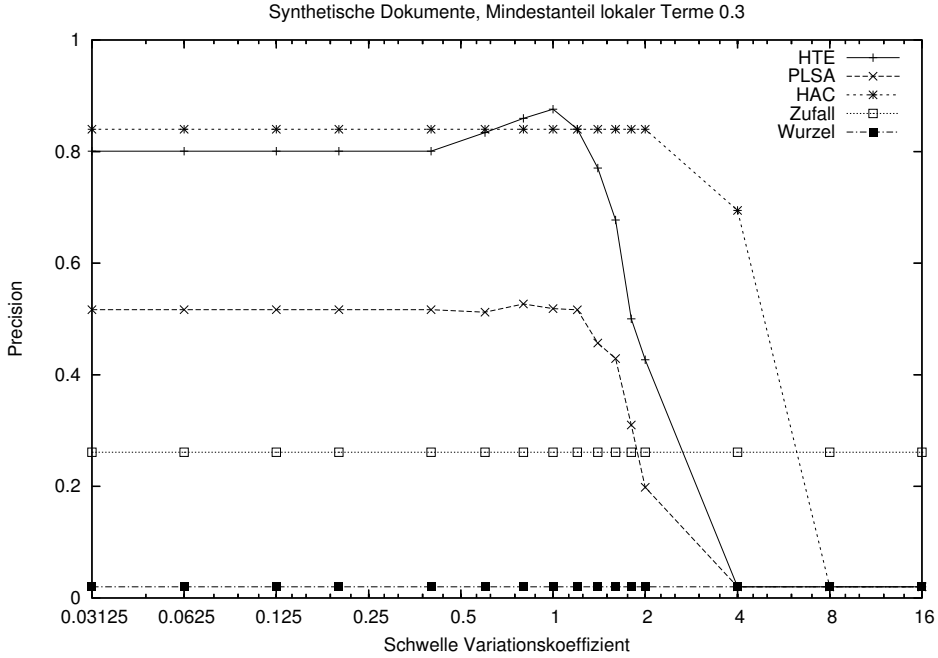

Abbildung A.31: Die Precision ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

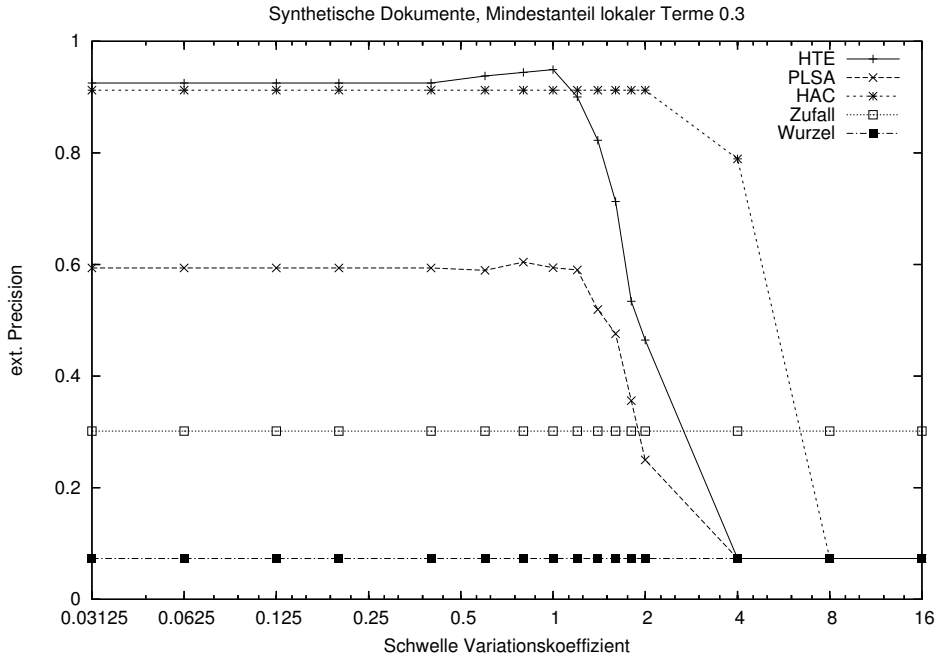

Abbildung A.32: Die ext. Precision ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

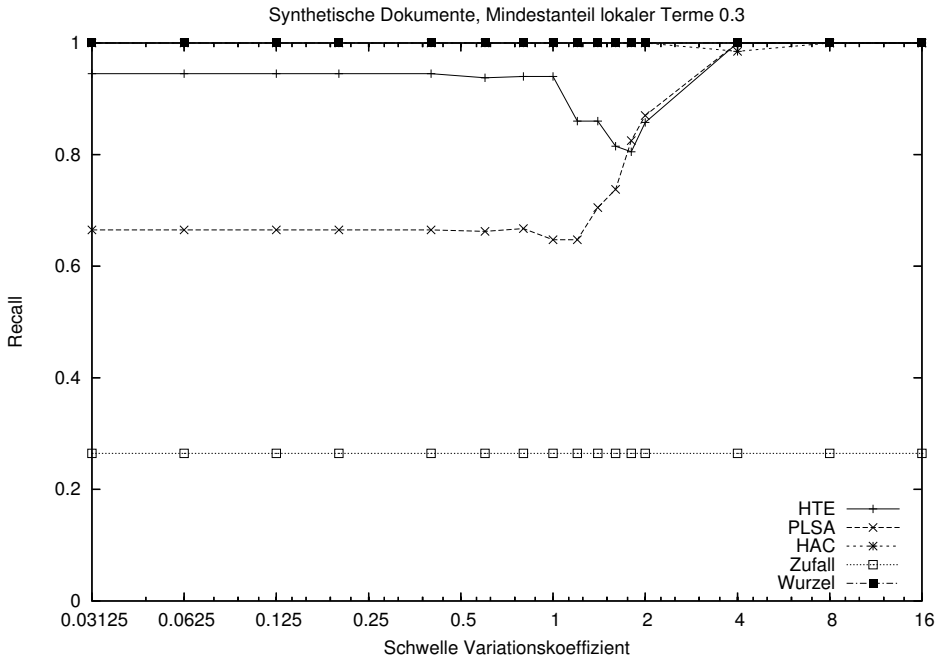

Abbildung A.33: Der Recall ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

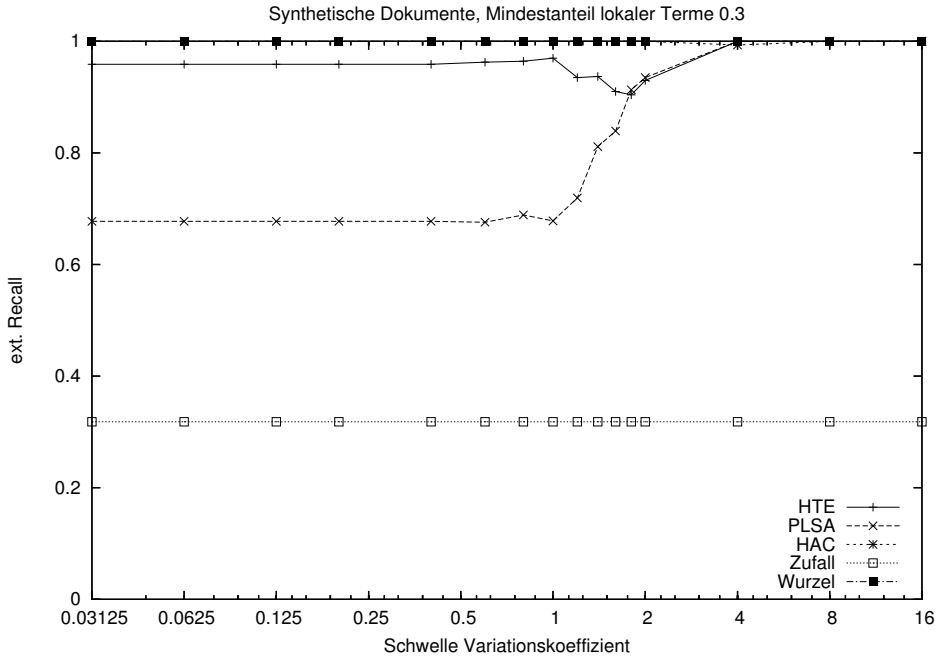

Abbildung A.34: Der ext. Recall ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

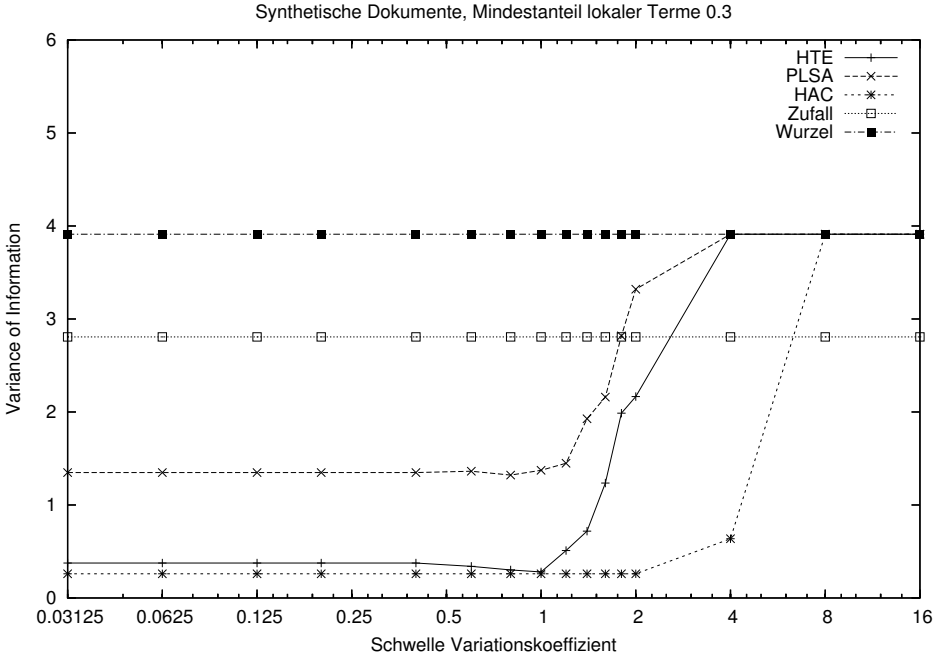

Abbildung A.35: Die Variance of Information ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

