Eine offene Architektur für die agentenbasierte, adaptive, personalisierte Informationsfilterung

vorgelegt von
Diplom-Informatiker
Andreas Lommatzsch

Von der Fakultät IV – Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
– Dr.-Ing. –

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender:	Prof. Dr.-Ing. A. Wolisz
Berichter:	Prof. Dr.-Ing. S. Albayrak
Berichter:	Prof. Dr. M. Opper

Tag der wissenschaftlichen Aussprache: 20. April 2009

Berlin 2009

D 83

Inhaltsverzeichnis

1 Einleitung **5**
- 1.1 Motivation . . . 5
  - 1.1.1 Ziele und Herausforderungen . . . 6
  - 1.1.2 Suchmaschinen und Filterstrategien . . . 7
  - 1.1.3 Personalisierung . . . 7
  - 1.1.4 Koordinationsstrategien . . . 8
- 1.2 Forschungsgebiete . . . 8
  - 1.2.1 Gewählter Ansatz . . . 9
  - 1.2.2 Entwickelte Verfahren . . . 9
- 1.3 Aufbau der Arbeit . . . 10

2 Problembeschreibung **13**
- 2.1 Szenarien . . . 13
  - 2.1.1 Wissenschaftler . . . 13
  - 2.1.2 Lernende . . . 15
  - 2.1.3 Geschäftsleute . . . 17
  - 2.1.4 Senioren . . . 17
- 2.2 Anforderungen an personalisierte IR-Systeme . . . 18
- 2.3 Probleme klassischer Informationssysteme . . . 20
- 2.4 Zusammenfassung . . . 23

3 Existierende Systeme und Verfahren **25**
- 3.1 Such- und Metasuchmaschinen . . . 25
  - 3.1.1 Google . . . 25
  - 3.1.2 MetaGer . . . 26
- 3.2 Themenbereichsfokussierte Suche . . . 27
  - 3.2.1 Verzeichnisdienste . . . 28
  - 3.2.2 Themenbereichsspezialisierte Suchmaschinen . . . 30
  - 3.2.3 Themenorientierte Ergebnispräsentation . . . 31
  - 3.2.4 Erweitern von Suchanfragen . . . 32
  - 3.2.5 Diskussion . . . 34

3.3 Personalisierte Informationssysteme 34
3.3.1 Ziele und Strategien der Personalisierung 34
3.3.2 Personalisierung im Information Retrieval 35
3.3.3 Probleme bei der Personalisierung 36
3.4 Systeme für die personalisierte Suche 37
3.4.1 Web Personae 37
3.4.2 WebACE 38
3.4.3 Syskill and Webert 39
3.4.4 Letizia 40
3.4.5 Let's Browse 40
3.4.6 Web Watcher 41
3.4.7 Keyword Highlighting 42
3.4.8 WebMate 42
3.4.9 Diskussion der Informationssysteme 43
3.5 Verteilte Systeme 43
3.5.1 Multi-Agentensysteme 44
3.5.2 Ein Framework für die Entwicklung von intelligenten verteilten Systemen 48
3.6 Zusammenfassung 50

4 Die Architektur des Systems **51**
4.1 Aufgabenbereiche und Funktionen 51
4.2 Ablauf einer personalisierten Suche 55
4.3 Schichtenmodell 57
4.4 Abgrenzung zu bestehenden Systemen 61
4.5 Zusammenfassung 62

5 Ermitteln relevanter Dokumente **65**
5.1 Auswahl von Informationsquellen 65
5.2 Optimieren von Anfragen 68
5.3 Kombinieren von Anfrageoptimierungsverfahren 75
5.4 Metadaten und Bewertungen 78
5.5 Zusammenfassung 81

6 Zusammenstellen von Ergebnismengen **83**
6.1 Gründe für die Kombination von Filtern 84
6.1.1 Collection Fusion 84
6.1.2 Data Fusion 85
6.1.3 Delegieren von Anfragen 86
6.2 Evaluieren von Ergebnissen 87
6.2.1 Kriterien zum Bewerten von Filteralgorithmen 88

6.2.2 Die Exaktheit 89
6.2.3 Die Vollständigkeit 94
6.2.4 Auf einer Referenzstrategie basierende Kriterien 95
6.2.5 Personalisierung 101
6.2.6 Vielfältigkeit der Ergebnismenge 101
6.2.7 Diskussion der Evaluationsstrategien 102
6.3 Verwendete Benchmarkdatensätze 102
6.4 Zusammenstellen von Ergebnismengen 104
6.4.1 Wahlverfahren 104
6.4.2 Regeln und Heuristiken 105
6.4.3 Strategien der globalen Optimierung 105
6.4.4 Gruppieren von Ergebnissen 105
6.4.5 Diskussion der Strategien zum Zusammenstellen von Ergebnismengen 106
6.5 Kombinieren von Bewertungen 107
6.5.1 Kombination boolescher Bewertungen 107
6.5.2 Kombination numerischer Bewertungen 110
6.5.3 Kombination rangbasierter Bewertungen 113
6.5.4 Wahl der Kombinationsstrategie 115
6.6 Anpassen der Kombinationstrategien für das IR 116
6.6.1 Konvertieren von Bewertungen 116
6.6.2 Skalierung und Gewichtung 118
6.6.3 Auswählen von Filteragenten 119
6.7 Zusammenfassung 125

7 Algorithmen und Evaluation **127**
7.1 Zusammenstellen von Agentengremien 127
7.1.1 Analysiertes Szenario 127
7.1.2 Inkrementelles Zusammenstellen von Gremien 129
7.1.3 Agentengremien mit Gewichtungen 132
7.1.4 Optimieren von Gewichtungen 133
7.1.5 Inkrementelles Verkleinern von Agentengremien 136
7.1.6 Erstellen von Gremien auf der Basis von Clustern . . . 138
7.1.7 Ermitteln von Gewichtungen auf Basis von Regression 143
7.1.8 Delegieren von Anfragen 146
7.2 Verlässliche Agentengremien 151
7.2.1 Berechnen der Verlässlichkeit 152
7.2.2 Analyse der bezüglich MAP optimierten Gremien . . . 153
7.2.3 Optimieren der Verlässlichkeit 153
7.3 Adaptive Agentengremien 155
7.3.1 Inkrementelle Benutzerbewertungen 155

7.3.2 Agentengremien unterschiedlicher Größe 158
7.3.3 Integration eines neuen Agenten 162
7.3.4 Ausfall eines Agenten 163
7.4 Personalisierte Bewertungsmaße 165
7.5 Zusammenfassung 167

8 Personalisierte Informationsagenten 169
8.1 Überblick über die entwickelten Systeme 169
8.2 Implementierung und Besonderheiten der Systeme 171
8.3 Strategien zum Analysieren des Informationssystems 173
8.4 Erstellen und Optimieren von Anfragen 174
8.4.1 Optimieren von Anfragen 174
8.4.2 Auswahl von Anfrageoptimierungsstrategien 188
8.4.3 Erstellung von Anfragen aus Dokumenten 189
8.4.4 Zusammenfassung und Bewertung 190
8.5 Filter- und Bewertungsstrategien 191
8.6 Zusammenstellen und Darstellen von Ergebnissen 192
8.7 Zusammenfassung 196

9 Zusammenfassung und Ausblick 199
9.1 Multi-Agentensysteme 199
9.2 Systemarchitektur 200
9.3 Kombination von Filterstrategien 200
9.4 Personalisierung und Adaption 201
9.5 Realisierte Informationssysteme 201
9.6 Alternative Anwendungsgebiete 202
9.7 Ausblick 202

A Strategien des LETOR-TD2004**-Datensatzes 205**

B Glossar 207

Zusammenfassung

Im Informationszeitalter stellt das Auffinden von auf die individuellen Benutzerpräferenzen abgestimmten Informationen einen entscheidenden Erfolgsfaktor dar. Um die Relevanz von Informationen zu beurteilen, müssen unterschiedliche Bewertungskriterien herangezogen werden, die jeweils auf die persönlichen Vorlieben des Benutzers und den Kontext der Suche abgestimmt sind.

Im Rahmen dieser Arbeit werden die Anforderungen an ein adaptives personalisiertes Informationssystem motiviert und eine Architektur entwickelt, die die gestellten Anforderungen erfüllt. Die Architektur ermöglicht die Integration heterogener Datenquellen, die Berücksichtigung einer großen Anzahl von Bewertungskriterien sowie das Kombinieren der Bewertungen, um optimierte Dokumentenmengen zusammenzustellen. Die Auswahl und Kombination der für eine Informationsanfrage eingesetzten Algorithmen und Komponenten erfolgt jeweils unter Berücksichtigung des jeweiligen Suchszenarios, der verfügbaren Ressourcen und der individuellen Präferenzen des Benutzers. Die Architektur ermöglicht es, die von unterschiedlichen Quellen stammenden Dokumente umfassend (d. h. inhaltsbasiert, kollaborativ, personalisiert, auf Basis von semantischem Wissen) zu bewerten und die Dokumente für eine optimale Präsentation mit zusätzlichen Informationen (Metadaten) anzureichern. Auf Basis der Metadaten und der Bewertungen werden die Dokumente bezüglich verschiedener Gütefunktionale optimierter Ergebnismengen zusammengestellt.

Schwerpunkte der Arbeit bilden Verfahren zum Bewerten von Dokumenten und Methoden zum Kombinieren von Bewertungen, die von unterschiedlichen Bewertungsstrategien stammen. Darüber hinaus werden Verfahren zur Unterstützung des Benutzers beim Erstellen und Optimieren von Informationsanfragen entwickelt. Die Leistungsfähigkeit der Verfahren wird anhand von Benutzerfeedback und Benchmarkdatensätzen evaluiert.

Auf der Basis der entworfenen Architektur wurden unter Verwendung eines Multi-Agentenframeworks für verschiedene Anwendungsbereiche optimierte Informationssysteme entwickelt. Es wird gezeigt, dass die Informati-

onssysteme die gestellten Anforderungen bezüglich Qualität der Ergebnisse, Offenheit für neue Algorithmen sowie der Adaptivität an individuellen Benutzerpräferenzen und den Suchkontext erfüllen. Das einheitliche Bewerten von aus unterschiedlichen Quellen stammenden Informationen und die adaptive Kombination der Bewertungen ermöglicht das zuverlässige Ermitteln von für den Benutzer relevanten Dokumenten.

Abstract

As today the amount of accessible information is overwhelming and still growing, the intelligent and personalized filtering of available information constitutes a main challenge. In order to assess the relevance of a document a wide variety of criteria must be considered while taking into account individual user preferences and the context of the search.

This thesis analyzes typical information retrieval scenarios and derives the requirements for an adaptive personalized information system. Since existing information systems possess various disadvantages, the architecture for an information system is developed that allows the integration of heterogeneous sources, various relevance measurement methods as well as strategies for assembling optimized result sets. The system components are modeled as intelligent agents, which enables the adaption to future requirements and the integration of new components at runtime. Information requests are delegated to the most appropriate agents, in consideration of the individual user preferences, the available system resources and the search context. The proposed architecture ensures that documents from heterogeneous information sources (such as search engines, recommender systems, and databases) are evaluated uniformly. This is accomplished by employing diverse rating strategies (such as semantic, collaborative, content based and personalized rating algorithms). For an optimal presentation, the documents are enriched with various metadata so as to enable the user to understand and exploit the results of a particular query. On the basis of the comprehensive set of ratings, document sets are assembled optimized according to personalized quality criteria.

Furthermore, this thesis analyzes components for assessing documents and methods for combining various rating strategies in detail. Moreover, algorithms aiming to support the user with the generation and optimization of queries are discussed. The methods and algorithms are evaluated both with regard to user feedback and benchmark datasets.

Based on the proposed architecture, several information systems are implemented optimized for specific domains and scenarios. The evaluation of

these systems shows that the developed framework meets the requirements according to flexibility, result quality and personalization. The generation of optimized result sets which are based on the comprehensive relevance criteria ensures personalized high quality search results.

Kapitel 1

Einleitung

1.1 Motivation

Im Informationszeitalter wächst die Menge der digital verfügbaren Daten und Dokumente rasant. Das Auffinden von relevanten Informationen im Dschungel der verfügbaren Daten gewinnt damit eine wachsende Bedeutung. Die Relevanz von Informationen hängt von vielfältigen Kriterien ab, wie den Kenntnissen und Erwartungen des Benutzers sowie dem jeweiligen Suchkontext.

Die derzeit verfügbaren Systeme zur Informationssuche finden oft nur bedingt zu den Interessen des Benutzers passende Informationen, da sie nur eine beschränkte Menge von Datenquellen integrieren, auf wenige Bewertungskriterien beschränkt sind und die individuellen Präferenzen des Benutzers nur unzureichend berücksichtigen. Zudem bieten sie oft keine Unterstützung beim Definieren des Informationsbedarfes und passen sich nicht an die Wünsche des Benutzers an. Bei der Nutzung von derzeit verfügbaren Informationssystemen benötigt der Benutzer viel Zeit, Informationen von verschiedenen Informationssystemen zusammenzusuchen und sich mit den Besonderheiten der einzelnen Systeme vertraut zu machen. Zudem verwenden die Informationssysteme jeweils eigene Bewertungsstrategien, die individuelle Benutzerpräferenzen nicht berücksichtigen, so dass der Benutzer die zu einer Anfrage ermittelten Informationen prüfen muss, inwieweit diese zu seinem Interessen und Erwartungen passen.

Eine Möglichkeit die Probleme der verfügbaren Systeme zu überwinden, besteht in der Realisierung eines Informationssystems, das unterschiedliche Informationsquellen integriert, Informationen anhand vielfältiger, auf die Präferenzen des Benutzers abgestimmten Kriterien bewertet und optimale Ergebnismengen zusammenstellt. Dem Benutzer wird dabei ein elektroni-

scher Assistent zur Seite gestellt, der die Interessen und Kenntnisse des Benutzers erlernt und ihn aktiv bei der Suche nach Informationen unterstützt. Aufgaben, bei denen ein Assistent den Benutzer unterstützen kann, sind:

- Das Erstellen und Optimieren von Suchanfragen
- Das Weiterleiten und Anpassen von Suchanfragen an mehrere Informationsquellen (z. B. Suchmaschinen und Empfehlungssysteme)
- Das Prüfen der von den einzelnen Informationsquellen ermittelten Dokumente (hinsichtlich der für den Benutzer relevanten Merkmale)
- Das Zusammenführen der von den einzelnen Informationsquellen ermittelten Dokumente zu einer optimierten Ergebnismenge
- Das Aufbereiten der Suchergebnisse, so dass sie (unter Berücksichtigung der individuellen Erfahrungen und Wünsche) für den Anwender einfacher genutzt werden können
- Das regelmäßige Suchen nach neuen relevanten Informationen.

Intelligente Assistenten können den Benutzer bei der Informationssuche unterstützen, indem sie Routineaufgaben übernehmen, den Benutzer bei der Optimierung von Suchanfragen unterstützen und die Ergebnismenge an die individuellen Präferenzen des Benutzers anpassen.