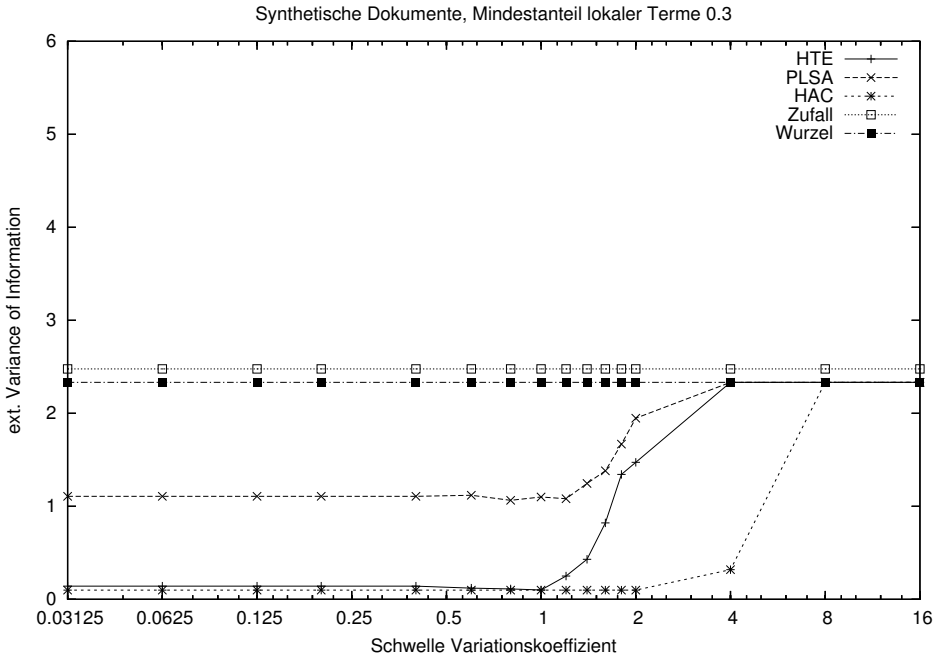

Abbildung A.36: Die ext. Variance of Information ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

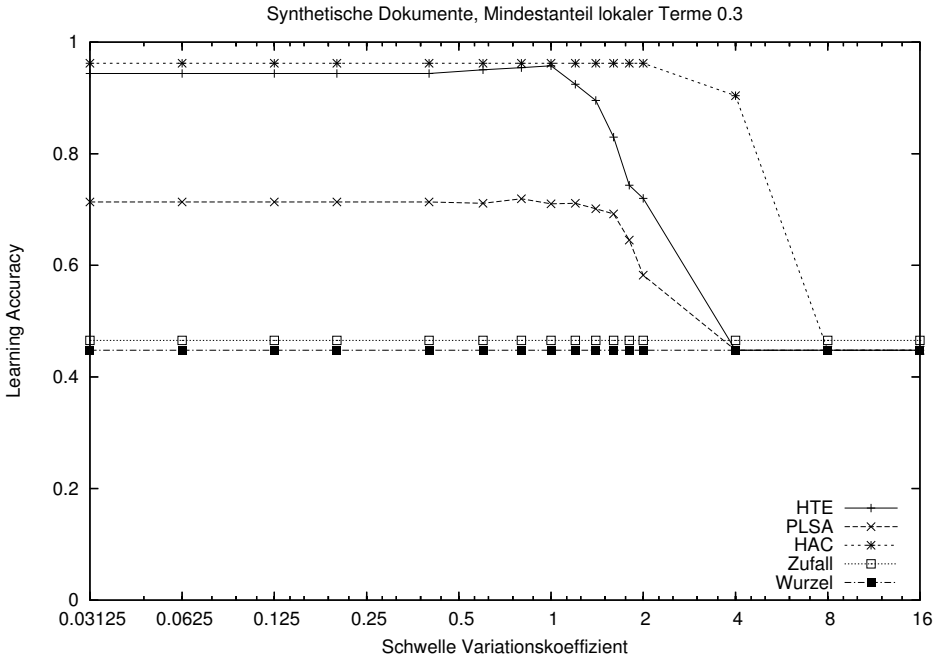

Abbildung A.37: Die Learning Accuracy ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.3)

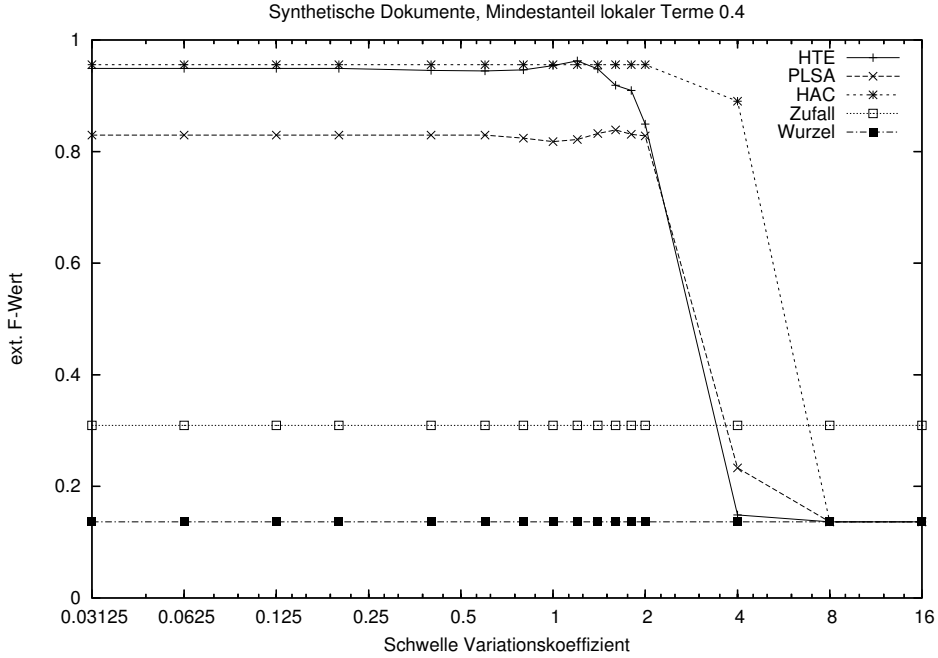

Abbildung A.38: Der F-Wert ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.4)

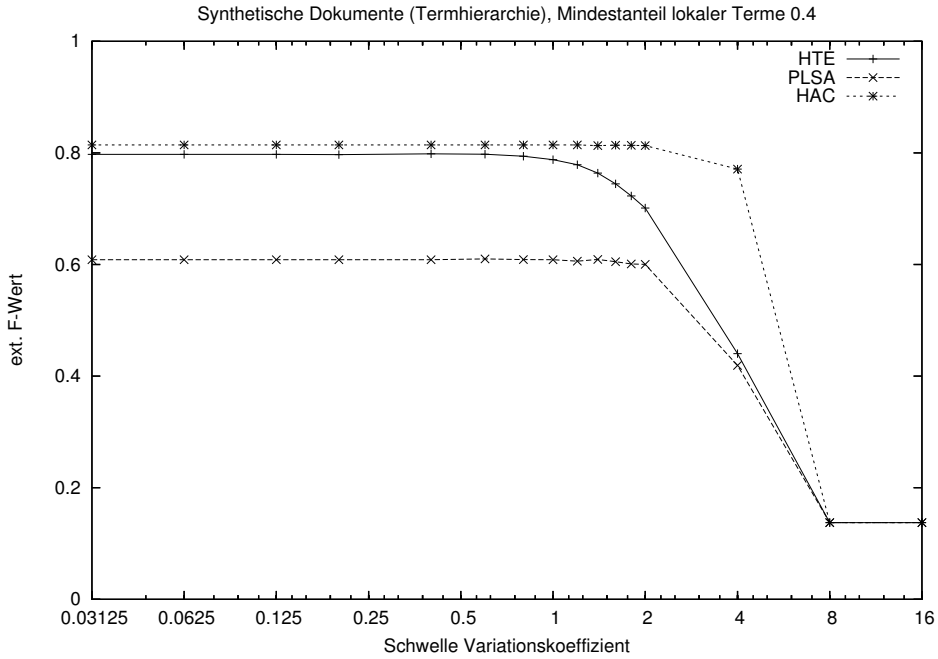

Abbildung A.39: Evaluierung der Termhierarchie: Der ext. F-Wert ggü. dem Varianzkoeffizienten (Synthetische Dokumente, Anteil lokaler Terme 0.4)

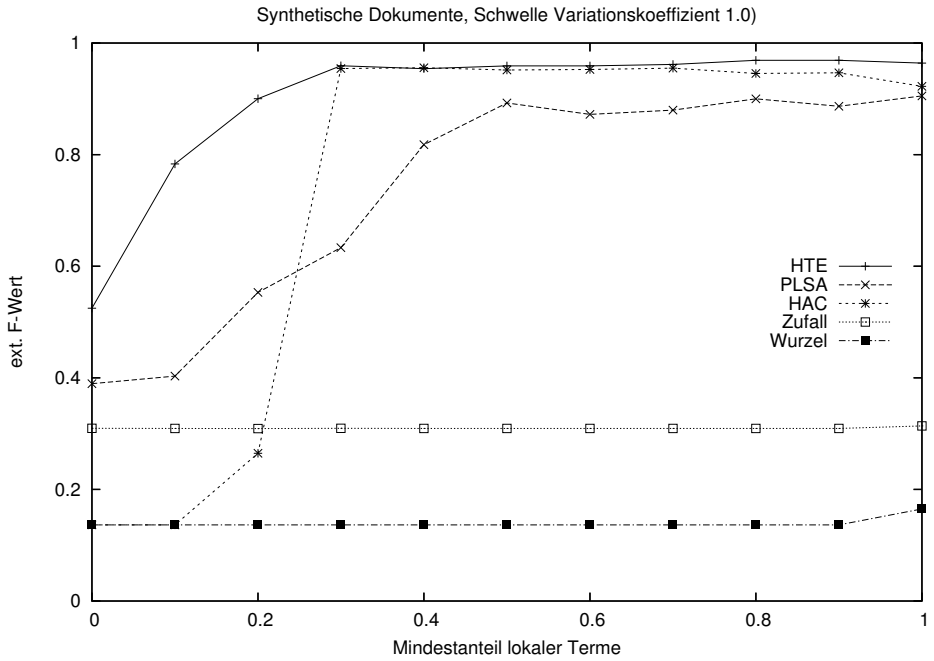

Abbildung A.40: Der ext. F-Wert ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 1.0)

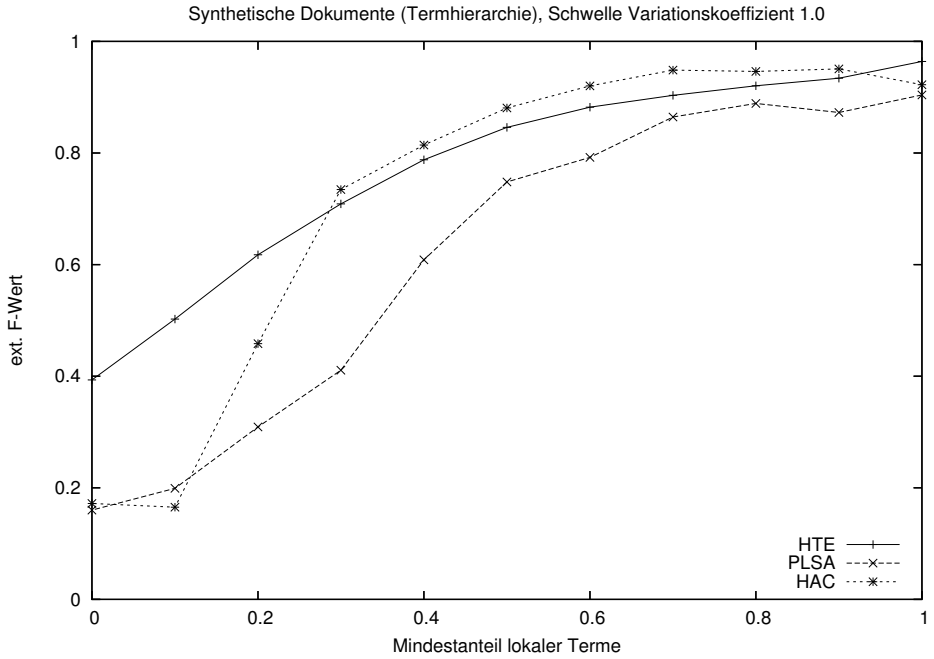

Abbildung A.41: Evaluierung der Termhierarchie: Der ext. F-Wert ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 1.0)

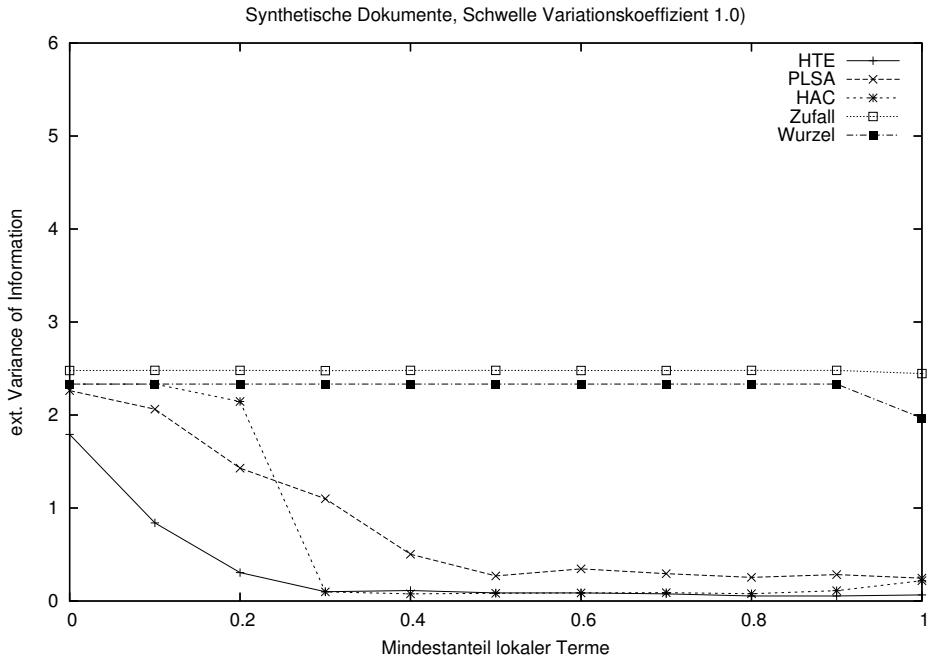

Abbildung A.42: Die ext. Variance of Information ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 1.0)

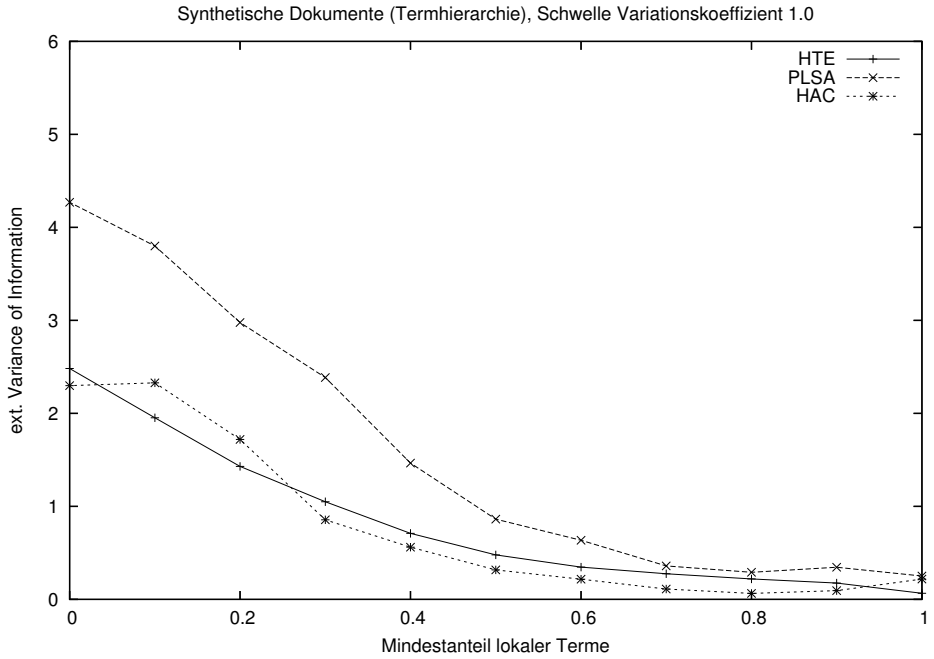

Abbildung A.43: Evaluierung der Termhierarchie: Die ext. Variance of Information ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 1.0)

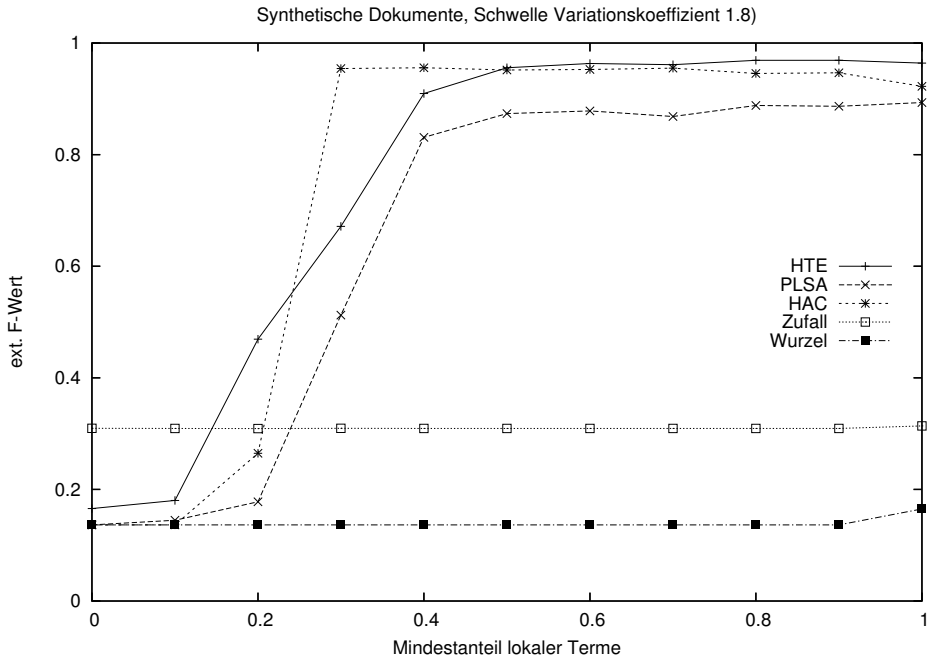

Abbildung A.44: Der ext. F-Wert ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 1.8)