1.1.1 Ziele und Herausforderungen

Ziel der Arbeit ist es, eine Architektur zu entwickeln, die die vielgestaltigen Anforderungen an ein modernes Informationssystem erfüllt. Schwerpunkte bilden dabei die Integration heterogener Informationsquellen, das Einbeziehen vielfältiger Dokumentenbewertungsalgorithmen, die Adaptivität an sich verändernde Benutzerinteressen sowie das Berücksichtigen individueller Benutzerpräferenzen. Das Informationssystem soll es ermöglichen, neue Komponenten zu integrieren und den Ausfall von Komponenten geeignet zu behandeln. Darüber hinaus werden Algorithmen entwickelt, um den Benutzer bei der Formulierung seines Informationsbedürfnisses zu unterstützen und um die Filterergebnisse auf die individuellen Interessen und Gütekriterien des Benutzers abzustimmen.

Die Herausforderungen bei der Realisierung eines offenen, adaptiven, personalisierten Informationssystems bestehen darin, dass vielfältige Komponenten und Algorithmen integriert werden müssen sowie die für Informationsanfragen jeweils am besten geeigneten Komponenten ausgewählt werden müssen. Zentrale Probleme stellen dabei das Management einer großen

Anzahl von verteilten Komponenten sowie Strategien zur Koordination dar. Die Koordinationsstrategien müssen dabei insbesondere die Integration neuer Komponenten, den Ausfall von Komponenten sowie das jeweilige Nutzungsszenario und die individuellen Präferenzen des Benutzers berücksichtigen.

1.1.2 Suchmaschinen und Filterstrategien

Das Auffinden von zu den Interessen eines Benutzers passenden Informationen erfordert es, Methoden verschiedener Forschungsbereiche zu verknüpfen. Der Bereich des Information Retrievals (IR) beschäftigt sich mit Methoden zum Ermitteln von relevanten Informationen in großen Datensammlungen. Es existiert dazu eine große Anzahl von Verfahren, die jeweils spezifische Stärken und Schwächen aufweisen. Die Leistungsfähigkeit eines Verfahrens ist im Allgemeinen abhängig vom Suchszenario und der verwendeten Anfrage. Die Schwachstelle der meisten Filterverfahren besteht in der Beschränkung auf ein einzelnes Merkmal (z.B. Übereinstimmung mit Anfragebegriffen) sowie der mangelnden Berücksichtigung des Suchkontextes und der individuellen Benutzerinteressen. Dies dient in der folgenden Arbeit als Motivation, um verschiedene Filterstrategien miteinander zu kombinieren. Es werden Methoden entwickelt, mit denen Informationen entsprechend dem individuellen Informationsbedürfnis des Benutzers unter Berücksichtigung verschiedener Bewertungskriterien zusammengestellt werden können. Dabei wird neben dem Auswählen und Kombinieren alternativer Filterstrategien auf die personalisierte Auswahl von Informationsquellen, das effiziente Berechnen von Metadaten sowie auf das Zusammenstellen optimaler Ergebnismengen eingegangen. Des Weiteren werden Verfahren erläutert, mit denen ein Informationssystem den Anwender bei der Optimierung von Informationsanfragen unterstützt.

1.1.3 Personalisierung

Ein wichtiges Kriterium zum Bewerten der Relevanz von Informationen ist, wie gut diese zu den individuellen Kenntnissen und Präferenzen des Benutzers passen. Die Bewertung und die Präsentation der Informationen muss auf den Benutzer abgestimmt, d. h. personalisiert werden. Die Personalisierung von Informationssystemen ist ein wichtiger Schritt, damit der Anwender komplexe Systeme einfacher und effektiver nutzen kann. Im Bereich der Informationssysteme kommen zum Personalisieren Verfahren zum Einsatz, die beim Filtern und Darstellen von Informationen die unterschiedlichen Erfahrungen, Fähigkeiten, Anforderungen und Vorlieben der Benutzer berücksichtigen und das System entsprechend anpassen.

Bereiche, die bei der Personalisierung eines Systems in Betracht kommen, sind z. B. der Funktionsumfang, eine Optimierung der Anwendung auf die vom Benutzer eingesetzten Geräte, die Unterstützung durch Assistenten sowie die Verwendung von auf die Wünsche des Benutzers abgestimmten Gütefunktionalen. Im Rahmen der personalisierten Informationssuche müssen die Kriterien zur Auswahl von Informationsquellen sowie die Kriterien zur Bewertung von Informationen an die individuellen Erfahrungen des Benutzers angepasst werden. Darüber hinaus sollten dem Benutzer möglichst viele Routineaufgaben abgenommen werden und die Informationen so präsentiert werden, dass diese intuitiv und effektiv genutzt werden können.

1.1.4 Koordinationsstrategien

Um bei der Filterung unterschiedliche Kriterien und Algorithmen miteinander zu kombinieren, müssen geeignete Koordinationsstrategien eingesetzt werden. Man verfolgt dabei das Ziel, verschiedene Algorithmen so zu verknüpfen, dass unter Berücksichtigung der verfügbaren Ressourcen optimal auf die Interessen des Benutzers abgestimmte Informationen ermittelt werden. Bei der Auswahl und Koordination der einzusetzenden Methoden müssen sowohl benutzerspezifische Präferenzen (wie z. B. bevorzugte Relevanzkriterien), anfragespezifischen Anforderungen (z. B. Antwortzeit, Menge der erwarteten Dokumente) als auch Abhängigkeiten zwischen den Methoden berücksichtigt werden.

Bei der Definition von geeigneten Koordinations- und Kombinationsstrategien sollte beachtet werden, dass ein Filterframework eine große Anzahl von Filteranfragen unterstützen muss („Skalierbarkeit“), offen für neue Algorithmen sein sollte („Erweiterbarkeit“, „Offenheit“), Fehlerfälle geeignet behandelt („Fähigkeit zur Selbstheilung“), die Bewertungen des Benutzers einbezieht („Lernfähigkeit“) sowie bei der Auswahl der eingesetzten Verfahren die Verfügbarkeit von Ressourcen berücksichtigt. Für die Bearbeitung von Filteranfragen müssen die am besten geeigneten Methoden ausgewählt und ggf. miteinander kombiniert werden. Dabei sollte trotz der komplexen Adaptions- und Koordinationsstrategien die Nachvollziehbarkeit der Arbeitsweise des Systems für den Benutzer sichergestellt werden.

1.2 Forschungsgebiete

Die Forschungsschwerpunkte der folgenden Arbeit liegen im Bereich der flexiblen Kombination von Filter- und Bewertungsverfahren. Es wird untersucht, wie Filterstrategien miteinander kombiniert werden können, um ver-

lässlich hochqualitative, auf die Präferenzen des Benutzers abgestimmte Dokumente bereitzustellen. Es werden dazu existierende Filter- und Benutzermodellierungssysteme analysiert und die dabei eingesetzten Verfahren erläutert. Anhand der diskutierten Systeme und deren Schwachstellen wird eine Architektur für ein personalisiertes agentenbasiertes Informationssystem entwickelt. Im Rahmen der Systemarchitektur werden vielfältige Algorithmen zum Bewerten von Informationen unter Berücksichtigung der Präferenzen des Benutzers implementiert. Die Informationen werden dem Benutzer personalisiert, unter Beachtung des Suchkontextes, zur Verfügung gestellt. Es werden Algorithmen zum Kombinieren verschiedener Bewertungskriterien entwickelt und deren Eigenschaften bezüglich der Ergebnisqualität, des Bedarfes an Ressourcen sowie der Verlässlichkeit der Ergebnisse analysiert. Besondere Berücksichtigung findet dabei die Integration neuer Relevanzbewertungsstrategien sowie das Zusammenstellen von Filteragentengremien auf der Basis von Benutzerfeedback.

1.2.1 Gewählter Ansatz

Aufbauend auf der Analyse existierender Informationssysteme wird eine Architektur zum Bereitstellen personalisierter Filterergebnisse motiviert, implementiert und evaluiert. Besondere Berücksichtigung finden Verfahren zur Integration heterogener Quellen, Methoden zur multidimensionalen Bewertung von Dokumenten sowie Algorithmen zum Zusammenstellen von auf die Interessen des Benutzers optimierten Ergebnismengen. Von Forschungsinteresse ist dabei das Zusammenstellen von geeigneten Bewertungsstrategien, die eine optimale Verknüpfung heterogener Filterverfahren erlauben. Die Verfahren werden anhand von Benchmarkdatensätzen evaluiert und mit State-Of-The-Art-Algorithmen verglichen. Außerdem werden Erfahrungen mit den auf Basis der entwickelten Architektur realisierten Systemen zur personalisierten Bereitstellung von Informationen vorgestellt und die Vorteile der Architektur gegenüber heutigen Informationssystemen aufgezeigt.

1.2.2 Entwickelte Verfahren

Die im Rahmen dieser Arbeit durchgeführten Analysen und die entwickelten Algorithmen konzentrieren sich auf Methoden für die Realisierung offener, intelligenter, personalisierter Informationssysteme. Schwerpunkte bilden dabei die folgenden Problembereiche:

- **Systemarchitektur:** Es wird die Architektur für ein adaptives personalisiertes Informationssystem entwickelt. Es werden Verfahren aufge-

zeigt, durch die heterogene Datenquellen integriert, Dokumente einheitlich beurteilt und optimierte Ergebnismengen erstellt werden können.

- **Anfrageoptimierung:** Es werden Verfahren zur Optimierung von Suchanfragen entwickelt.
- **Bewertungsalgorithmen:** Es werden Algorithmen zur Bewertung von Dokumenten entwickelt, die verschiedene Kriterien („inhaltsbasiert“, „kollaborativ“, „semantisch“) berücksichtigen.
- **Kombination von Algorithmen:** Es werden verschiedene Algorithmen zum Erstellen von Gremien aus Bewertungsstrategien entwickelt. Dabei finden der jeweils erforderliche Aufwand und unterschiedliche Gütefunktionale besondere Berücksichtigung.
- **Adaptive Gremien:** Es werden Verfahren entwickelt, mit denen Gremien aus Bewertungsstrategien an sich verändernde Bedingungen (z. B. Benutzerinteressen, neue Bewertungsstrategien) angepasst werden können.

Die Evaluation auf der Basis von Benchmarkdatensätzen und im Rahmen von realisierten Informationssystemen zeigt die Leistungsfähigkeit der entwickelten Verfahren.

1.3 Aufbau der Arbeit

Die folgende Arbeit gliedert sich wie folgt: Im Kapitel 2 werden anhand von für die Informationssuche typischen Szenarien die Anforderungen an ein personalisiertes Informationssystem identifiziert und die Probleme klassischer Informationssysteme diskutiert. Kapitel 3 gibt einen Überblick über derzeitige Informationssysteme und analysiert die zur Anwendung kommenden Verfahren. Darüber hinaus werden Methoden zum Personalisieren von Anwendungen vorgestellt und Frameworks zur Realisierung verteilter Systeme erläutert. Im Kapitel 4 wird die Architektur des entwickelten Informationssystems vorgestellt. Es werden die Komponenten des Systems und die Verknüpfungen erläutert. Die anschließenden drei Kapitel erklären die Elemente der entwickelten Architektur im Detail. Das Ermitteln von relevanten Dokumenten sowie das Berechnen von Bewertungen und Metadaten wird im Kapitel 5 erläutert. Algorithmen zum Kombinieren von Bewertungen und zum Zusammenstellen von Ergebnismengen werden im Kapitel 6 vorgestellt. Die Weiterentwicklung und die Anpassung dieser Verfahren an die Anforderungen eines offenen personalisierten Informationssystems werden im Kapitel 7 erläutert

und evaluiert. Kapitel 8 stellt die auf der Basis der entwickelten Architektur realisierten Informationssysteme vor. Es wird unter Verwendung von in Benutzerbefragungen erhobenen Bewertungen evaluiert, wie leistungsfähig die Architektur und die entwickelten Verfahren in den realisierten Informationssystemen sind. Eine Zusammenfassung sowie einen Ausblick zu weiteren Forschungsarbeiten sind im Kapitel 9 zu finden.

Kapitel 2

Problembeschreibung

In vielen Bereichen ist das effiziente Finden von relevanten Informationen von großer Bedeutung. Die bei der Informationssuche zu berücksichtigenden Anforderungen sind sehr vielfältig und hängen von der Nutzergruppe, der Art der benötigten Informationen und dem Anwendungsbereich („Domäne“) ab. Um die Anforderungen an intelligente adaptive Informationssysteme zu motivieren, werden zunächst Szenarien beschrieben, die für die Informationssuche typische Anwendungsfälle illustrieren.

2.1 Szenarien

Die gewählten Szenarien werden anhand von Nutzergruppen strukturiert. Es werden die spezifischen Anforderungen der Nutzer für den jeweiligen Anwendungsbereich analysiert.

2.1.1 Wissenschaftler

Wissenschaftler nutzen zum Suchen von Dokumenten die Webseiten von Konferenzen, Berufsverbänden und Fachverlagen sowie themenbereichsspezifischen Portalen. Für die Beurteilung eines Dokumentes spielen, neben der inhaltlichen Nähe zu einer Fragestellung, weitere Kriterien und Metadaten eine wichtige Rolle, wie z. B. das Ansehen des Autors, der Verlag oder die Aktualität. Wissenschaftliche Dokumente zu aktuellen Forschungsergebnissen liegen dabei meist in englischer Sprache in für den Ausdruck optimierten Dateiformaten (z. B. Postscript oder PDF) vor.

Einarbeitung in einen neues Themengebiet

Möchte sich ein Wissenschaftler in ein neues Themengebiet einarbeiten, so besteht eine Strategie darin, die häufig zitierten (Basis-)Dokumente zu diesem Bereich zu bestimmen. Ausgehend von diesen Dokumenten ist es oft möglich, wichtige Persönlichkeiten und Konferenzen zu einem Forschungsbereich zu identifizieren und weitergehende Informationen zu finden. Ein System, das einen Wissenschaftler bei der Einarbeitung in einen neuen Themenbereich unterstützt, sollte somit Ergebnisse liefern, die die folgenden Anforderungen erfüllen:

- Die Ergebnismenge enthält die grundlegenden, oft zitierten Publikationen für das Themengebiet, wobei die Aktualität der Dokumente nur von begrenzter Wichtigkeit ist.
- Die Ergebnismenge sollte Dokumente von unterschiedlichen Forschergruppen einbeziehen und damit sicherstellen, dass mehrere Meinungen und Forschungsrichtungen berücksichtigt werden.
- Als Ergebnis auf eine Filteranfrage sollten neben wissenschaftlichen Texten auch Informationen zu weiteren Informationsquellen, wie z. B. Konferenzen ermittelt werden.

Als Ausgangspunkte für eine auf die Anforderungen von Wissenschaftlern abgestimmte Informationssuche können sowohl Dokumente aus einem Themenbereich als auch Schlagwörter und Namen von Forschern aus dem jeweiligen Interessengebiet dienen.

Verfolgen eines spezifischen Themengebietes

Um Wissenschaftler beim Verfolgen von neuen Entwicklungen in einem Themengebiet zu unterstützen, sollte ein Informationssystem den Benutzer über aktuelle Konferenzbeiträge und relevante Publikationen auf dem Laufenden halten. Da sich Wissenschaftler oft innerhalb ihres Bereiches spezialisieren, sind für ein solches Informationssystem Funktionen wichtig, die beim Finden und Bewerten von Dokumenten die Veränderungen des Interessenprofils erkennen und bei der Ermittlung relevanter Dokumente berücksichtigen. Kriterien für die Bewertung von Dokumenten in diesem Szenario sind beispielsweise die Aktualität der Dokumente, die inhaltliche Nähe zu den Interessengebieten des Benutzers und die Zielgruppe des Dokumentes (d. h. richtet sich das Dokument an Wissenschaftler oder handelt sich beim Dokument um eine vereinfachte „populärwissenschaftliche“ Darstellung für einen größeren Personenkreis).

Die Menge der vorgeschlagenen Dokumente sollte sich am Arbeitsrhythmus des Wissenschaftlers orientieren. Es sollten nicht zu viele Dokumente empfohlen werden, so dass wichtige Informationen nicht in einer zu großen Menge von Dokumenten verloren gehen. Außerdem sollte das System auf Anfrage interessante Veröffentlichungen zur Verfügung stellen, wenn der Forscher (z. B. auf einer längeren Reise) außerhalb der normalen Zeitplanung die Möglichkeit zum Lesen interessanter Dokumente hat.

2.1.2 Lernende

Für Lernende (z. B. Schüler und Studenten) spielt das Finden relevanter Dokumente eine wichtige Rolle, z. B. für die Vorbereitung von Vorträgen oder als Ergänzung zu den jeweiligen Lehrinhalten. Die Dokumente müssen dabei auf den jeweiligen Wissensstand abgestimmt sein und sollten den Lernenden nicht überfordern. Junge Benutzer sind zudem oft aufgeschlossen gegenüber sozialen Netzwerken und kollaborativen Bewertungsverfahren.

Ein Schüler erstellt einen Vortrag

Zur Vorbereitung von Vorträgen benötigt ein Schüler Dokumente, die auf seinen jeweiligen Wissensstand abgestimmt sind und sowohl eine Einführung in den Themenbereich geben als auch Texte, die weitergehende Informationen bereitstellen. Ein für dieses Szenario angepasstes Informationssystem sollte die folgenden Anforderungen erfüllen:

- Um Basiswissen für die im jeweiligen Themengebiet wichtigen Begriffe bereitzustellen, sollten Nachschlagewerke (z. B. Lexika) beim Zusammenstellen der Ergebnismenge berücksichtigt werden.

- Da zu vielen aktuellen Themenbereichen bereits Ausarbeitungen von anderen Schülern existieren, sollten bei der Informationssuche Portale berücksichtigt werden, die (auf die Zielgruppe abgestimmte) Vortragsausarbeitungen bereitstellen.

- Eine wichtige Informationsquelle stellen zudem Lehrbücher sowie aktuelle Fachliteratur dar. Das Filtersystem sollte deshalb auch relevante Bücher (z. B. Online-Buchläden, Bibliothekskataloge) in die Suche einbeziehen.

- Wissenschaftliche Konferenzbeiträge, die die wissenschaftliche Gemeinschaft als Zielgruppe haben, sind für die Ausarbeitung von Vorträgen nur bedingt geeignet. Da sich diese Publikationen oft auf ausgewählte Details beschränken und keine Einführung in einen Themenbereich

geben, sollten diese Dokumentenquellen in diesem Szenario nicht einbezogen werden.

- Schüler haben oft Probleme beim Lesen von Fachtexten in einer Fremdsprache. Das Informationssystem sollte beim Bewerten der Relevanz von Dokumenten die Sprachkenntnisse des Benutzers berücksichtigen.

Bei dem Zusammenstellen der Ergebnismenge sollten die Dokumente der unterschiedlichen relevanten Ergebniskategorien (z.B. Nachschlagewerke, Ausarbeitungen anderer Lernender, Bücher, weiterführende Informationen) in einem angemessenen Verhältnis stehen.

Technikbegeisterte Benutzer

Technikbegeisterte Benutzer geben sich oft nicht mit den bereits verfügbaren Funktionen eines Systems zufrieden, sondern wollen die Arbeitsweise des Systems verstehen, optimieren und ggf. auch erweitern. Personen dieser Gruppe sind daher möglicherweise daran interessiert, bestehende Filter- und Bewertungsverfahren zu verbessern und neue innovative Algorithmen zu integrieren. Ein Filtersystem sollte daher offen für Erweiterungen sein; dabei aber sicherstellen, dass neu hinzukommende Filterstrategien die Leistungsfähigkeit des Informationssystems nicht beeinträchtigen. Die Möglichkeit, eigene Erweiterungen für ein System zu realisieren, kann bei technikbegeisterten Benutzern die Akzeptanz des Systems steigern und helfen, neue Algorithmen so in das System zu integrieren, dass davon alle Nutzer profitieren. Um die Nachvollziehbarkeit der eingesetzten Algorithmen und der berechneten Ergebnisse sicherzustellen, sollten die Filterverfahren jeweils eine Erklärung bereitstellen, anhand welcher Kriterien eine Bewertung ermittelt worden ist. Darüber hinaus sollte interessierten Benutzern die Möglichkeit gegeben werden, Filterkomponenten explizit auf ihre individuellen Präferenzen zu trainieren.

Student schreibt Diplomarbeit

Beim Anfertigen einer Diplomarbeit muss ein Student den Stand der Forschung beschreiben und dabei aktuelle wissenschaftliche Arbeiten einbeziehen. Zusätzlich zu den Kriterien, die bereits im Szenario „Einarbeitung in ein neues Themengebiet“ diskutiert wurden, muss berücksichtigt werden, dass einige Forschungsberichte über verschiedene Quellen bezogen werden können und dass Journal- und Konferenzbeiträge teilweise nur gegen eine Gebühr verfügbar sind. Um die finanzielle Belastung in einem akzeptablen Rahmen zu halten, sollte das Filtersystem Empfehlungen ableiten können, ob es lohnt,

kostenpflichtige Angebote bei der Zusammenstellung der Ergebnismenge zu berücksichtigen, und ggf. auf Alternativen (z. B. Bibliotheken) hinweisen.

Da bei Diplomarbeiten der Betreuer häufig als Ausgangspunkt für eine Literaturrecherche grundlegende wissenschaftliche Dokumente zur Verfügung stellt, ist bei der Suche nach relevanten Dokumenten für eine Diplomarbeit ein Schlüsselwortassistent hilfreich. Dieser kann aus den gegebenen Dokumenten relevante Begriffe und Konzepte ableiteten und so den Benutzer aktiv bei der Recherche unterstützen.

2.1.3 Geschäftsleute

Für Geschäftsleute sind aktuelle Informationen wichtig, um die richtigen Entscheidungen zu treffen. Da Geschäftleute oft unterwegs sind, werden Informationssysteme per PDA genutzt, so das bereitgestellte Dokumente für die mobile Nutzung („begrenzte Übertragungskapazität“, „Verwendung spezieller Dateiformate“, „geringe Bildschirmgröße“) optimiert werden müssen.

Ein interessantes Anwendungsszenario ist dabei die elektronische Zeitung, welche die Eigenschaften einer klassischen Zeitung (wie z. B. allgemein anerkannte Struktur, einfache Handhabung) mit den Vorteilen innovativer Filter- und Personalisierungssysteme (z. B. höhere Aktualität, bessere Verfügbarkeit im Ausland, einfacheres Suchen und Verwalten von Artikeln, Anpassung an die individuellen Interessen des Nutzers, geringerer Platzbedarf) verbindet.

Bei der Erstellung von personalisierten Zeitungen muss, in Abhängigkeit vom Wochentag ein ausgewogenes Verhältnis zwischen berufsrelevanten Nachrichten (z. B. Patentanträge, Forschungsergebnisse, Börsennachrichten), Informationen zur Freizeitgestaltung und den persönlichen Hobbys sichergestellt werden. Zusätzlich zu einem Online-Zugriff sollte ein Informationssystem die Zustellung der personalisierten Zeitung per Push-Service (z. B. E-Mail, FAX) anbieten.

2.1.4 Senioren

Menschen, die sich nur wenig für neue Technik interessieren und Schwierigkeiten im Umgang mit dem Computer haben, legen bei der Nutzung eines Informationssystems viel Wert auf eine möglichst einfache und unkomplizierte Bedienung. Für die Informationssuche ergeben sich daraus die folgenden Anforderungen:

- Ein an diese Benutzergruppe angepasstes Informationssystem sollte verschiedene Informationsquellen berücksichtigen, so dass der Nutzer

nur die Funktionen eines Systems „erlernen“ muss und trotzdem alle für ihn relevanten Informationen findet.

- Als Informationsquelle sollten neben Suchmaschinen und Portalen auch Datenquellen zum Fernsehprogramm (z.B. die Daten des elektronischen Programmführers – EPG) integriert werden, da das Fernsehen eine wichtige Informationsquelle für diese Personengruppe ist.
- Das Filtersystem sollte keine Dokumente vorschlagen, die in einem Dateiformat vorliegen, die dem Benutzer unbekannt sind und aufgrund fehlender Software nicht geöffnet werden können. Ggf. sollte das Informationssystem eine geeignete Konvertierung vornehmen.
- Um den Benutzer beim Formulieren von Suchanfragen zu unterstützen, sollte das Informationssystem den Anwender durch einen Assistenten bei der Optimierung von Suchanfragen unterstützen.

Senioren haben spezielle Anforderungen an ein Informationssystem, denen durch den Einsatz von auf diese Benutzergruppe optimierten Strategien zum Bewerten und Zusammenstellen von Dokumenten Rechnung getragen werden muss.

2.2 Anforderungen an personalisierte Informationssysteme

Anhand der im Abschnitt 2.1 vorgestellten Szenarien lassen sich die folgenden Anforderungen an ein adaptives personalisiertes Informationssystem ableiten.

Personalisierung Um dem Benutzer hochqualitative, auf die individuellen Präferenzen abgestimmte Dokumente bereitstellen zu können, muss das Informationssystem in der Lage sein, die Interessen des Benutzers zu lernen und die Ergebnisse darauf anzupassen. Bei der Bewertung von Dokumenten muss daher jeweils eine geeignete Gewichtung verschiedener Gütekriterien (z.B. Exaktheit, Verlässlichkeit, Geschwindigkeit, Größe der Ergebnismenge) gewählt werden können.

Offenheit und Erweiterbarkeit Das Informationssystem muss eine große Anzahl verschiedener Informationsquellen, Filterverfahren, Bewertungsalgorithmen und Strategien zum Zusammenstellen von Ergebnismengen integrieren. Das System muss offen für neu hinzukommende Quellen und Filter-

verfahren sein („Erweiterbarkeit“) und eine große Anzahl von Anfragen und Filterverfahren handhaben können („Skalierbarkeit“).

Kontextbasierte Auswahl von Strategien Das Informationssystem sollte die kontext- und benutzersensitive Auswahl von Informationsquellen, Filterverfahren und Algorithmen zur Informationsaufbereitung ermöglichen. Darüber hinaus müssen neben den Wünschen des Benutzers die Anforderungen des Systembetreibers berücksichtigt werden. Wichtige Kriterien für die Administration sind die einfache Integration und Anpassung von Systemkomponenten sowie die Auswahl von Verfahren unter Berücksichtigung der jeweiligen Systemauslastung („Load-Balancing“).

Berücksichtigen vielfältiger Informationsquellen Ein Informationssystem sollte vielfältige Informationsquellen einbeziehen und in Abhängigkeit vom Szenario und der Anfrage die jeweils am besten geeigneten Quellen auswählen.

Auswahl von Informationsquellen und Anpassung der Anfragen: Um dem Benutzer jeweils hochwertige Ergebnisse zur Verfügung stellen zu können, muss das System die für eine Anfrage relevanten Informationsquellen auswählen und die Anfrage für die spezifischen Merkmale der Quelle anpassen.

Erstellen und Optimieren von Anfragen Zum Erreichen von auf die individuellen Interessen des Benutzers abgestimmten Filterergebnissen, sollte das Informationssystem den Benutzer bei der Erstellung und Optimierung von Anfragen unterstützen.

Bereitstellen von umfassenden Metadaten und Relevanzbewertungen Um die Relevanz von Dokumenten richtig einschätzen zu können, sollten mehrere Kriterien für die Beurteilung von Dokumenten herangezogen werden. Darüber hinaus sollten die Metadaten bereitgestellt werden, die dem Benutzer die Nutzung der Dokumente erleichtern und die für eine optimale Präsentation der Informationen benötigt werden (z. B. zur Präsentation einer Dokumentenmenge als thematisch strukturierter Baum).

Kombination von Bewertungsverfahren Für die Bewertung von Dokumenten sollten verschiedene (konkurrierende und orthogonale) Bewertungskriterien (abgestimmt auf die individuellen Präferenzen des Benutzers) miteinander kombiniert werden.

Nachvollziehbarkeit der Ergebnisse Das System sollte dem Benutzer erklären können, anhand welcher Kriterien eine Dokumentenbewertung ermittelt worden ist. Dies unterstützt den Benutzer bei der Beurteilung von Dokumenten und ermöglicht es, einzelne Bewertungsstrategien zu bewerten.

Unterstützung bei Routineaufgaben Das System sollte den Benutzer bei Routineaufgaben assistieren, z. B. durch die Prüfung der Verfügbarkeit von Dokumenten oder durch das Bewertung der Zuverlässigkeit von Informationsquellen. Darüber hinaus sollte das Informationssystem das regelmäßige automatische Suchen nach Dokumenten anbieten, um dem Benutzer das Verfolgen von aktuellen Entwicklungen in einem Bereich zu erleichtern.

Berücksichtigen der Verfügbarkeit von Ressourcen Um schnell und effizient Anfragen bearbeiten zu können, sollte das System Verfahren zur Lastverteilung und Adaption an die jeweils verfügbaren Ressourcen einsetzen.

Die Liste der Anforderungen zeigt, dass an moderne Informationssysteme vielfältige Anforderungen gestellt werden, so dass eine Architektur gewählt werden muss, die eine flexible Kombination von spezialisierten Komponenten ermöglicht.

2.3 Probleme klassischer Informationssysteme

Traditionelle Suchmaschinen bewerten die Relevanz von Dokumenten im Allgemeinen anhand eines statisch definierten Bewertungsalgorithmus. Dabei kommen überwiegend *inhaltsbasierte* Filterstrategien zum Einsatz („content based filtering“), die die Ähnlichkeit eines Dokumentes zu den in einer Anfrage definierten Suchbegriffen berechnen. Kommen inhaltsbasierte Filterstrategien zum Einsatz, müssen die folgenden Probleme berücksichtigt werden:

- Bei der Verwendung von inhaltsbasierten Filterverfahren hängt die Qualität der Suchergebnisse stark von den gewählten Suchbegriffen ab. Unerfahrene Benutzer finden aufgrund von ungeeigneten Begriffen oftmals nicht die gewünschten Dokumente [Wol05].

- Bei der inhaltsbasierten Suche werden zur Komplexitätsreduktion im Allgemeinen verschiedene Vorverarbeitungsschritte durchgeführt (z. B. Reduktion auf den Wortstamm, Entfernen von Stoppwörtern), für die die Sprache des Textes bekannt sein muss.

- Aus sehr vielen Begriffen bestehende Suchanfragen (wie sie z. B. bei der Expansion von Wildcards entstehen) sind im Allgemeinen mit einem hohen Filteraufwand verbunden.

- Rechtschreibfehler in Dokumenten und Suchbegriffen beeinträchtigen die Güte der Filterergebnisse.