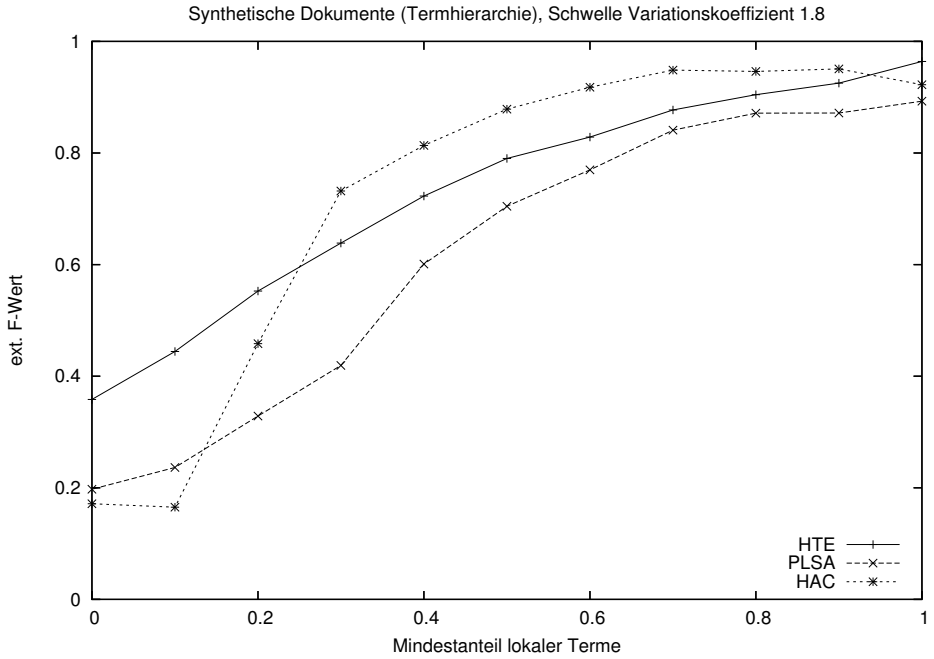

Abbildung A.45: Evaluierung der Termhierarchie: Der ext. F-Wert ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 1.8)

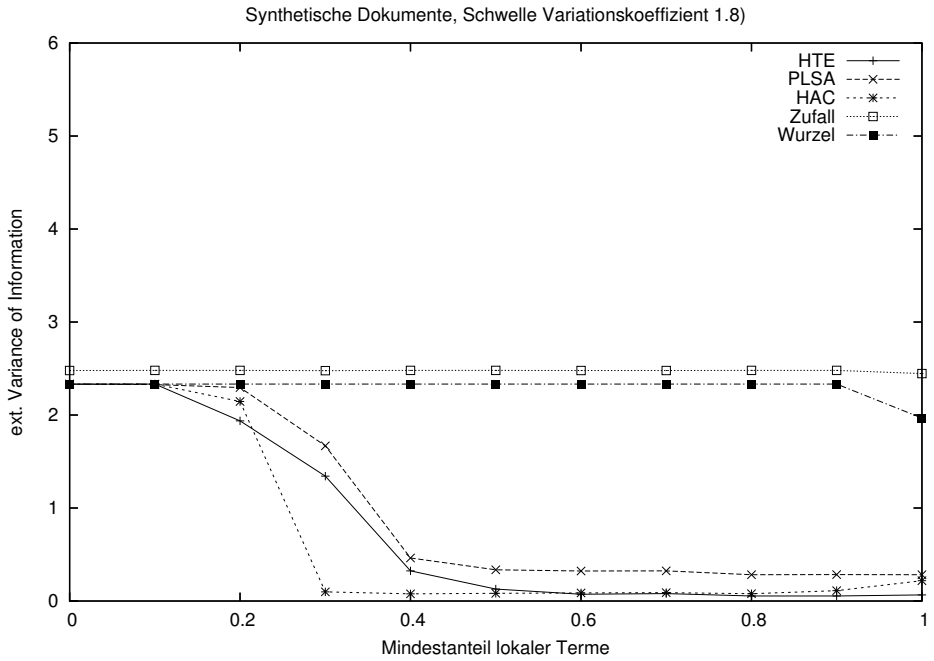

Abbildung A.46: Die ext. Variance of Information ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 1.8)

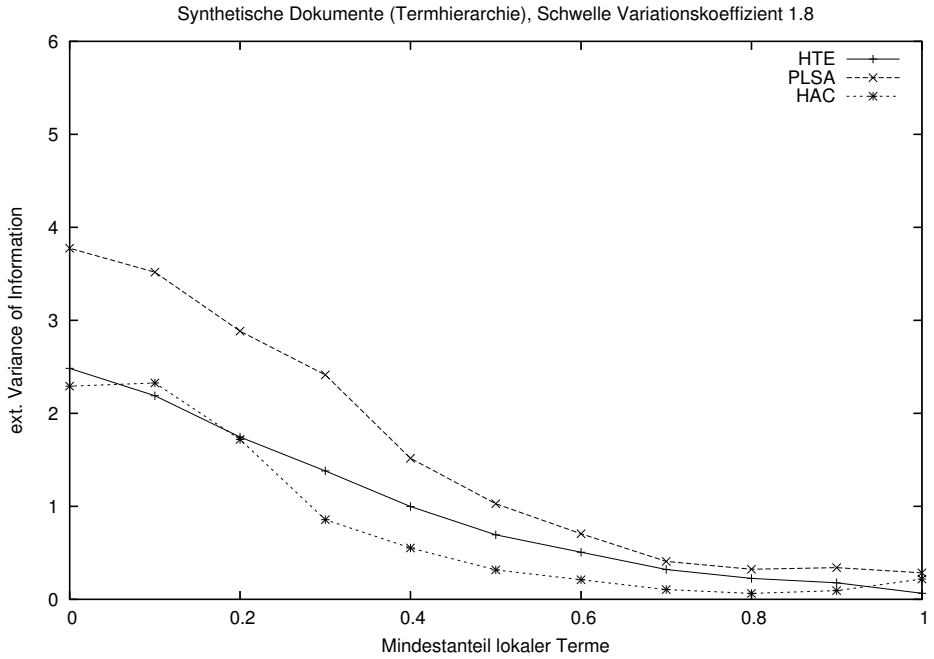

Abbildung A.47: Evaluierung der Termhierarchie: Die ext. Variance of Information ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 1.8)

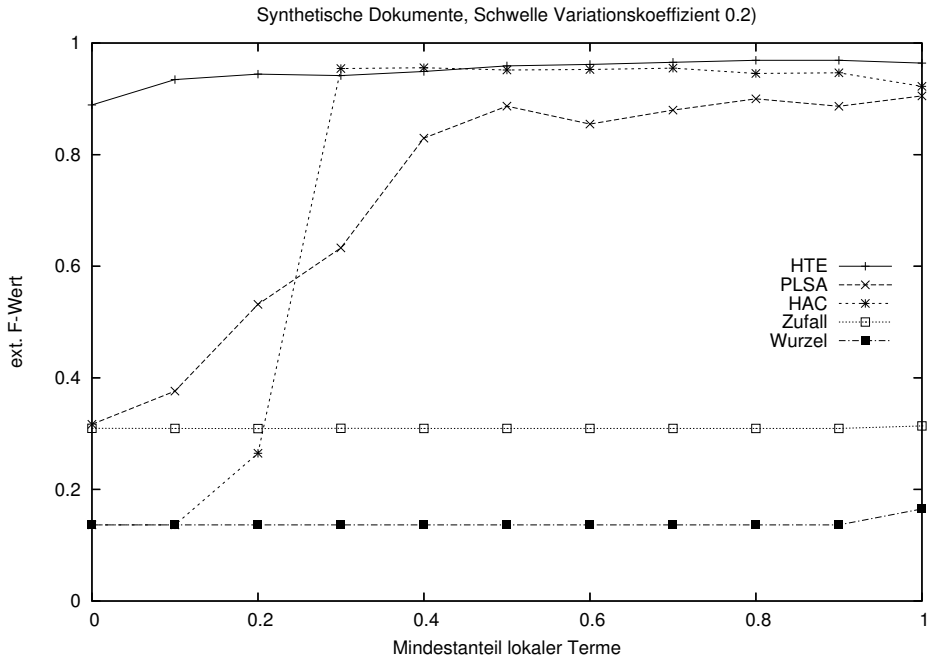

Abbildung A.48: Der ext. F-Wert ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 0.2)

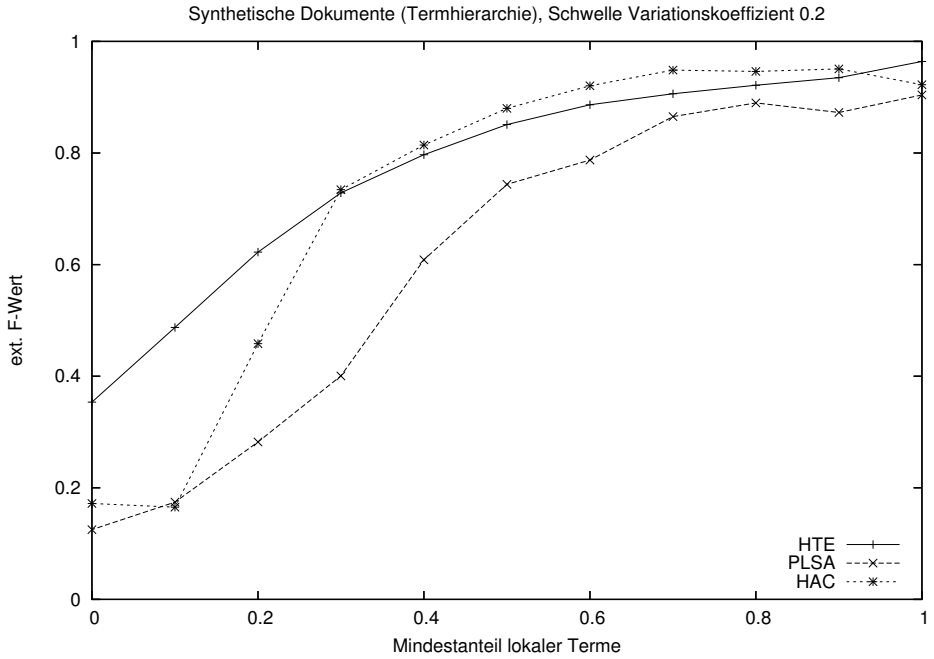

Abbildung A.49: Evaluierung der Termhierarchie: Der ext. F-Wert ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 0.2)