- Dokumente, die Synonyme zu einem Suchbegriff, nicht aber den Suchbegriff selbst enthalten, werden nicht gefunden.

- Das Bewerten der Relevanz eines Dokumentes auf Basis von (statistischen) Term-Häufigkeiten ist für unerfahrene Benutzer nur schwer nachvollziehbar.

- Der Autor eines Dokumentes kann durch das Einfügen von nicht sichtbaren Texten die Relevanzbewertung des Dokumentes manipulieren („Cloaking“, „Keyword-Spamming“).

Eine Alternative zu inhaltsbasierten Filterverfahren stellen die *kollaborative* Verfahren dar. Kollaborative Filterverfahren analysieren die von Nutzern abgegebenen Bewertungen und ermitteln anhand der Ähnlichkeit zwischen Benutzern und Dokumenten Relevanzbewertungen. Diese Algorithmen verfolgen dabei die Grundidee, dass sich untereinander ähnliche Benutzer (die beispielsweise Dokumente gleich bewertet haben) auch für dieselben Informationen interessieren.

Das Berechnen der Relevanz von Dokumenten erfolgt bei kollaborativen Filterverfahren in mehreren Schritten: Zunächst wird die Menge der Personen ermittelt, die dem Benutzer am ähnlichsten sind, für den relevante Dokumente ermittelt werden sollen (z. B. auf Basis von Bewertungen oder Interessenprofilen). Anschließend werden dem suchenden Benutzer die Dokumente empfohlen, die Benutzer mit ähnlichen Interessen positiv bewertet haben. Großen Einfluss auf die Güte der Ergebnisse haben die verwendeten Maße zum Berechnen der Ähnlichkeit zwischen Dokumenten und Benutzern. Bei der Verwendung kollaborativer Filterverfahren müssen die folgenden Problembereiche beachtet werden:

- **Cold Start Problem:** Eine Voraussetzung für die Einsetzbarkeit von kollaborativen Bewertungsverfahren ist, dass eine hinreichend große Menge von Bewertungen zur Verfügung steht. Daher sollten in der Startphase eines Systems, bei der im Allgemeinen nur wenige Bewertungen zur Verfügung stehen, alternative Filterverfahren (z. B. inhaltsbasierte Strategien) eingesetzt werden.

- **New User Problem:** Neue Benutzer müssen zunächst ein Profil im System hinterlegen oder eine hinreichend große Anzahl von Bewertungen abgeben, damit auf der Basis kollaborativen Verfahren zuverlässige Bewertungen ermittelt werden können. Dies stellt eine beträchtliche Einstiegshürde zur Nutzung eines kollaborativen Systems dar.

- **Sparsity:** Bei einer geringen Anzahl von Benutzern und Bewertungen lassen sich keine verlässlichen Aussagen über die Ähnlichkeit von Benutzern treffen, was oft zu einer reduzierten Ergebnisqualität führt.

- **Biased ratings, different rating meanings:** Benutzer vergeben Bewertungen unter Berücksichtigung verschiedener Kriterien, d. h. auch wenn zwei Benutzer identische Bewertungen abgeben haben, sind ihre Intensionen und Interessen möglicherweise verschieden. Dies kann insbesondere bei einer geringen Anzahl von Bewertungen zu fehlerhaften Ergebnissen führen. Um das Biased-Rating-Problem zu reduzieren, müssen Benutzerbewertungen jeweils geeignet normalisiert werden.

- **Popularity Bias:** Kollaborative Filteralgorithmen liefern keine zuverlässigen Ergebnisse für Personen mit speziellen Interessen. Kollaborative Filter tendieren dazu, Dokumente vorzuschlagen, die von einer großen Anzahl von Benutzern bewertet wurden.

- **Synonymy problem:** Da beim kollaborativen Filtern die Dokumente im Allgemeinen als „Blackbox“ betrachtet werden, tritt das Problem auf, dass gleiche Dokumente (die unterschiedliche IDs besitzen, weil sie über unterschiedliche Anbieter zur Verfügung gestellt werden) nicht als Synonyme erkannt werden. Dies verringert die Aussagekraft der berechneten Ähnlichkeitsmaße und damit die Güte der ermittelten Bewertungen.

Kollaboratives Filtern stellt einen alternativen Ansatz zum inhaltsbasierten Filtern dar und kann gute Ergebnisse liefern, wenn bestimmte Randbedingungen erfüllt sind [BS97, ASST05]. Um zuverlässig hochqualitative Ergebnisse zu erhalten, sollten kollaborative Filterverfahren mit weiteren Filterverfahren (z. B. inhaltsbasierten Verfahren) verknüpft werden.

Kombination verschiedener Filterverfahren

Da die verschiedenen Filterstrategien jeweils spezifische Stärken und Schwächen aufweisen und die Benutzer individuelle Präferenzen beim Bewerten von Dokumenten haben, sollten verschiedene Filterstrategien kombiniert werden.

In Abhängigkeit vom jeweiligen Szenario und den persönlichen Wünschen des Benutzers müssen geeignete Filteralgorithmen, Parameterkonfigurationen und Strategien zum Kombinieren von Bewertungen gewählt werden.

Kriterien zur Bewertung der Filterqualität

Um die Güte von Filterergebnissen zu bewerten, können vielfältige Faktoren herangezogen werden, wie z. B. die Exaktheit der Ergebnismenge, die Vollständigkeit der Ergebnismenge, die Aktualität der ermittelten Dokumente, die Verlässlichkeit der Ergebnisse sowie die Übereinstimmung der Dokumente mit individuellen Benutzerpräferenzen. Für den Betreiber eines Systems ist darüber hinaus von Bedeutung, welche System-Ressourcen für das Ermitteln relevanter Dokumente benötigt werden. Da jeder Benutzer eigene Anforderungen an relevante Dokumente stellt, sollte ein Framework für die personalisierte Bereitstellung von Informationen in der Lage sein, verschiedene Gütekriterien zu berücksichtigen und in Abhängigkeit von den individuellen Präferenzen des Benutzers mehrere Kriterien zu kombinieren [CS07, AM01, SKR02].

2.4 Zusammenfassung

An Informationssysteme werden hohe Anforderungen bezüglich Personalisierung, Offenheit und Adaptivität gestellt. Die Analyse der für die Informationssuche relevanten Szenarien zeigt die Vielfältigkeit der zu berücksichtigenden Nebenbedingungen und Benutzerpräferenzen. Es müssen jeweils geeignete Datenquellen ausgewählt werden und die potentiell relevanten Dokumente unter Verwendung von auf die Präferenzen und Kenntnisse des Benutzers abgestimmten Gütefunktionalen bewertet werden, um anhand dieser Daten optimale Filterergebnismengen zusammenzustellen. Derzeit verfügbare Filtersysteme können diese Anforderungen nur teilweise erfüllen, da diese Systeme nur eine geringe Anzahl von Filterstrategien anwenden, wenige Erweitungsmöglichkeiten bieten und keine Möglichkeit zum Personalisieren von Relevanzbewertung bieten. Bei der Verwendung von nur einem Typ von Filterstrategie (z. B. nur inhaltsbasiert oder nur kollaborativ) können nicht für jede Anfrage gut passende Ergebnisse erreicht werden. Ein adaptives Informationssystem sollte jeweils unter Berücksichtigung der gestellten Anfrage, dem jeweiligen Szenario und den individuellen Benutzerpräferenzen leistungsfähige Filterstrategien auswählen und diese geeignet kombinieren.

Kapitel 3

Existierende Systeme und Verfahren

Im Bereich der Informationssysteme sowie der Personalisierung von komplexen Anwendungen existiert eine große Anzahl interessanter Systeme und Algorithmen. Neben den populären Suchmaschinen und Verzeichnisdiensten existieren zahlreiche Forschungsprototypen, die neue Ansätze zur Realisierung personalisierter Informationssysteme einsetzen. Um einen Einblick in den aktuellen Stand der Forschung und die eingesetzten Technologien zu geben, werden in diesem Abschnitt ausgewählte Informationssysteme beschrieben und diskutiert.

3.1 Such- und Metasuchmaschinen

Die bekanntesten Systeme zur Dokumentensuche stellen Such- und Metasuchmaschinen dar. Suchmaschinen erstellen im Allgemeinen zu einer gegebene Dokumentenmenge (z. B. Texten, Bilder) einen Index in dem unter Verwendung einer schlüsselwortbasierten Anfrage nach passenden Dokumenten gesucht werden kann. Metasuchmaschinen leiten Anfragen an mehrere Suchmaschinen weiter und fassen die ermittelten Ergebnismengen zusammen. Beispielhaft werden im Folgenden eine Such- und eine Metasuchmaschine vorgestellt.

3.1.1 Google

Google[1] ist die weltweit am häufigsten verwendete Suchmaschine (Marktanteil in Deutschland ca. 89%, weltweit ca. 54% [web08, Six06]). Google

[1] `http://www.google.com/`

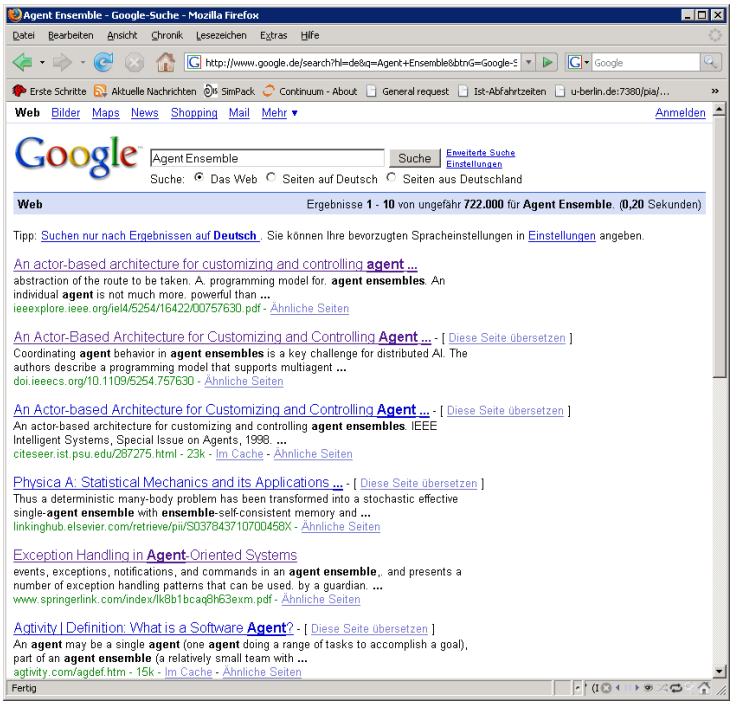


Abbildung 3.1: Die Abbildung zeigt das durch die Suchmaschine GOOGLE ermittelte Ergebnis für die Anfrage `Agent Ensemble`. Jedes Dokument der Ergebnisliste wird durch eine Überschrift, einen zur Anfrage passenden Dokumentenausschnitt und eine Quellenangabe repräsentiert. [10.03.2008]

indiziert Internet-Objekte (z. B. Texte, Bilder, Videos, Produkte) und berechnet die Relevanz der indizierten Objekte zu einer Anfrage. Die Relevanz eines Dokumentes wird vorrangig anhand der Häufigkeit des Auftretens von Suchbegriffen und auf Basis der Analyse von Referenzen auf das Dokument berechnet [ACD+07, Goo07]. Die Suchmaschine GOOGLE verwendet eine sehr einfach gestaltete Internetseite [Sul98], die Suchergebnisse in Form einer sortierten List darstellt (Abbildung 3.1).

3.1.2 MetaGer

METAGER[2] ist eine von der Universität Hannover betriebene Metasuchmaschine, die eine Suchanfrage an über 30 Suchmaschinen weiterleiten kann. Die Metasuchmaschine bietet dem Benutzer die Möglichkeit, die gewünschten Suchmaschinen auszuwählen sowie Kriterien zum Zusammenstellen von Ergebnismengen zu definieren. Die für eine Anfrage von verschiedenen Such-

[2] `http://www.metager.de`

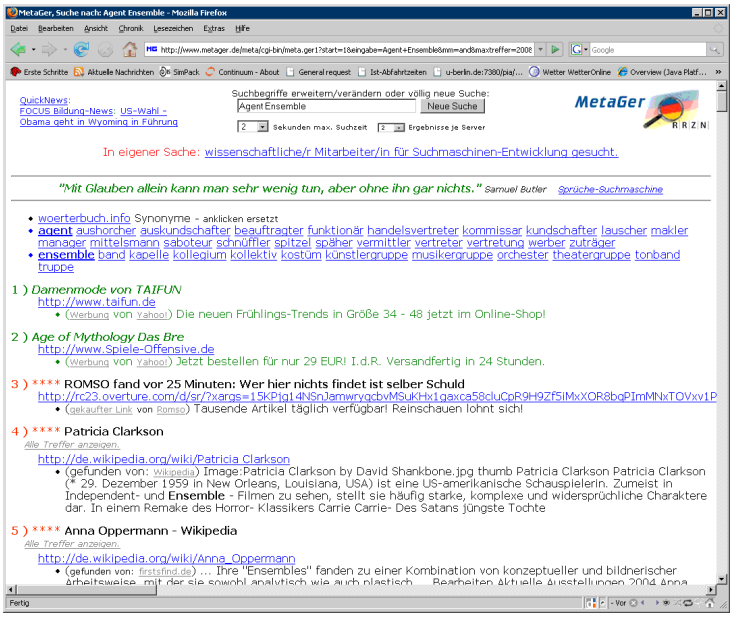


Abbildung 3.2: Die Abbildung zeigt die Ergebnisliste der Metasuchmaschine METAGER zur Anfrage `Agent Ensemble`. Jedes Dokument der Ergebnisliste besteht dabei aus einem Titel, einem „Abstract“ und einer Quellenangabe. Zusätzlich ist jeweils angegeben, welche Suchmaschinen das Dokument gefunden haben und wie METAGER die Güte des Dokumentes einschätzt. [10.03.2008]

maschinen abgefragten Dokumente werden auf Duplikate geprüft und in einer Liste zusammengefasst. Die von METAGER ermittelten Metadaten sowie die verwendete Ergebnissortierung ist für den Benutzer nur schwer nachvollziehbar (Abbildung 3.2). Anfragen bei METAGER sind im Allgemeinen langsam (ca. 3 Sekunden) und die Ergebnisliste ist aufgrund der großen Anzahl von Dokumenten oft schwer zu überblicken. Die Metasuchmaschine bietet vielfältige Einstellungsmöglichkeiten. Es existieren aber keine Funktionen zum Anpassung an Benutzerbewertungen oder implizites Feedback. Beim Ermitteln von Suchbegriffen unterstützt METAGER den Benutzer mit einem Assistenten (WEB-ASSOZIATOR[3]).

3.2 Themenbereichsfokussierte Suche

Da die Effektivität von Filteralgorithmen in Abhängigkeit von der jeweiligen Domäne variiert, kann durch die Beschränkung auf einen Themenbereich

[3] `http://www.metager.de/asso.html`

eine Verbesserung der Filterqualität erreicht werden. Themenbereichsfokussierte Suchmaschinen begrenzen im Allgemeinen die Anzahl der berücksichtigten Quellen oder klassifizieren Dokumente bezüglich ihrer Zugehörigkeit zu themenbereichsspezifischen Kategorien. Die eingesetzten Filter- und Bewertungsstrategien werden an die jeweilige Domäne und die Zielgruppe angepasst. Das Fokussieren einer Suchmaschine auf einen Bereich kann daher als eine spezielle Art der Personalisierung angesehen werden.