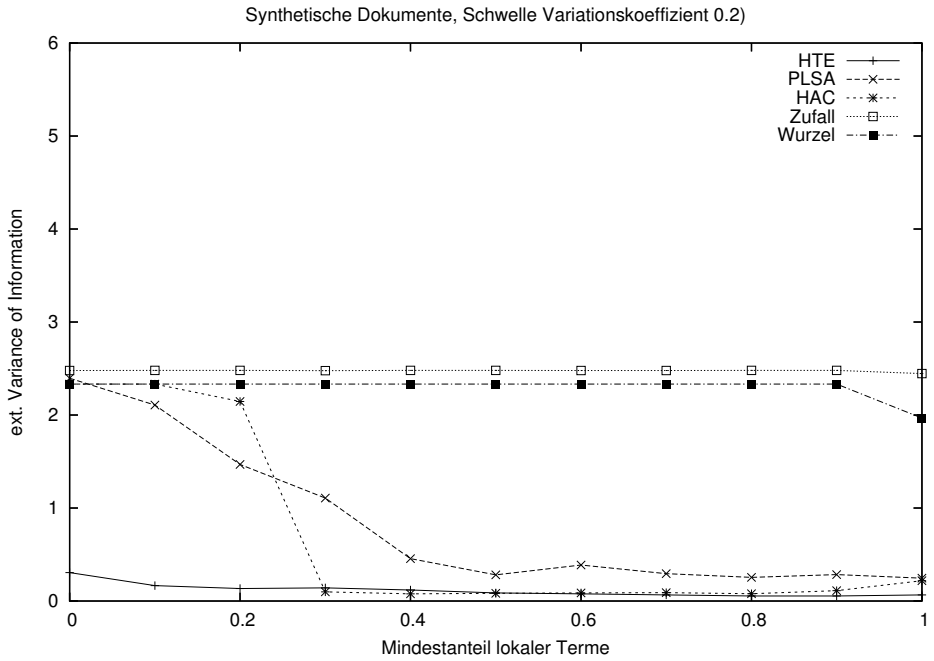

Abbildung A.50: Die ext. Variance of Information ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 0.2)

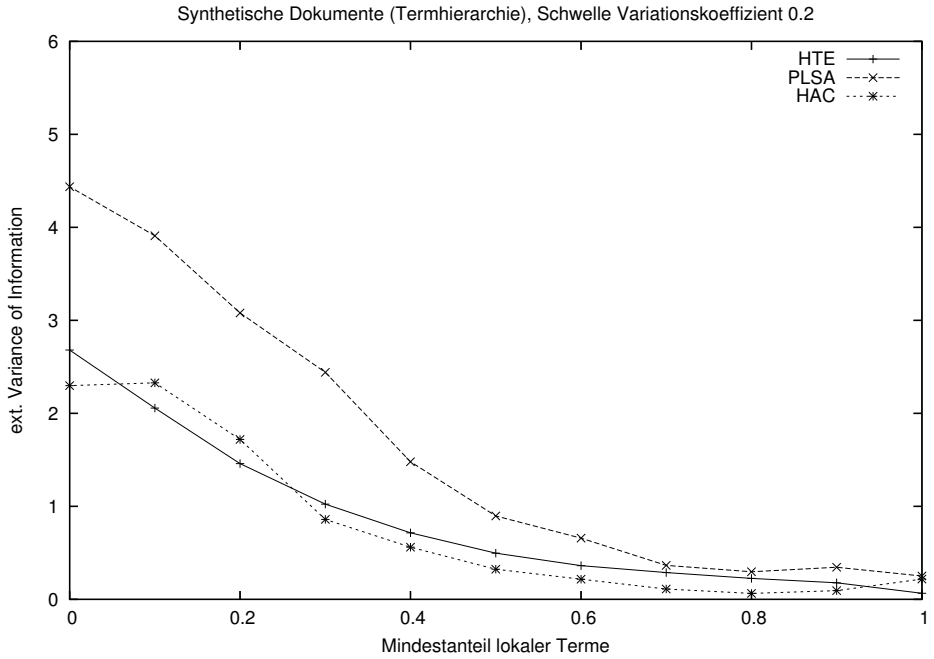

Abbildung A.51: Evaluierung der Termhierarchie: Die ext. Variance of Information ggü. dem Anteil lokaler Terme (Synthetische Dokumente, Schwelle Variationskoeffizient 0.2)

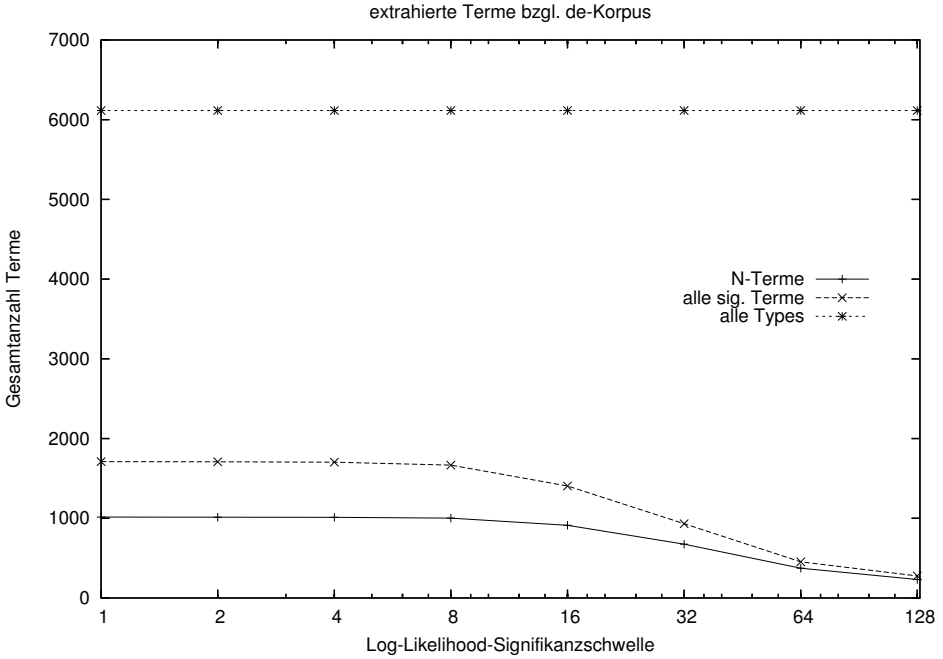

Abbildung A.52: Die Anzahl der extrahierten Types ggü. der Log-Likelihood-Signifikanzschwelle (WRT, de-Korpus, Schwelle Variationskoeffizient 1.0)

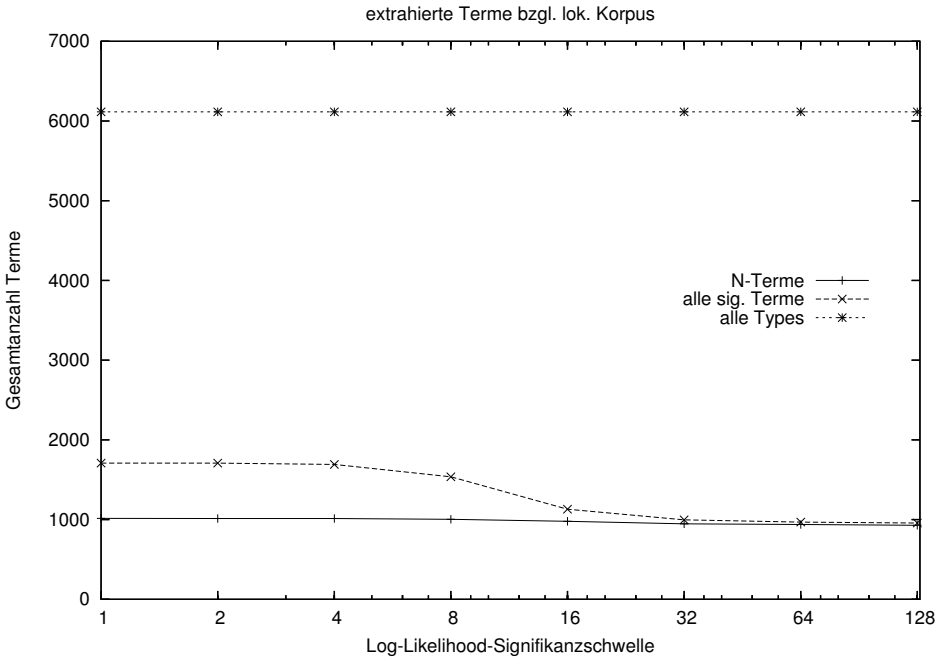

Abbildung A.53: Die Anzahl der extrahierten Types ggü. der Log-Likelihood-Signifikanzschwelle (WRT, lok. Korpus, Schwelle Variationskoeffizient 1.0)

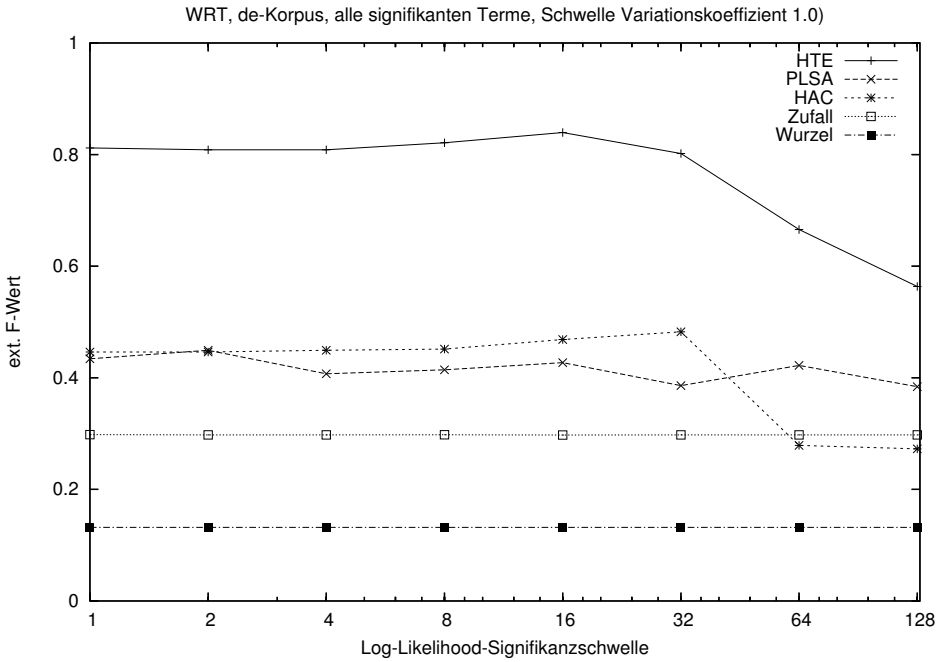

Abbildung A.54: Der ext. F-Wert ggü. der Log-Likelihood-Signifikanzschwelle (WRT, de-Korpus, Schwelle Variationskoeffizient 1.0)

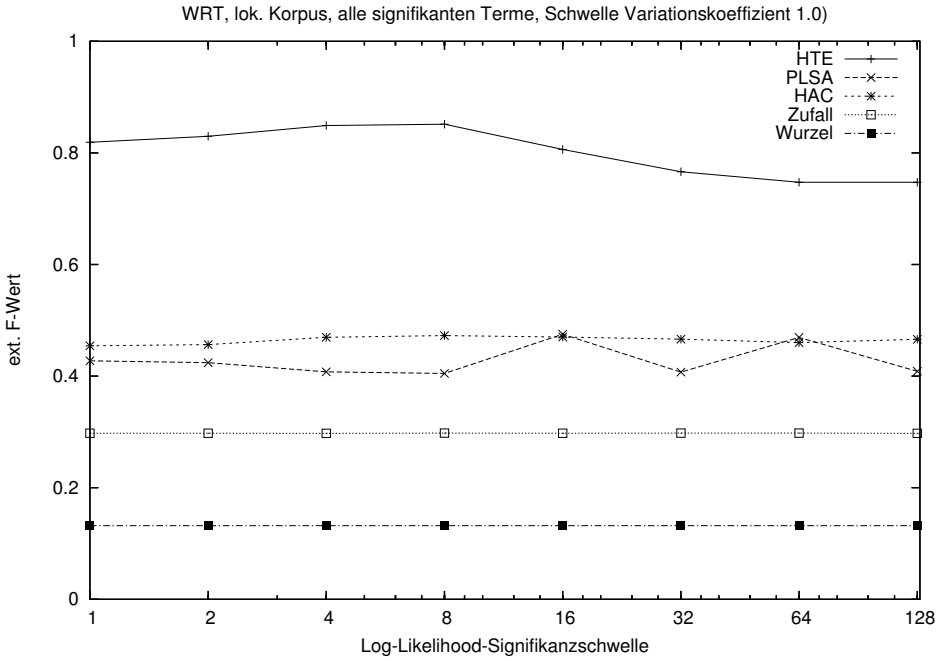

Abbildung A.55: Der ext. F-Wert ggü. der Log-Likelihood-Signifikanzschwelle (WRT, lok. Korpus, Schwelle Variationskoeffizient 1.0)

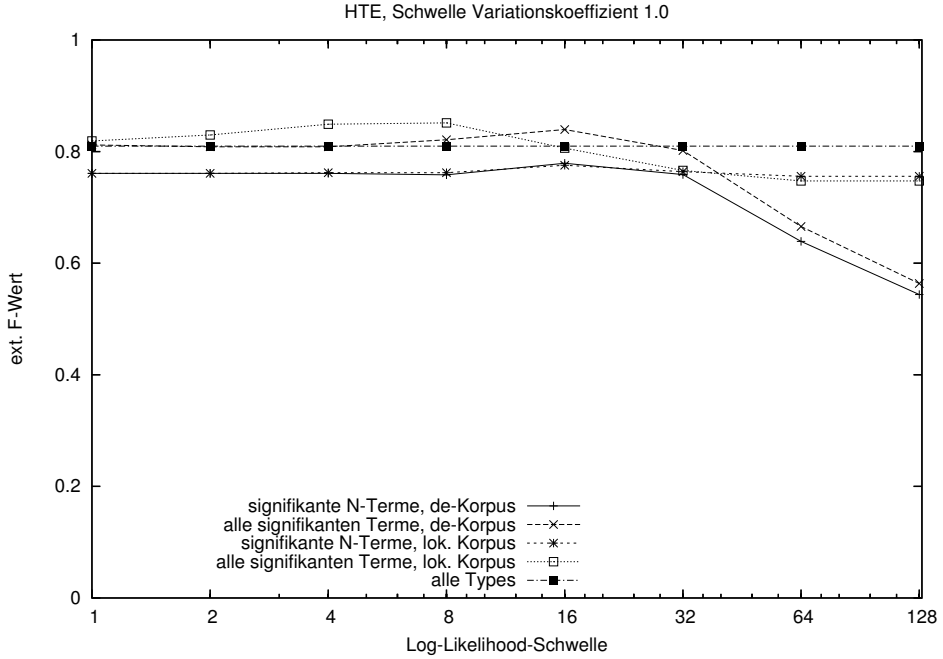

Abbildung A.56: Der ext. F-Wert ggü. der Log-Likelihood-Signifikanzschwelle (WRT, HTE, Schwelle Variationskoeffizient 1.0)

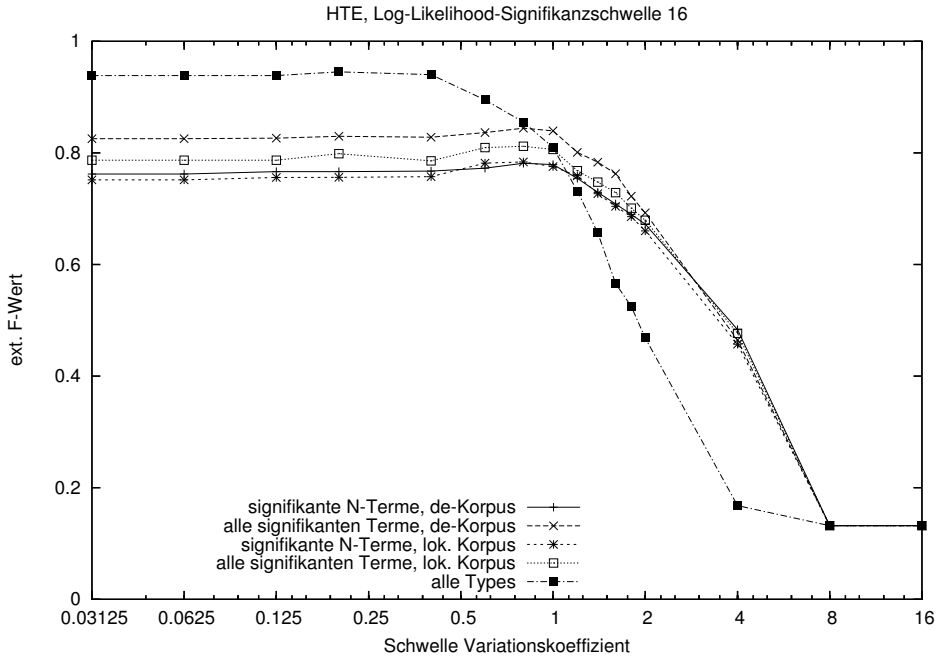

Abbildung A.57: Der ext. F-Wert ggü. dem Variationskoeffizient (WRT, HTE, Log-Likelihood-Signifikanzschwelle 16)

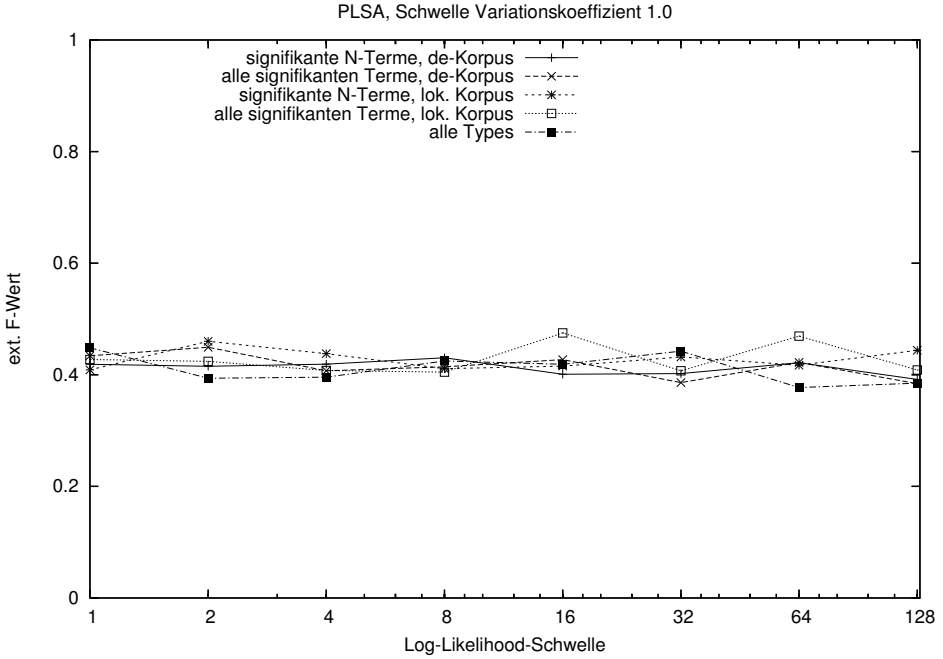

Abbildung A.58: Der ext. F-Wert ggü. der Log-Likelihood-Signifikanzschwelle (WRT, PLSA, Schwelle Variationskoeffizient 1.0)