3.2.1 Verzeichnisdienste

Verzeichnisdienste definieren eine Hierarchie von Themenbereichen („Kategorien“) und ordnen Datenquellen und Informationsanbieter diesen Kategorien zu. Die Kategorien und Datenquellen können im Allgemeinen mit zusätzlichen Metadaten beschrieben werden. Die Zuordnung von Informationsquellen zu Themenbereichen wird durch die Nutzer eines Internetdienstes (z. B. DMoz[4]) oder durch Redakteure (z. B. Yahoo - Verzeichnis[5]) vorgenommen.

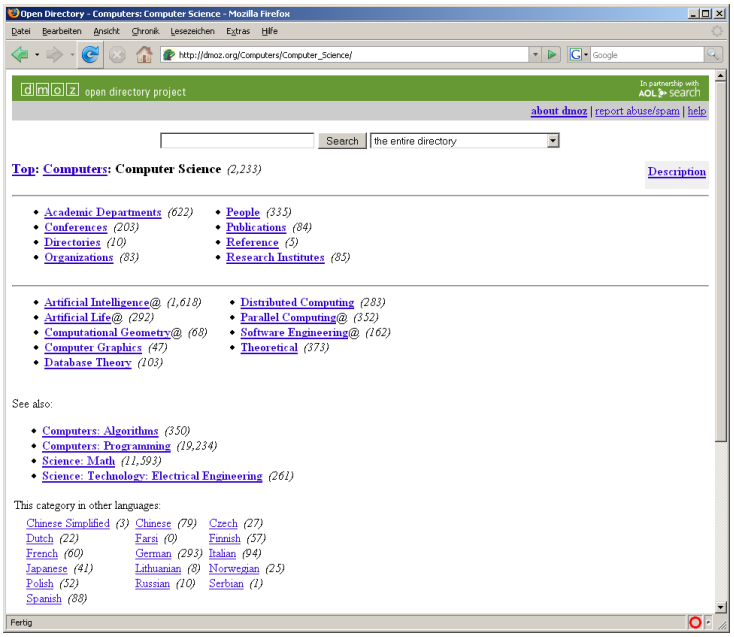


Abbildung 3.3: Die Abbildung zeigt die Kategorien, die im Verzeichnisdienst DMoz dem Themenbereich `Computer Science` zugeordnet sind. [10.03.2008]

Verzeichnisdienste haben (im Vergleich zu alternativen Ansätzen zur themenbereichsfokussierten Suche) mehrere Vorteile:

[4] `http://www.dmoz.org/`
[5] `http://www.yahoo.com/`

- **Verlässliche Zuordnung:** Da die Zuordnung von Informationsanbietern zu Kategorien durch Menschen vorgenommen wird, können fehlerhafte Bewertungen Informationsquellen, wie sie bei maschinellen Klassifikation (z. B. bei homographen Begriffen mit unterschiedlicher Bedeutung) auftreten, weitgehend vermieden werden. Die durch Redakteure ergänzten Metadaten (z. B. Seitenbeschreibungen) können den Benutzern helfen, die Relevanz einer Internetseite bezüglich ihrer Interessen einzuschätzen.

- **Erkennung von Spam:** Da die Relevanz der Informationsangebote durch Menschen geprüft wird, kann Spam effektiv erkannt und ausgeschlossen werden.

- **Hierarchische Struktur:** Die hierarchische Struktur von Kategorien ermöglicht eine präzise Zuordnung von Informationsquellen zu einem Themenbereich.

Die Nachteile von Verzeichnisdiensten sind:

- **Redaktioneller Aufwand:** Um die korrekte Zuordnung von Informationsquellen zu einem Themenbereich sicherzustellen, muss durch einen Redakteur jedes Informationsangebot geprüft werden. Dies ist bei umfangreichen Verzeichnissen mit einem erheblichen Aufwand verbunden. Neben der Einordnung neuer Informationsangebote müssen zudem die bestehenden Einträge regelmäßig überprüft werden, um auf Veränderung der registrierten Quellen zu reagieren. Bei umfangreichen Verzeichnisdiensten (DMOZ speichert die Adressen von ca. 4,8 Millionen Internetseiten) besteht das Problem von veralteten und fehlerhaften Einträgen.

- **Komplexe Navigation:** Bei großen Verzeichnissen werden die Themen-Hierarchien unübersichtlich. Für den Benutzer ist es daher oft nicht einfach, die zu den individuellen Interessen passende Kategorie im Verzeichnis zu finden.

- **Unzureichende Personalisierung:** Die Definition von Themenbereichen erfolgt bei den Verzeichnisdiensten nach allgemeingültigen Kriterien. Benutzerspezifische Kriterien für die Definition von Themenbereichen werden nicht unterstützt.

Abbildung 3.3 zeigt einen Ausschnitt aus dem Verzeichnisdienst DMOZ. Durch die große Anzahl von Unterkategorien zur gewählten Kategorie

`Computer Science` ist es für den Benutzer schwierig, die relevanten Informationen zu finden, da die Kategoriennamen nur begrenzte Aussagekraft besitzen.

3.2.2 Themenbereichsspezialisierte Suchmaschinen

Um die Relevanzberechnung und die unterstützten Suchoptionen auf einen thematisch begrenzten Bereich anzupassen, können Suchmaschinen auf ein Themengebiet eingeschränkt werden. Die auf einen Themenbereich spezialisierten Suchmaschinen orientieren sich in der Funktionsweise stark an allgemeinen Suchmaschinen, berücksichtigen aber nur eine beschränkte Menge von Datenquellen und bieten im Allgemeinen auf die jeweilige Domäne abgestimmte Kriterien zur Relevanzbestimmung an. Das Beschränken der berücksichtigen Datenquellen stellt sicher, dass keine Suchergebnisse aus nicht zum gewünschten Themenbereich passenden Domänen in der Filterergebnismenge auftreten und dass die Relevanzbewertung von Termen (z. B. auf der Basis von Tf – Idf-Statistiken) nicht durch themengebietsfremde Dokumente beeinträchtigt wird.

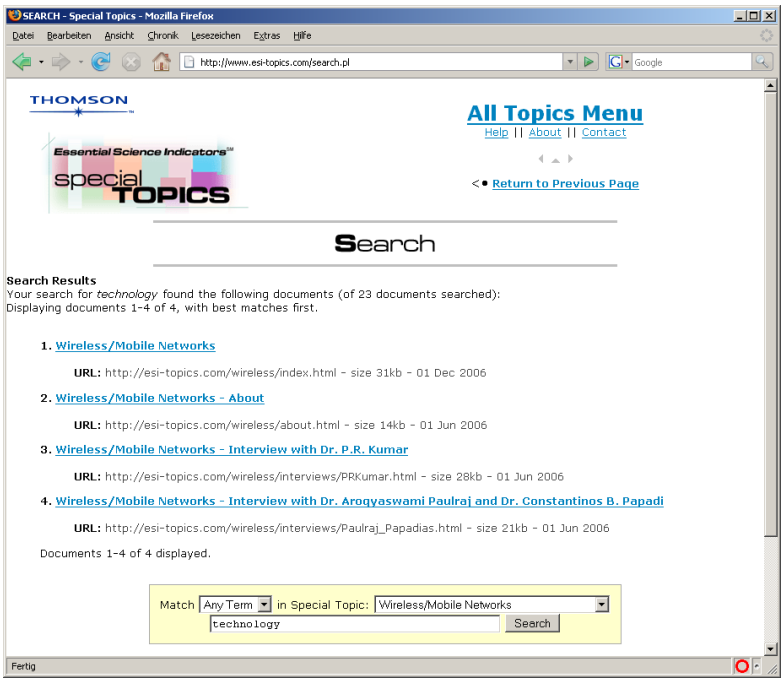


Abbildung 3.4: Die Abbildung zeigt eine auf den Themenbereich `Wireless/Mobile Networks` beschränkte Suche mit der Suchmaschine ISI®. [10.03.2008]

Die Nachteile dieses Ansatzes bestehen darin, dass durch die Beschränkung der berücksichtigen Datenquellen potentiell für den Benutzer relevante Dokumente, die zu einem Randbereich des Themenbereichs gehören, nicht gefunden werden können. Darüber hinaus ist das Erstellen von themenbereichsspezialisierten Portalen mit einem redaktionellen Aufwand verbunden, da die zu berücksichtigenden Datenquellen definiert und die Bewertungskriterien an den jeweiligen Themenbereich angepasst werden müssen. Eine umfangreiche Personalisierung auf die individuellen Benutzerpräferenzen ist oft nicht möglich. Allerdings erlauben einige Systeme (z. B. ISI®) dem Benutzer eigene Quellen zum Suchindex hinzuzufügen. Suchmaschinen, die eine domänenbeschränkte Suche anbieten, sind beispielsweise GIGABLAST[6] und ISI®[7]. Abbildung 3.4 zeigt eine domänenbeschränkte Suchanfrage mit der Suchmaschine ISI®. Der Benutzer muss zusätzlich zu den Suchbegriffen jeweils die Domäne definieren, in der gesucht werden soll.

3.2.3 Themenorientierte Ergebnispräsentation

Eine Alternative zum Beschränken von Suchmaschinen auf ausgewählte Themenbereiche besteht in der themenorientierten Präsentation der Filterergebnisse [FWRZ06]. Der Benutzer muss sich bei diesem Ansatz nicht von vornherein auf einen Themenbereich festlegen; stattdessen werden automatisch die Themenbereiche identifiziert, in denen die Suchanfrage relevante Ergebnisse liefert. Damit eignet sich dieser Ansatz besonders für Benutzer, die unscharfe Suchanfragen verwenden und einen Überblick erhalten möchten, in welchen Bereichen die Suchbegriffe auftreten. Die Strukturierung der Ergebnismenge hilft dem Benutzer zu bewerten, inwieweit ein Suchergebnis zu seinen Interessen passt.

Zur dynamischen Ermittlung von Themenbereichen werden Algorithmen zur Analyse der Verlinkungsstruktur und Methoden des Clusterings eingesetzt [Boi05]. Portale, die Suchergebnisse graphisch aufbereiten oder gruppiert darstellen, sind beispielsweise CLUSTY[8], UJIKO[9] und KARTOO[10] (Abbildung 3.5).

[6] http://www.gigablast.com/cts.html
[7] http://www.esi-topics.com/search.pl
[8] http://clusty.com/
[9] http://www.ujiko.com/
[10] http://www.kartoo.com/

Abbildung 3.5: Die Abbildung veranschaulicht die graphische Präsentation einer Menge von Filterergebnissen durch die Suchmaschine KartOO. [10.03.2008]

3.2.4 Erweitern von Suchanfragen

Unerfahrene Anwender haben bei der Verwendung von Suchmaschinen häufig Schwierigkeiten, die zur Beschreibung ihres Informationsbedürfnisses geeigneten Suchbegriffe zu finden. Einige Informationssysteme bieten daher Assistenten, die dem Benutzer zu einem gegebenen Begriff Wörter vorschlagen, die helfen sollen, eine Anfrage zu verfeinern. Die zum Generieren von Vorschlägen eingesetzten Algorithmen basieren im Allgemeinen auf einer statistischen Analyse von zusammen auftretenden Wörtern („term ooccurrence“). Der MetaGer-Web-Assoziator[11] schlägt dem Benutzer die am häufigsten zusammen mit einem Begriff auftretenden Wörter vor; die Yahoo-Suche[12] ermittelt Empfehlungen, wie Suchbegriffe zu Wortgruppen erweitert werden können. Abbildung 3.6 zeigt die vom MetaGer-Web-Assoziator ermittelten Begriffsvorschläge für die Anfrage `Agent Ensemble`. Die vorgeschlagenen Begriffe sind für eine Anfrageoptimierung ungeeignet. Für den Benutzer ist nicht ersichtlich, warum diese Worte vorgeschlagen werden.

Um das Verfolgen von aktuellen Entwicklungen innerhalb eines Themengebietes über längere Zeit zu erleichtern, bieten einige Informationssysteme die Möglichkeit, Suchbegriffe im Benutzerprofil zu speichern und die gespei-

[11] `http://www.metager.de/asso.html`
[12] `http://www.yahoo.de`

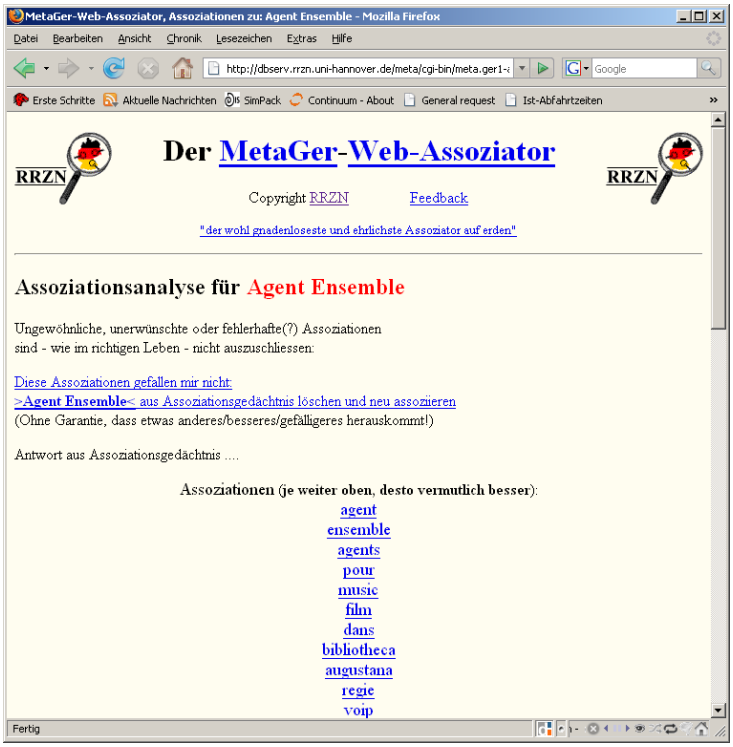


Abbildung 3.6: Die Abbildung zeigt die vom METAGER-WEB-ASSOZIATOR ermittelten Vorschlägen zum Begriff `Agent Ensemble`. [10.03.2008]

cherten Begriffe für spätere Suchanfragen erneut zu verwenden. Auf der Basis von Benutzerbewertungen werden Gewichtungen für die Suchbegriffe berechnet und das Benutzerprofil angepasst. Durch die inkrementelle Anpassung des Benutzerprofils an die vom Anwender gegebenen Bewertungen, kann ein komplexes Interessenprofil erstellt werden, auf dessen Basis Suchanfragen verfeinert und optimiert werden können.

Den Vorteilen der inkrementellen Profiladaption stehen verschiedene Nachteile gegenüber. So führt das automatische Erweitern von im Suchprofil gespeicherten Begriffen oft zu einem komplexen, nur schwer nachvollziehbaren Benutzerprofil. Durch die potentiell sehr große Anzahl von Schlüsselwörtern im Profil verliert der Benutzer den Überblick. Es besteht die Gefahr, dass sich sie Suchergebnisse stärker an den automatisch ermittelten Begriffen als an dem vom Benutzer vorgegeben Schlüsselwörtern orientieren. Derzeitige Systeme zur Profiladaption konzentrieren sich auf das Ermitteln von Begriffen und die Anpassung von Gewichtungen (auf Basis inhaltsbasierter Verfahren). Die Vielfältigkeit der für Gütebewertung relevanten Kriterien (z. B. Ansehen des Autors, Art des Dokumentes) bleibt dabei unberücksichtigt.

Informationssysteme, die auf Basis von Benutzerbewertungen Profile zur Optimierung von Suchanfragen erstellen, sind beispielsweise AWARESEARCH[13] und PERSONALIZED GOOGLE[14].

3.2.5 Diskussion

Die vorgestellten Informationssysteme zur Unterstützung der themenbereichsfokussierten Suche zeigen unterschiedliche Herangehensweisen, von denen jede spezifische Vor- und Nachteile aufweist. Die verwendeten Verfahren reichen dabei von Assistenten zum Ermitteln geeigneter Suchbegriffe (z. B. mit Hilfe des METAGER-WEB-ASSOZIATORs) über die Verwendung erweiterter Anfragen (die sich aus Begriffen und einer Domäne zusammensetzen bei der Suchmaschine ISI®) bis hin zu Systemen, die Suchergebnisse graphisch aufbereiten (z. B KARTOO). Bei der Realisierung eines adaptiven personalisierten Informationssystems sollten die Verfahren zum Erfassen der Benutzerpräferenzen und zum Filtern von Dokumenten berücksichtigt werden sowie in Abhängigkeit von der jeweiligen Suchanfrage und den Präferenzen des Nutzers geeignet mit weiteren Methoden kombiniert werden. Durch einen Assistent zum Verbessern von Suchanfragen sowie durch eine Zuordnung von Informationsquellen zu Themenbereichen (wie sie beispielsweise durch Verzeichnisdienste vorgenommen wird), kann die Leistungsfähigkeit eines Informationssystems gesteigert werden.

3.3 Personalisierte Informationssysteme

Moderne Informationssysteme sind oftmals sehr komplex und erfordern daher einen beträchtlichen Einarbeitungsaufwand. Um dem Benutzer trotzdem eine einfache und effiziente Nutzung zu ermöglichen, sollte ein Informationssystem die Fähigkeit besitzen, sich an die individuellen Präferenzen und Kenntnisse eines Benutzers anzupassen.

Der folgende Abschnitt gibt eine kurze Einführung in den Themenbereich *Personalisierung* und erläutert die im Rahmen von Informationssystemen eingesetzten Personalisierungsstrategien.

3.3.1 Ziele und Strategien der Personalisierung

Der Begriff *Personalisierung* bezeichnet in der Informationstechnik die Anpassung von Programmen, Diensten oder Informationen an die persönlichen

[13] `http://www.awaresearch.com/`
[14] `http://www.google.de/ig`

Vorlieben und Abneigungen, Bedürfnisse und Fähigkeiten eines Benutzers. Die Personalisierung soll dem Anwender eine verbesserte Nutzung eines Programms oder Service-Angebotes ermöglichen: Dienste oder Waren werden dazu so angeboten, dass diese den tatsächlichen oder vermuteten Präferenzen des Benutzers (bzw. Kunden) entsprechen.

Bei der Anpassung eines Dienstes an die individuellen Präferenzen des Benutzers unterscheidet man zwischen expliziter und impliziter Personalisierung. Die explizite Personalisierung gibt dem Benutzer die Möglichkeit, eine Anwendung oder einen Dienst bezüglich seiner Wünsche manuell anzupassen, z. B. in dem er ein Farbschema oder das gewünschte Ausgabeformat explizit definiert. Im Gegensatz dazu bezeichnet die implizite Personalisierung das Ableiten von Benutzerpräferenzen aus dem Verhalten und den Bewertungen des Benutzers. Um das Benutzerverhalten zu erfassen, protokolliert man, wie der Benutzer das System benutzt. Dies kann z. B. dadurch erfolgen, indem man aufzeichnet, welche Funktionen der Benutzer wie häufig benutzt oder indem man Mausbewegungen und Tastatureingaben protokolliert.

Eine Voraussetzung für eine Personalisierung ist, dass der Benutzer durch das System identifiziert werden kann (beispielsweise dadurch, dass sich der Benutzer am System anmeldet).

3.3.2 Personalisierung im Information Retrieval

Im Bereich der Informationssysteme können Personalisierungsverfahren auf verschiedenen Teilelementen des Systems eingesetzt werden [AM05, EV03].

Für die Adaption der **Benutzeroberfläche** werden im Allgemeinen Verfahren der expliziten Personalisierung eingesetzt, die es dem Benutzer erlauben, das Aussehen der Benutzeroberfläche zu ändern sowie die Elemente der Oberfläche nach individuellen Wünschen anzuordnen.

Ein weiterer Bereich in dem Personalisierungsstrategien im Rahmen von Informationssystemen angewendet werden, ist das **Optimieren von Anfragen**. Auf Basis von Benutzerbewertungen werden Vorschläge zum Anpassen der Anfragen an die individuellen Präferenzen abgeleitet. Darüber hinaus können in einem Informationssystem die **Kriterien zur Dokumentenbewertung** und die Strategie zum **Zusammenstellen der Filterergebnismenge** personalisiert werden, wobei die folgenden Verfahren eingesetzt werden können:

- Die von einem Benutzer abgegebenen Bewertungen können zum Bestimmen der Relevanz neuer Dokumente herangezogen werden.
- Anhand von abgegebenen Bewertungen können Benutzer mit gemeinsamen Interessen ermittelt werden. Die Bewertungen von Benutzern

mit ähnlichen Interessen können als ein Kriterium bei der Bewertung von Dokumenten eingesetzt werden.

- Die Gewichtung der Kriterien, die bei der Bewertung von Dokumenten herangezogen werden, kann an die individuellen Benutzerpräferenzen angepasst werden.

- Die Strategie zum Zusammenstellen der Ergebnismenge kann an die persönlichen Vorlieben des Benutzers angepasst werden (z. B. indem, anstatt der Beschränkung der Filterergebnisse auf einen Themenbereich, eine Maximierung der Diversität der Dokumente angestrebt wird).

3.3.3 Probleme bei der Personalisierung

Bei der Personalisierung von Informationssystemen können verschiedene Schwierigkeiten auftreten, für die jeweils geeignete Lösungsstrategien gewählt werden müssen.

- **Cold Start Problem:** Um ein System an die Präferenzen des Benutzers anpassen zu können, ist es notwendig, die Interessen des Benutzers zu kennen. Registriert sich ein neuer Benutzer beim System, so stehen zumeist nur sehr wenige Daten über den Benutzer zur Verfügung. Um auch für neue Benutzer Methoden zur Personalisierung anwenden zu können, besteht ein möglicher Ansatz darin, dem Benutzer zunächst das Profil eines „Durchschnittsbenutzers“ (bzw. ein stereotypisches Benutzerprofil) zuzuordnen, dass jeweils bei Vorliegen neuer Daten über den Benutzer an die individuellen Präferenzen angepasst wird.

- **Moving Target Problem:** Die Interessen des Benutzers ändern sich. Es muss somit regelmäßig geprüft werden, inwiefern die Daten über den Benutzer noch aktuell sind und für eine Personalisierung verwendet werden können.

- **Einmalige Suche in einem neuen Themenbereich:** Sucht der Benutzer einmalig nach Dokumenten, die außerhalb seiner Interessenschwerpunkte liegen, so darf dies das Benutzerprofil nicht zu stark beeinflussen, da aus einer einmaligen Suche nicht auf eine starke Veränderung der Interessen des Benutzers geschlossen werden kann.

- **Mehrdeutigkeit von Feedback:** Das Verhalten des Benutzers und abgegebenen Bewertungen können unterschiedlich interpretiert werden.

Um zuverlässige Aussagen über die Benutzerinteressen ableiten zu können, muss eine hinreichend große Menge von Bewertungen zur Verfügung stehen. Ggf. kann auch ein Nachfragen beim Benutzer helfen, Unklarheiten zu beseitigen (Verwendung von Verfahren des aktiven Lernens).

- **Ressourcenbedarf:** Die Auswertung des Benutzerverhaltens, das Anpassen des Benutzerprofils sowie das Erstellen personalisierter Empfehlungen erfordern im Vergleich zu unpersonalisierten Systemen zusätzliche Ressourcen. Um dieses Problem zu lösen, können Verteilungs- und Load-Balancing-Strategien (z. B. auf Basis von Multi-Agentensystemen) eingesetzt werden.

- **Verwendung persönlicher Daten:** Hat ein Benutzer Vorbehalte gegen das Erstellen von Benutzerprofilen, so können Verfahren der „anonymen Personalisierung“ verwendet werden. Man ermittelt dabei anhand der aus der aktuellen Session verfügbaren Daten die Benutzer, die dem aktuellen Benutzer ähnlich sind, und verwendet die zugeordneten Benutzerprofile zum Adaptieren der Session.

Bei der Personalisierung von Informationssystemen müssen zahlreiche Schwierigkeiten beachtet werden und Methoden angewendet werden, die in Abhängigkeit vom jeweils betrachteten Szenario geeignete Lösungen ermöglichen. Dies kann beispielsweise durch eine intelligente Kombination verschiedener Filter- und Personalisierungsalgorithmen erfolgen.

3.4 Systeme für die personalisierte Suche

Eine Voraussetzung für den Einsatz von Verfahren für die personalisierte Filterung ist, den Benutzer und seine Interessen gut zu kennen. Ausgewählte Systeme, bei denen Verfahren zur Benutzermodellierung und Strategien zur personalisierten Filterung zum Einsatz kommen, werden in den folgenden Abschnitten vorgestellt und diskutiert.

3.4.1 Web Personae

Das System Web Personae [MKS02] generiert Benutzerprofile auf Basis der vom Benutzer besuchten Informationsangebote („Webseiten“). Es wird davon ausgegangen, dass ein Benutzer mehrere Interessengebiete hat, zu denen er regelmäßig Dokumente sucht. Jedes Interessengebiet wird im System

durch ein Profil („personae“) repräsentiert. Dabei werden die für ein Interessenprofil relevanten Daten durch die Auswertung des Benutzerverhaltens ermittelt. Die vom Benutzer besuchten Dokumente werden geclustert (unter Verwendung des Kosinus-Abstandes auf Wortstamm-Häufigkeitsvektoren), wobei ein Hierarchical agglomerative clustering-Algorithmus zum Einsatz kommt. Jeder Cluster repräsentiert ein Interessengebiet des Benutzers. Besucht ein Benutzer ein neues Dokument, so wird ermittelt, zu welchem der Interessengebiete des Benutzers dieses Dokument am besten passt, indem die Ähnlichkeit des Dokumentes mit den Clustermittelpunkten verglichen wird. Die vom System WEB PERSONAE durch das Clustern von Dokumenten ermittelten Daten, sollen dem Benutzer helfen, wichtige Wörter aus den Interessenschwerpunkten zu identifizieren, und das Einordnen neuer Dokumente erleichtern.

Problematisch bei dem verwendeten Vorgehen ist, dass nach dem Hinzufügen neuer Dokumente die Cluster neu berechnet werden müssen, wodurch Dokumentengruppierungen entstehen können, die sich von den vorherigen Clustern deutlich unterscheiden. Dies ist im Allgemeinen für den Benutzer nur schwer nachvollziehbar.

3.4.2 WebACE

WEBACE [HBG+98] ist ein webbasiertes System, das das Verhalten des Benutzers analysiert und Vorschläge zur Optimierung von Suchanfragen generiert. Das Benutzerverhalten wird dabei unter Verwendung eines Proxy-Servers aufgezeichnet, der für den Benutzer transparent, die besuchten Webseiten sowie die Verweilzeit auf den betrachteten Internetseiten protokolliert. Für die Webseiten, die der Benutzer häufig besucht und für die Webseiten, auf denen der Benutzer lange Zeit verweilt, werden die Häufigkeiten der auftretenden Wörter bestimmt, um für die Webseiten charakteristische Begriffe abzuleiten. Die ermittelten Begriffe werden eingesetzt, um Anfragen zu erweitern, um weitere für den Benutzer interessante Dokumente zu finden und um die Relevanz von neuen Dokumenten zu bewerten.

Das Auswerten des Leseverhaltens des Benutzers ist ein wichtiger Faktor bei der Realisierung personalisierter Informationssysteme, da (ohne dass dem Benutzer zusätzlicher Aufwand entsteht) Empfehlungen generiert werden können. Problematisch bei diesem Ansatz ist, dass die Verwendung eines Proxy-Servers nur beschränkt möglich ist bzw. vom Benutzer nicht immer akzeptiert wird. Bei der Verwendung von impliziten Benutzerbewertungen besteht zudem die Gefahr, dass aus den gesammelten Daten fehlerhafte Rückschlüsse über den Benutzer abgeleitet werden, die zu nicht sinnvollen Empfehlungen führen. So kann eine lange Verweildauer auf einer Seite auf

eine besonders interessante Internetseite hindeuten; es kann aber beispielsweise auch bedeuten, dass die Internetseite irrelevant ist und der Benutzer zwischenzeitlich den Computer verlassen hat.

Das System WEBACE beschränkt sich beim Ermitteln von Wörtern für die Erstellung neuer Anfragen auf die statistische inhaltsbasierte Analyse von bewerteten Dokumenten. Kollaborative und semantische Bewertungsalgorithmen bleiben unberücksichtigt.

3.4.3 Syskill and Webert

Das System SYSKILL AND WEBERT [PMB96] generiert unter Verwendung eines Benutzerprofils individuelle Empfehlungen für Internetseiten. Das Profil für einen Benutzer kann dabei jeweils mehrere „Interessenbereiche“ umfassen. Das System bietet für jeden Interessenbereich eine Einstiegsseite zur Administration, die neben Referenzen zu den jeweils relevanten Datenquellen auch Schlagwörter zur Beschreibung des Themenbereiches beinhaltet. Beim Erstellen eines neuen Themenbereiches muss der Benutzer einen Namen sowie Referenzen auf Dokumente angeben, die für diesen Themenbereich relevant sind.

Das System unterscheidet drei Modi:

- Der erste Modus dient dazu, Bewertungen des Benutzers zu sammeln. In diesem Modus soll der Benutzer besuchte Internetseiten explizit bewerten. Es stehen dazu die Optionen „hot“ (zwei Daumen nach oben), „indifferent“ (ein Daumen nach oben, einen Daumen nach unten) und „cold“ (zwei Daumen nach unten) zur Verfügung. Die Bewertungen dienen als Trainingsdaten, um die Interessen des Benutzers zu lernen.

- Im zweiten Modus gibt das System dem Benutzer Hinweise, welche von einer aktuell besuchten Webseite ausgehenden Referenzen zu Dokumenten führen, die zum Interessenprofil des Benutzers passen. Dabei wird unter Verwendung eines BAYES-Klassifikators abgeschätzt, wie gut die referenzierten Dokumente zu den Interessen des Benutzers passen. Die ermittelten Bewertungen werden zum Annotieren der Referenzen verwendet.