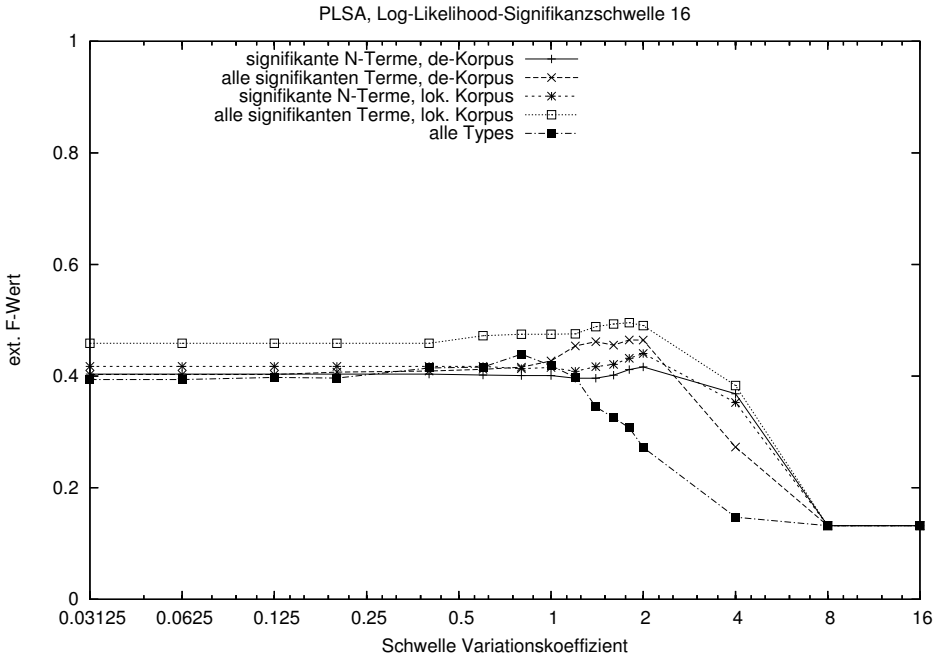

Abbildung A.59: Der ext. F-Wert ggü. dem Variationskoeffizient (WRT, PLSA, Log-Likelihood-Signifikanzschwelle 16)

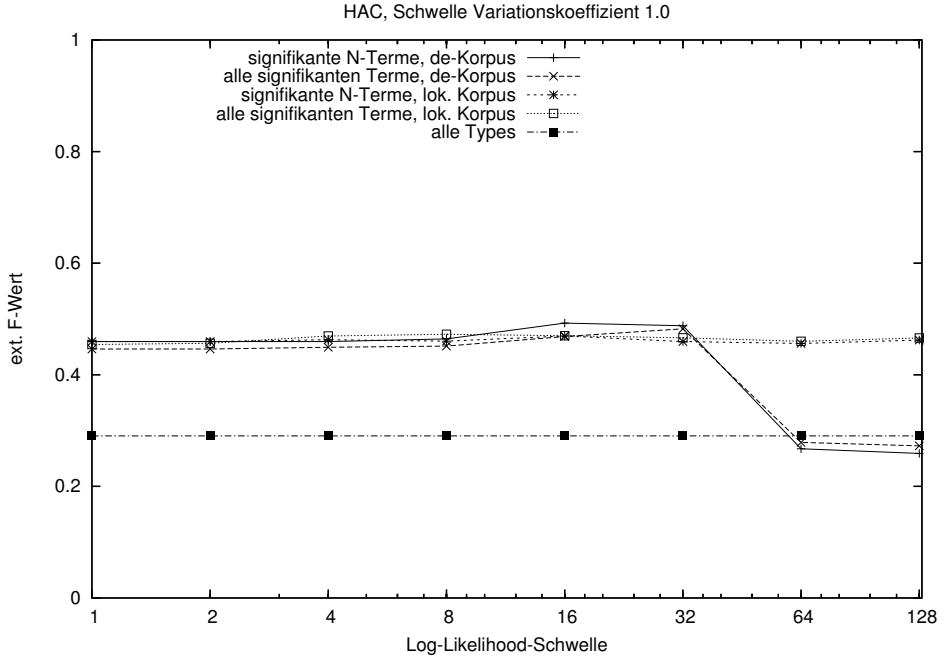

Abbildung A.60: Der ext. F-Wert ggü. der Log-Likelihood-Signifikanzschwelle (WRT, HAC, Schwelle Variationskoeffizient 1.0)

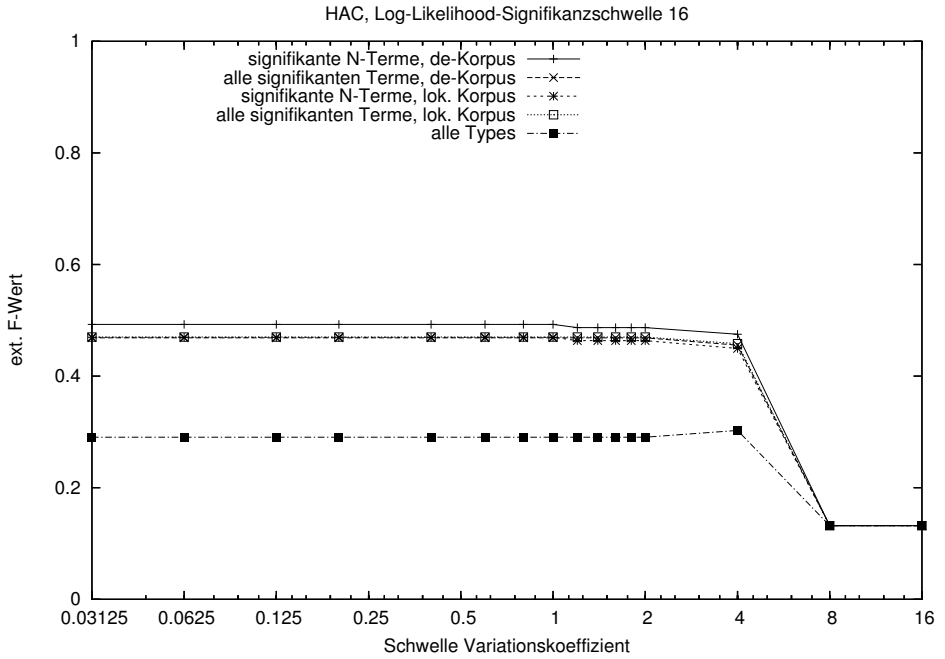

Abbildung A.61: Der ext. F-Wert ggü. dem Variationskoeffizient (WRT, HAC, Log-Likelihood-Signifikanzschwelle 16)

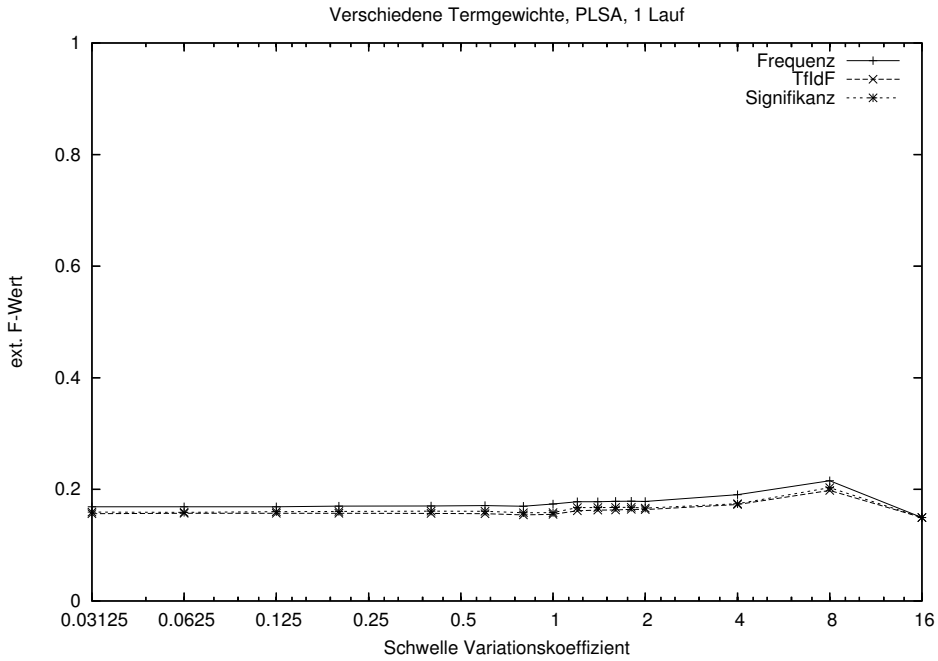

Abbildung A.62: Der F-Wert ggü. dem Varianzkoeffizienten (Spiegel, PLSA, 1 Lauf)

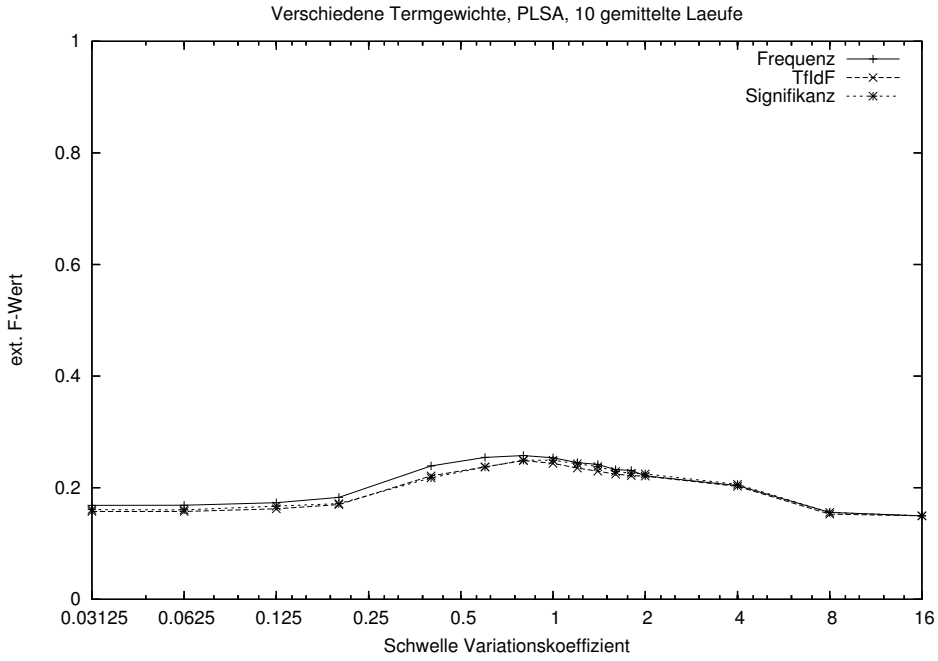

Abbildung A.63: Der F-Wert ggü. dem Varianzkoeffizienten (Spiegel, PLSA, 10 Läufe gemittelt)