- Der dritte Modus bietet die Möglichkeit, mit den durch Analyse des Benutzerverhaltens ermittelten Begriffen, Anfragen an eine Suchmaschine zu stellen und so neue relevante Dokumente zu ermitteln.

Das System SYSKILL AND WEBERT verbindet Verfahren zum Ermitteln von expliziten und impliziten Bewertungen mit inhaltsbasierten Strategien zum

Bewerten von Dokumenten und zum Generieren von Empfehlungen. Problematisch ist dabei vor allem, die Beschränkung auf eine Strategie zum Ermitteln von Begriffen aus bewerteten Dokumenten sowie die fehlende Unterstützung von kollaborativen Verfahren.

3.4.4 Letizia

LETIZIA [Lie95] ist ein System, das den Benutzer beim Navigieren durch das Internet unterstützt. Die Leitidee des Systems ist, das Verhalten des Benutzers zu verfolgen und daraus Vorschläge für interessante Seiten abzuleiten. LETIZIA analysiert die von der geöffneten Internetseite aus verlinkten Dokumente und berechnet unter Verwendung des Benutzerprofils die Relevanz dieser Dokumente für den Benutzer. Im Gegensatz zu Menschen, die Internetseiten typischerweise nach dem Prinzip der Tiefensuche zugreifen, wird in LETIZIA ein auf Breitensuche basierender Ansatz zum Bewerten von Seiten verwendet.

LETIZIA setzt beim Ermitteln der Interessen des Benutzers auf die Analyse des Benutzerverhaltens (implizites Feedback). Als Indizien für die Relevanz einer Webseite werden die vom Benutzer ausgeführten Aktionen verwendet, wie z. B. das Speichern von Dokumenten, das Zurückkehren zu einer in der Vergangenheit besuchten Webseite oder das Besuchen verlinkter Dokumente. Unter Verwendung verschiedener Heuristiken generiert das System (auf Basis des protokollierten Benutzerverhaltens) Empfehlungen, die in einem separaten Fenster dem Benutzer präsentiert werden.

Das System konzentriert sich auf die Analyse von impliziten Bewertungen. Durch die Beschränkung auf eine Strategie zum Erfassen der Benutzerinteressen und die Mehrdeutigkeit von impliziten Bewertungen ist die Qualität der abgeleiteten Vorschläge begrenzt.

3.4.5 Let's Browse

LET'S BROWSE [LDV99] ist eine Erweiterung des LETIZIA Systems und wurde ursprünglich entwickelt, um gemeinsame Interessen von Benutzern zu ermitteln und Konferenzteilnehmern mit ähnlichen Interessen bei der Suche von interessanten Informationen zu unterstützen. Die Benutzer werden dabei durch elektronische Sender identifiziert, die über Infrarot Daten austauschen. Die Benutzerprofile werden offline unter Verwendung eines Web-Crawlers erstellt, der ausgehend von der persönlichen Webseite des Benutzers (bzw. seiner Organisation) die verlinkten Dokumente analysiert. Zusätzlich wird für jede Webseite, die ein Nutzer besucht, eine Termfrequenz-Statistik erstellt

und mit dem Benutzerprofil verglichen. Auf diese Weise wird ermittelt, ob die gerade aufgerufene Seite zum Interessenprofil des Benutzers passt.

Das System LET'S BROWSE verbindet Verfahren der inhaltsbasierten Filterung mit kollaborativen Verfahren. Interessant ist dabei vor allem der Einsatz von Infrarot-Sendern, die als Ergänzung zur Filterung von Dokumenten das Finden von Benutzern mit ähnlichen Interessen ermöglichen.

3.4.6 Web Watcher

Das System WEB WATCHER [JFM97] stellt dem Benutzer einen Softwareagenten als „Tourguide" zur Seite, der den Anwender bei der Navigation im Internet unterstützen soll. Der Agent protokolliert das Verhalten des Benutzers und empfiehlt Dokumente, die zur aktuell besuchten Internetseite passen. Um die Vorschläge den Wünschen des Benutzers anzupassen, analysiert der Agent, welchen der gemachten Vorschläge der Benutzer folgt. Darüber hinaus kann der Benutzer Vorschläge des WEB WATCHER Agenten explizit bewerten. Hat der Agent hinreichend lange das Benutzerverhalten verfolgt und ist der Agent damit auf die Interessen des Benutzers trainiert, kann der WEB WATCHER Agent besuchte Webseiten durch das Einfügen von Metadaten und das Verweisen auf weiterführende Dokumente anreichern. Der Agent agiert dabei als Proxy, der die besuchten Internetseiten folgendermaßen modifiziert:

- Auf der aktuell besuchten Webseite werden Referenzen auf Dokumente hervorgehoben, die der WEB WATCHER-Agent als besonders empfehlenswert für den Benutzer einstuft.

- In die besuchte Webseite wird eine Referenz auf das WEB WATCHER-Portal eingefügt, damit der Benutzer direkt auf die Konfiguration des WEB WATCHER-Agenten zugreifen kann (z. B. um die persönlichen Einstellungen anzupassen).

- Die Referenzen auf einer besuchten Web-Seite werden so angepasst, dass WEB WATCHER als Proxy-Server verwendet wird und damit auf die Dokumente schneller zugegriffen werden kann (d.h. die Dokumente werden durch den Proxy-Server vorab geladen und zwischengespeichert).

Zum Ermitteln von Dokumentenempfehlungen bietet das System WEB WATCHER zwei Verfahren.

- Für potentiell interessante Dokumente (d. h. Inhalte, die von der aktuellen Webseite verlinkt sind), wird eine Termfrequenz-Statistik erstellt

und die Ähnlichkeit zu vom Benutzer positiv bewerteten Dokumenten berechnet (auf der Basis von Tf – Idf Vektoren).

- Unter Verwendung von Reinforcement-Learning wird ein Navigationspfad ermittelt, durch den der Benutzer zu möglichst vielen für ihn relevanten Webseiten gelangt. Anhand des ermittelten Pfades wird dem Benutzer die als nächstes zu besuchende Webseite vorgeschlagen.

Das System WEB WATCHER verknüpft Verfahren zum Ermitteln von impliziten und expliziten Bewertungen mit inhaltsbasierten Dokumentenbewertungsalgorithmen. Das System ist auf das Vorschlagen von Webseiten beschränkt und lässt Verfahren des kollaborativen Filterns unberücksichtigt.

3.4.7 Keyword Highlighting

Das in [YMI02] vorgestellte System verfolgt den Ansatz, anhand einer Menge von besuchten Internetadressen, relevante Begriffe zu identifizieren. Zum Erfassen der vom Benutzer aufgerufenen Webseiten wird ein Proxy-Server eingesetzt. Es wird davon ausgegangen, dass die Wörter, die häufig in den vom Benutzer besuchten Seiten auftreten, dazu geeignet sind, die Interessen des Benutzers zu beschreiben. Die extrahierten Begriffe werden im Benutzerprofil gespeichert und zur Beschreibung der Nutzerinteressen verwendet. Öffnet der Benutzer ein neues Dokument, so werden in diesem die Begriffe hervorgehoben, die zum Profil des Benutzers gehören. Dies soll dem Benutzer das Auffinden interessanter Stellen im neuen Dokument zu erleichtern.

Das System bietet dem Benutzer nur eine geringe Unterstützung beim Finden relevanter Informationen. Das Hervorheben von Begriffen ist hilfreich, wenn in einem umfangreichen Dokument die Absätze identifiziert werden sollen, die besonders interessant sind.

3.4.8 WebMate

WEBMATE [CS98] ist ein System, welches das Ziel verfolgt, verschiedene Information Retrieval-Techniken zu integrieren. Das System versucht, die für den Benutzer interessanten Themenbereiche zu identifizieren und unter Verwendung von Wortvektoren (mit fester Länge) zu beschreiben. WEBMATE generiert die Benutzerprofile auf der Basis der aktuellen Session, so dass keine Anmeldung des Benutzers erforderlich ist.

Besucht der Benutzer eine neue Seite, so wird unterstellt, dass diese Seite für den Nutzer interessant ist. Die charakteristischen Begriffe der Seite werden unter Verwendung des Tf – Idf-Algorithmus ermittelt und in einem

neuen Wortvektor gespeichert. Enthält das bisherige Interessenprofil weniger als die angestrebte Anzahl von Wortvektoren, wird der neue Wortvektor dem Benutzerprofil hinzugefügt. Ist die Anzahl der angestrebten Wortvektoren bereits erreicht, so wird der neue Wortvektor mit dem ähnlichsten bereits bestehenden Vektor zusammengeführt, indem nur die am höchsten bewerteten Wörter übernommen werden. Die erstellten Vektoren aus Wörtern verwendet WebMate als Grundlage für verschiedene Dienste: So kann der Benutzer einen persönlichen Newsletter abonnieren. Das System prüft dazu neu verfügbare Dokumente auf ihre Ähnlichkeit mit den gespeicherten Wortvektoren und stellt dem Benutzer die Dokumente zu, deren Ähnlichkeit zu einem Interessengebiet größer ist als eine vordefinierte Schwelle. Des Weiteren werden die Wortvektoren genutzt, um den Benutzer beim Optimieren seiner Anfrage bzw. beim Erstellen neuer Anfragen zu unterstützen.

Das System WebMate verknüpft Verfahren des inhaltsbasierten Filterns mit Methoden zum Gruppieren von Interessensgebieten. Die dabei eingesetzten Verfahren konzentrieren sich auf einfache schlüsselwortbasierte Algorithmen und das Clustern der ermittelten Wortvektoren.

3.4.9 Diskussion der Informationssysteme

Die vorgestellten Systeme zur Benutzermodellierung und zum Personalisieren der Suche demonstrieren unterschiedliche Lösungsansätze, wie in Informationssystemen die Interessen des Benutzers erfasst und für die Generierung von Vorschlägen genutzt werden können. Die verwendeten Algorithmen eignen sich dabei meist nur für das ausgewählte Szenario und konzentrieren sich im Allgemeinen auf inhaltsbasierte Bewertungsverfahren. Um für einen größeren Einsatzbereich hochqualitative personalisierte Filterergebnisse und Empfehlungen bereitzustellen, sollten in einem Informationssystem verschiedene Verfahren zum Ermitteln von (expliziten und impliziten) Bewertungen berücksichtigt und mehrere Algorithmen (inhaltsbasierte *und* kollaborative) zum Bewerten von Dokumenten geeignet kombiniert werden. Dies erlaubt es, die Schwächen der Verwendung eines einzelnen Algorithmus zu umgehen und für eine größere Anzahl von Anwendungsfällen gute Ergebnisse zu ermitteln.

3.5 Verteilte Systeme

An Systeme zur personalisierten Informationsfilterung werden hohe Anforderungen bezüglich Erweiterbarkeit, Adaptivität, Offenheit, Skalierbarkeit und Sicherheit gestellt. Ein Ansatz diesen Anforderungen gerecht zu werden, besteht in der Verwendung einer serviceorientierten Architektur, bei der die

Komponenten des Systems als autonome Entitäten modelliert werden, die über Dienste untereinander kommunizieren.

Der folgende Abschnitt gibt einen Überblick über Technologien zur Realisierung offener verteilter intelligenter Systeme. Dabei wird detailliert die Multi-Agentenplattform JIAC IV vorgestellt, die als eine Möglichkeit zum Verknüpfen der Komponenten des implementierten Filterframeworks eingesetzt wird.

3.5.1 Multi-Agentensysteme

Im Gegensatz zu traditionellen Anwendungen, bei denen die Systemkomponenten statisch miteinander verknüpft werden, bestehen agentenbasierte Systeme aus einer Vielzahl intelligenter, lose miteinander gekoppelten Entitäten („Agenten"). Der Begriff *Agent* beschreibt dabei eine autonom agierende Softwareentität, die eine Funktionalität oder einen Dienst anbietet. Softwareagenten interagieren mit anderen Agenten auf der Basis von Diensten unter Verwendung einer Agenten-Kommunikationssprache [Ses02]. Merkmale von Agenten sind ihre Autonomie, die Fähigkeit zur Kommunikation sowie die Verwendung von Methoden der künstlichen Intelligenz [WJ95]. Software-Agenten erweitern das aus der objektorientierten Programmierung bekannte Konzept des *Objektes*, indem Agenten über eine eigene Ablaufsteuerung verfügen, die das autonome Initiieren von Kommunikationsprozessen und Lernverfahren ermöglicht. Für die Kommunikation werden meist semantische (ontologiebasierte) Datenmodelle und Kommunikationsprotokolle eingesetzt.

Merkmale von Multi-Agentensystemen

Bei der Realisierung verteilter adaptiver Systeme ist der Einsatz von Multi-Agentensystemen mit verschiedenen Vorteilen verbunden [Jen00, Syc98]:

- **Reduktion der Systemkomplexität:** Da das Gesamtsystem in autonom agierende, spezialisierte Softwareagenten zerlegt wird, besitzen die von den einzelnen Agenten übernommenen Aufgaben eine begrenzte Komplexität. Zum Verknüpfen der Agenten bieten Agentenframeworks (Agentenplattformen) im Allgemeinen Koordinations- und Kommunikationsmechanismen an.

- **Anpassung des Systems zur Laufzeit:** Da die Agenten nicht statisch miteinander verbunden sind, sondern erst zur Laufzeit die jeweils notwendigen Kooperationspartner ermittelt werden, können fehlerhafte oder uneffektive Entitäten zur Laufzeit des Systems durch verbesserte Agenten ersetzt werden, ohne das Gesamtsystem zu beeinträchtigen.

Darüber hinaus ist es möglich, neue Agenten und Funktionalitäten zu integrieren, ohne das System neu zu starten.

- **Verteilung von Systemkomponenten im Netzwerk:** Multi-Agentensysteme ermöglichen die transparente Verteilung von Agenten in einem Computernetzwerk. Die transparente Verteilung von Systementitäten erlaubt es, verfügbare Hardware-Ressourcen optimal auszunutzen und an den Bedarf anzupassen.

- **Adaptive Lastverteilung:** Unter Nutzung der Möglichkeit, Agenten transparent in einem Netzwerk zu verteilen, können adaptive Ansätze zur Lastverteilung realisiert werden. Dies erlaubt es beispielsweise, zur Laufzeit des Systems zusätzliche Agenten hinzuzufügen, um eine Überlastung von Agenten zu vermeiden.

- **Verbesserung der Ausfallsicherheit:** Durch die Verteilung des Systems auf autonome Entitäten kann eine Reduzierung der Störanfälligkeit erreicht werden. Bei Ausfall eines Agenten wird nicht das komplette System in der Stabilität gefährdet: Aufgaben können an alternative Agenten delegiert werden sowie fehlerhafte Entitäten des Systems durch verbesserte Agenten ausgetauscht werden.

- **Wiederverwendbarkeit:** Softwareagenten bieten spezifische Dienste und Funktionalitäten an. Werden bereits implementierte Funktionen in einem neuen Projekt benötigt, so können diese Funktionen dem neuen System zur Verfügung gestellt werden, indem die benötigten Agenten dem System hinzugefügt werden.

- **Fähigkeiten zur Selbstheilung:** Um auf Fehlerfälle flexibel reagieren zu können, werden in Multi-Agentensystemen Verfahren der künstlichen Intelligenz eingesetzt, die es ermöglichen, Fehler zu erkennen (z. B. eine ungültige Parameterkonfiguration oder unzureichender Speicherplatz) und durch das Initiieren geeignete Aktionen zu behandeln (z. B. durch den Neustart einer Komponente, den Austausch eines Agenten oder durch das Benachrichtigen des Administrators).