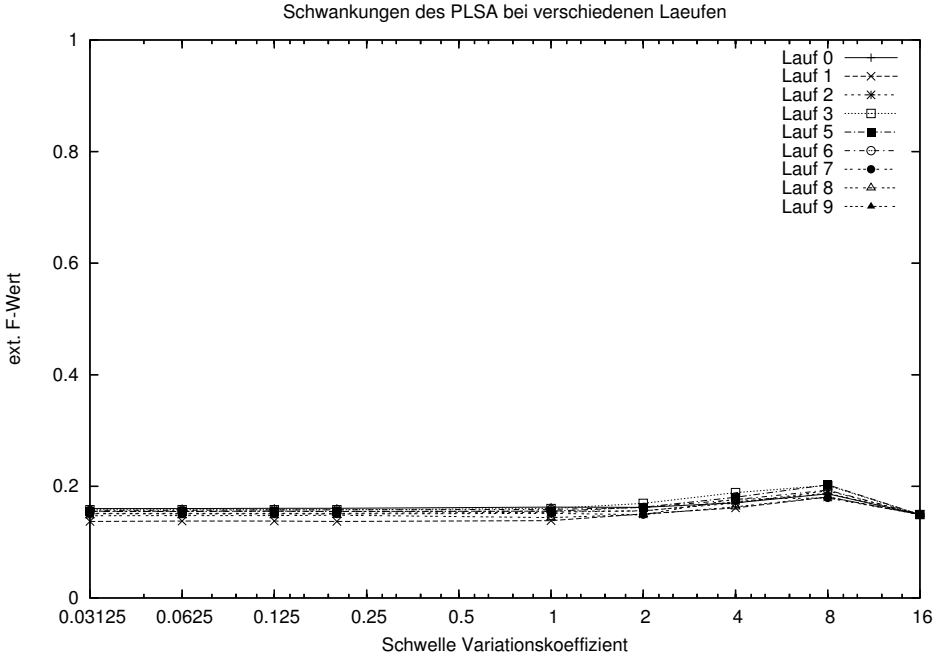

Abbildung A.64: Der ext. F-Wert ggü. dem Varianzkoeffizienten (Spiegel, PLSA, 10 Läufe nebeneinander)

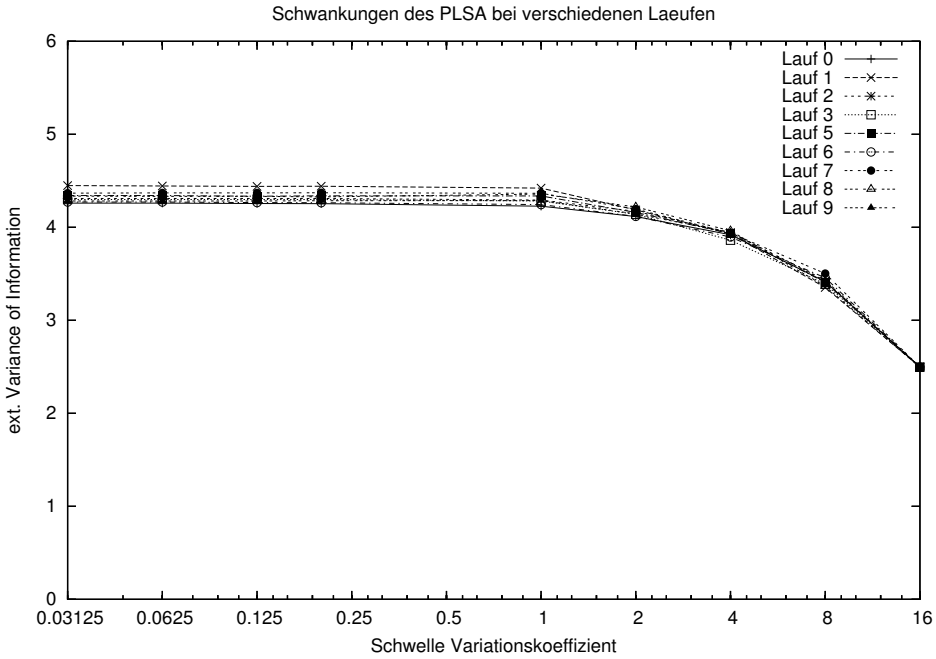

Abbildung A.65: Die ext. Variance of Information ggü. dem Varianzkoeffizienten (Spiegel, PLSA, 10 Läufe nebeneinander)

Literaturverzeichnis

[1] E. Alfonseca, S. Manandhar: Proposal for Evaluating Ontology Refinement Methods, Language Resources and Evaluation, LREC (2002)

[2] E. Alfonseca, S. Manandhar: Extending a Lexical Ontology by a Combination of Distributional Semantics Signatures, Lecture Notes in Artificial Intelligence **2473**, 1–7, Springer (2002)

[3] R. Bekkerman, R. El-Yaniv, A. McCallum: Multi-Way Distributional Clustering via Pairwise Interactions, Proceedings of the ICML 2005 (2005)

[4] C. Biemann: Ontology Learning from Text - A Survey of Methods, LDV-Forum **20**(2), 75–93 (2005)

[5] C. Biemann: Chinese Whispers – An Efficient Graph Clustering Algorithm and its Application to Natural Language Processing Problems, Proceedings of the HLT-NAACL-06 Workshop on Textgraphs (2006)

[6] D. Blei, T. Griffith, M. Jordan, J. Tenenbaum: Hierarchical Topic Models and the Nested Chinese Restaurant Process, Advances in Neural Information Processing Systems **16**, MIT Press (Cambridge 2004)

[7] D. M. Blei, A. Y. Ng, M. I. Jordan: Latent Dirichlet Allocation, Journal of Machine Learning Research **3**, 993–1022 (2003)

[8] Ph. Cimiano, A. Hotho, St. Staab: Comparing Conceptual, Divisive and Agglomerative Clustering for Learning Taxonomies from Text, Proceedings of the European Conference on Artificial Intelligence, 435–439 (2004)

[9] Ph. Cimiano, A. Hotho, St. Staab: Learning Concept Hierarchies from Text Corpora Using Formal Concept Analysis Journal of Artificial Intelligence Research **24**, 305–339 (2005)

[10] Ph. Cimiano, A. Hotho, St. Staab: Learning Concept Hierarchies from Text with a Guided Hierarchical Clustering Algorithm, Proceedings of the ICML 2005 (2005)

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

[12] C. Fellbaum: *Wordnet: An Electronic Lexical Database*, MIT Press (Cambridge 1998)

[13] U. Hahn, K. Schnattinger: Towards Text Knowledge Engineering, AAAI/IAAI, 524–531 (1998)

[14] G. Heyer, U. Quasthoff, T. Wittig: *Text Mining: Wissensrohstoff Text – Konzepte, Algorithmen, Ergebnisse*, W3L-Verlag (Herdecke, Dortmund 2003)

[15] T. Hofmann: Unsupervised Learning by Probabilistic Latent Semantic Analysis, Machine Learning **42**, 177–196 (2001)

[16] D. E. Knuth: *The Art of Computer Programming*, Volume **2** – Seminumerical Algorithms, Addison-Wesley (1998)

[17] R. Lokaiczyk: Clustering und Klassifikation von Personenamen, Diplomarbeit, Institut für Informatik, Universität Leipzig (2005)

[18] P. Makagonov, A. R. Figueroa, K. Sboychakov, A. Gelbukh: Learning a Domain Ontology from Hierachically Structured Texts, Proceedings of the ICML 2005 (2005)

[19] P. Makagonov, A. R. Figueroa: Study of Knowledge Evolution in Parallel Computing by Short Texts Analysis, CIARP 2004, 439–445 (2004)

[20] C. D. Manning, H. Schütze: *Foundations of Statistical Natural Language Processing*, MIT Press (Cambridge 1999)

[21] M. Meila: Comparing Clusterings, Technical Report **418**, Department of Statistics, University of Washington (2002)

[22] M. Meila: Comparing Clusterings - An Axiomatic View, Proceedings of the ICML 2005 (2005)

[23] Z. Michalewicz: *Genetic Algorithms + Data Structures = Evolution Programs*, Springer (Berlin 1999)

[24] V. Pekar, M. Krkoska: Weighting Distributional Features for Automatic Semantic Classification of Words, Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP-03), 369–373 (2003)

[25] V. Pekar, M. Krkoska, St. Staab: Feature Weighting for Co-occurrence-based Classification of Words, Proceedings of the 20th Conference on Computational Linguistics COLING-2004 (2004)

[26] V. Pekar, St. Staab: Taxonomy Learning – Factoring the Structure of a Taxonomy into a Semantic Classification Decision, Proceedings of the 19th International Conference on Computational Linguistics, 786–792 (2002)

[27] V. Pekar, St. Staab: Word Classification Based on Combined Measures of Distributional and Semantic Similarity, Proc. Research Notes of the 10th Conference of the European Chapter of the Association for Computational Linguistics, 147–150 (2003)

[28] D. Pullwitt: Explorative Analyse von Textkorpora mit Clusterverfahren, Dissertation, Institut für Informatik, Universität Leipzig (2003)

[29] A. I. Schein, A. Popescul, L H. Ungar: PennAspect: Two-Way Aspect Model Implementation, http://www.cis.upenn.edu/datamining /software_dist/PennAspect/index.html

[30] N. Slonim, N. Tishby: Document Clustering Using Word Clusters via the Information Bottleneck Method, Research and Development in Information Retrieval, 208–215 (2000)

[31] N. Tishby, F. C. Pereira, W. Bialek: The Information Bottleneck Method, Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing, 368–377 (1999)

[32] J. Weeds, D. Weir, D. McCarthy, Characterising Measures of Lexical Distributional Similarity, Proceedings of CoLing 2004 (2004)

[33] H. F. Witschel: Text, Wörter, Morpheme – Möglichkeiten einer automatischen Terminologie-Extraktion, Diplomarbeit, Institut für Informatik, Universität Leipzig (2004)

[34] WordNet 2.1 Database Statistics, http://wordnet.princeton.edu/man/wnstats.7WN

[35] K. Yu, Sh. Yu, V. Tresp: Soft clustering on graphs, Advances in Neural Information Processing Systems **18** (2005)

Erklärung: „Ich versichere, daß ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe.“

Leipzig, 11.01.2007