- **Vereinfachtes Systemmanagement:** Multi-Agentenframeworks bieten oft vorgefertigte Komponenten und Dienste für das Management verteilter Systeme, die Integration von Verzeichnisdiensten, die Anbindung von Authentifikations- und Verschlüsselungsalgorithmen sowie eine Unterstützung von mehreren Kommunikations- und Transaktionsprotokollen.

- **Verwendung von semantischen Modellen:** Agentenkommunikationssprachen unterstützen im Allgemeinen die Verwendung von Ontologien und Sprechakten, so dass Informationen semantisch beschrieben werden und das Risiko der Fehlinterpretation von übertragenen Daten reduziert wird. Darüber hinaus erleichtert die semantische Beschreibung von Daten die Wiederverwendung bereits implementierter Dienste.

Neben den technischen Vorteilen von Agenten bei der Realisierung von adaptiven verteilten Systemen, wird die Agenten-Metapher oft als ein leicht verständliches Konzept bei der Modellierung komplexer Systeme verwendet. Ein Agent beschreibt dabei eine auf eine Aufgabe spezialisierte Komponente, die mit weiteren Agenten interagiert. So kann beispielsweise ein Agent als persönlicher Assistent fungieren, der den Benutzer bei der Nutzung des Systems unterstützt und ihn bei Abwesenheit vertritt.

Da in Multi-Agentensystemen komplexe (ontologiebasierte) Kommunikationsprotokolle verwendet sowie eine große Anzahl autonomer Entitäten verwaltet werden, kann die Verwendung einer Multi-Agentenplattform mit einem höheren Bedarf an Ressourcen verbunden sein (im Vergleich zu objektorientiert implementierten Systemen). Darüber hinaus ist die Erweiterbarkeit und Offenheit von Multi-Agentensystemen mit einem erhöhten Management- und Planungsaufwand verbunden, so dass Multi-Agentensysteme im Allgemeinen nicht echtzeitfähig sind.

Für einen Überblick zu Agentensystemen und ihre Abgrenzung zu objektorientierten Ansätzen wird auf [FG96, TIS+07, Jen00, Syc98] verwiesen.

Plattformen und Anwendungsbereiche

Anhand ausgewählter Szenarien werden die Einsatzgebiete von Multi-Agentensystemen erläutert und einige Agentenplattformen vorgestellt. Die Beispiele illustrieren die Vor- und Nachteile von Agentensystemen sowie die Besonderheiten der vorgestellten Plattformen.

- **Realisierung verteilter Systeme:** Da Agentenplattformen (z. B. JADE [BPR01b, FGL02], JACK [BRHL99], OOA [MCM99]) bereits Funktionen zur Kommunikation zwischen in einem Netwerk verteilten Entitäten bieten sowie Werkzeuge zur Verwaltung eines verteilten Systems beinhalten, eignen sich Multi-Agentensysteme sehr gut für die Implementierung verteilter Systeme. Einige Agentenplattformen (z. B. JIAC IV [FBK+01], SEMOA [RPP05]) bieten darüber hinaus eine Unterstützung für mobile Agenten. Dies ermöglicht es, zur Laufzeit des

Systems zusätzliche Hardware zu integrieren, Agenten auf einem Computer zu ersetzen und die Zuordnung von Agenten zu Hardware zur Laufzeit zu optimieren.

- **Verteilte Lernverfahren:** Da Agentensysteme auf autonomen, verteilten Softwareentitäten basieren, sind Agenten gut geeignet, um verteilte Lernverfahren und Lernen in partiell beobachtbaren Systemen zu modellieren. Die Unabhängigkeit und Verteilungstransparenz der Agenten kann dabei genutzt werden, um verschiedene Aufgaben zu parallelisieren und die Leistung eines Computersystems optimal auszunutzen [LA07].

- **Agenten zum Schutz personenbezogener Daten:** Agenten können im Auftrag eines Benutzers agieren und dadurch die Anonymität des Benutzers schützen [CA07]. Bei der Realisierung von Informationssystemen können beispielsweise personenbezogene Daten so gefiltert werden, dass jeder Agent jeweils nur die für die Erbringung eines Dienstes notwendigen Daten erhält. Die Identität des Benutzers wird nicht preisgegeben und seine persönlichen Daten geschützt. Für den Einsatz von Agentensystemen spricht dabei auch, dass einige Agentenplattformen eine Reihe von Sicherheitsmechanismen bereitstellen, wie das Verschlüsseln von Daten oder das Verwalten und Prüfen von Schlüsseln [FBK+01]. Damit ist es möglich die Vertrauenswürdigkeit von Dienstanbietern zu prüfen und Daten verschlüsselt zu übertragen. Im Rahmen von Informationssystemen können damit die Vertrauenswürdigkeit von Datenquellen und Filteragenten geprüft sowie die im Rahmen von Agentendiensten erforderlichen Daten gesichert übertragen werden.

- **Informationssysteme:** Bei der Suche nach Informationen müssen im Allgemeinen mehrere, verteilt vorliegende Datenquellen integriert werden. Dabei ist zu berücksichtigen, dass sich die Menge der relevanten Quellen verändern kann. Aufgrund ihrer Vorteile im Bereich der verteilten Systeme (Bereitstellung von Funktionen zur Kommunikation und zur Registrierung von Diensten) eignen sich Agentenframeworks zur Realisierung von Informationssystemen. Des Weiteren können durch die Verwendung von Multi-Agentensystemen die in Informationssystemen auftretenden Anforderungen bezüglich der Automatisierung von Routineaufgaben und dem Schutz persönlicher Daten erfüllt werden.

Analysiert man die Vorteile und die Einsatzgebiete von Multi-Agentensystemen, so erscheint der Einsatz einer Multi-Agentenplattform für

die Implementierung eines adaptiven, personalisierten Informationssystems zweckmäßig. Die Nachteile von Agentensystemen, wie fehlende Echtzeitfähigkeit oder teilweise nichtdeterministisches Verhalten, fallen bei Informationssystemen weniger ins Gewicht.

3.5.2 Ein Framework für die Entwicklung von intelligenten verteilten Systemen

Ein Multi-Agentenframework, das die effiziente Entwicklung und Administration intelligenter, verteilter Anwendungen ermöglicht, ist JIAC IV[15]. Die Basiskonzepte von JIAC IV umfassen die dienstbasierte Interaktion zwischen autonomen Agenten, ein Modell zur Integration von Komponenten und Diensten, Mechanismen zur Verteilung von Agenten in einem Netzwerk sowie Komponenten zur Autorisierung, Authentifikation und Abrechnung (AAA). Darüber hinaus bietet das Agentenframework Werkzeuge und Komponenten zur Administration und Erstellung von Agenten und Agentenplattformen.

Die Multi-Agentenplattform JIAC IV basiert auf einem Komponentenmodell, das eine BDI-Architektur[16] implementiert. Das Framework bietet vorgefertigte Module für die Realisierung grundlegender Agentenfähigkeiten, z. B. für die Kommunikation zwischen Agenten, für die Wissensverarbeitung und -speicherung sowie für die Planung von Aktionen. Es existieren Komponenten für das Management verteilter Systeme, die je nach Anwendungsszenario dem System als Bausteine hinzugefügt werden können.

Für die effiziente Entwicklung von Anwendungen beinhaltet das Multi-Agentenframework JIAC IV eine Entwicklungsumgebung und eine Software-Engineering-Methodologie. Der Entwickler erhält damit eine umfassende Unterstützung für alle Phasen der Anwendungsentwicklung, beginnend mit der Anforderungsanalyse und den Systementwurf über die Implementierung bis zum Deployment des Systems. Des Weiteren beinhaltet das Agentenframework Werkzeuge für die Konfiguration, Installation und Wartung eines Multi-Agentensystems [HKHA06].

Eine JIAC-basierte Anwendung besteht aus einer oder mehreren Agentenlaufzeitumgebungen (JIAC-Plattformen), die über heterogene Hardware verteilt werden können. Jede Laufzeitumgebung wird durch einen Manager-Agenten verwaltet, der Infrastrukturdienste (z. B. Verzeichnisdienste, Dienst zur Agenten-Migration) anbietet. Ein Agent ist jeweils aus mehreren Komponenten aufgebaut, die über ihre Rolle adressiert werden. Dies ermöglicht den Austausch von Komponenten zur Laufzeit sowie die Erweiterung eines

[15] Akronym für **J**ava based **I**ntelligent **A**gent **C**omponentware (Version 4)
[16] **B**elif, **D**esire, **I**ntesion

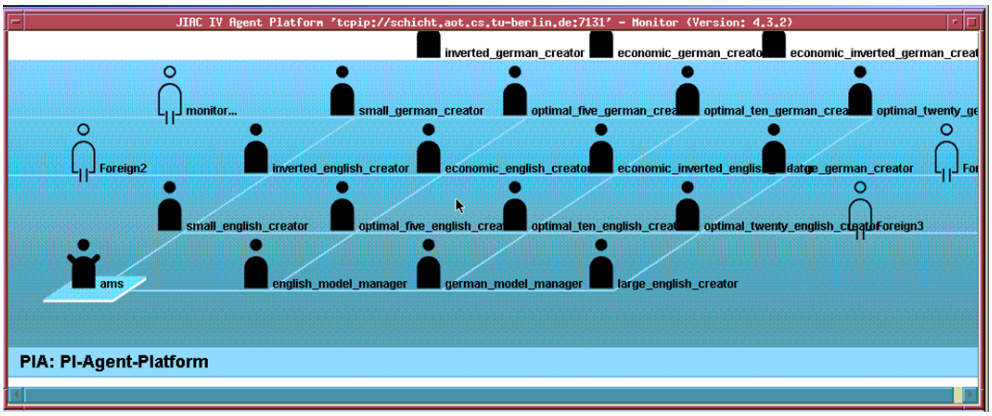


Abbildung 3.7: Die Abbildung zeigt am Beispiel des Informationssystems PIA den Administrationsmonitor einer JIAC IV Agentenplattform. Jede Plattform wird durch einen Manager-Agenten (im Bild links) verwaltet, der verschiedene Infrastrukturdienste für die auf der Plattform platzierten Agenten anbietet. Die Agenten sind jeweils auf einen Aufgabenbereich spezialisiert und interagieren untereinander über Dienste. Der Agentenmonitor bietet neben einer Visualisierung der Plattform verschiedene Funktionen zum Management der Agentenplattform an.

Agenten um zusätzliche Funktionalitäten. JIAC IV-Agenten verfügen über Autonomie und können unter Verwendung von Verfahren der Künstlichen Intelligenz und der Situationserkennung sowie Planungsalgorithmen entscheiden, wie Dienste erbracht werden sollen. Die Infrastrukturdienste der JIAC-Plattform ermöglichen das Hinzufügen und Entfernen von Agenten zur Laufzeit des Systems und bieten darüber hinaus die Unterstützung von starker Migration (d. h. Programmcode und Zustand können auf eine neue Plattform übertragen werden). Zum Schutz der übertragenen Daten und zum Verhindern von Manipulationen unterstützt JIAC IV mehrere Sicherheitsstandards [FBK+01, Sch02, Bun05]. Ein Beispiel für die Visualisierung einer JIAC IV-Agentenplattform zeigt die Abbildung 3.7.

Das Agentenframework JIAC IV stellt eine Entwicklungsumgebung und zahlreiche vorgefertigte Komponenten für die Implementierung von flexiblen, offenen, verteilten Systemen zur Verfügung. Von alternativen Agentenframeworks hebt sich JIAC IV durch die konsequente Unterstützung des gesamten Entwicklungsprozesses eines Agentensystems (von der Analyse bis zum Deployment), die große Anzahl bereits verfügbarer Komponenten, die Integration vielfältiger Sicherheitsmechanismen sowie die Unterstützung eines komplexen Wissensmanagements ab.

3.6 Zusammenfassung

In diesem Kapitel wurden Methoden für eine themenbereichsfokussierte Suche sowie für die personalisierte Informationssuche vorgestellt. Die Verfahren wurden anhand von existierenden Systemen illustriert und die Stärken und Schwächen der unterschiedlichen Ansätze diskutiert. Die durchgeführte Analyse zeigt, dass die eingesetzten Algorithmen jeweils nur für ein Szenario optimiert sind. Um für ein breiteres Spektrum von Anwendungsszenarien hochqualitative Ergebnisse zu erhalten, sollten unterschiedliche Algorithmen untereinander kombiniert werden. Die Kombination mehrerer Strategien kann unter Verwendung von Multi-Agentensystemen erfolgen. Multi-Agentensysteme vereinfachen die Entwicklung von verteilten, aus heterogenen Komponenten bestehenden Systemen. Die dienstbasierte Koppelung autonomer intelligenter Entitäten ermöglicht die Realisierung von fehlertoleranten, adaptiven Informationssystemen.

Kapitel 4

Eine Architektur für ein adaptives personalisiertes Informationssystem

Nachdem im Kapitel 2 die Anforderungen an ein personalisiertes Informationssystem erläutert und im Kapitel 3 Systeme zur personalisierten Filterung sowie Technologien zur Implementierung offener verteilter Anwendungen diskutiert wurden, wird in diesem Kapitel eine Architektur für ein personalisiertes adaptives Informationssystem abgeleitet. Es werden dabei die vielfältigen Anforderungen und Erwartungen an ein flexibles Filtersystem berücksichtigt und die bei der Modellierung gewählten Randbedingungen motiviert. Es wird ein Überblick über die Funktionen und die Verknüpfungen der einzelnen Elemente gegeben. Ausgewählte Funktionsbereiche (wie das Optimieren von Anfragen und das Zusammenfassen von Bewertungen) werden in den anschließenden Kapiteln im Detail erläutert.

4.1 Aufgabenbereiche und Funktionen

Die Aufgabenbereiche, die bei der Realisierung eines personalisierten adaptiven Informationssystems berücksichtigt werden müssen, sind neben dem personalisierten Filtern von Dokumenten das Bereitstellen einer Benutzerschnittstelle, das Verwalten von Benutzerprofilen, das Erfassen von Benutzerbewertungen sowie das Zustellen von Informationen (z. B. per Push-Diensten). Der Bereich der Filterung unterteilt sich in vier Aufgabengebiete:

- Die Integration heterogener Datenquellen und die Anpassung von Filteranfragen für die jeweilige Quelle

- Das Anpassen und Optimieren von Anfragen (Query-Forming)
- Das Ermitteln von Bewertungen und Metadaten zu potentiell relevanten Dokumenten
- Das Zusammenstellen von Ergebnismengen

Zu diesen vier Aufgabenbereichen kommt noch der Bereich der Ablaufoptimierung, der die Auswahl und Koordination der Filterkomponenten steuert. Darüber hinaus müssen für die Realisierung eines Informationssystems Komponenten für eine Benutzerschnittschnittstelle, das Erfassen des Benutzerverhaltens sowie zum Management von Benutzerprofilen implementiert werden. Die Aufgabebereiche eines personalisierten Informationssystems sowie die jeweiligen Verantwortlichkeiten zeigt die Abbildung 4.1.

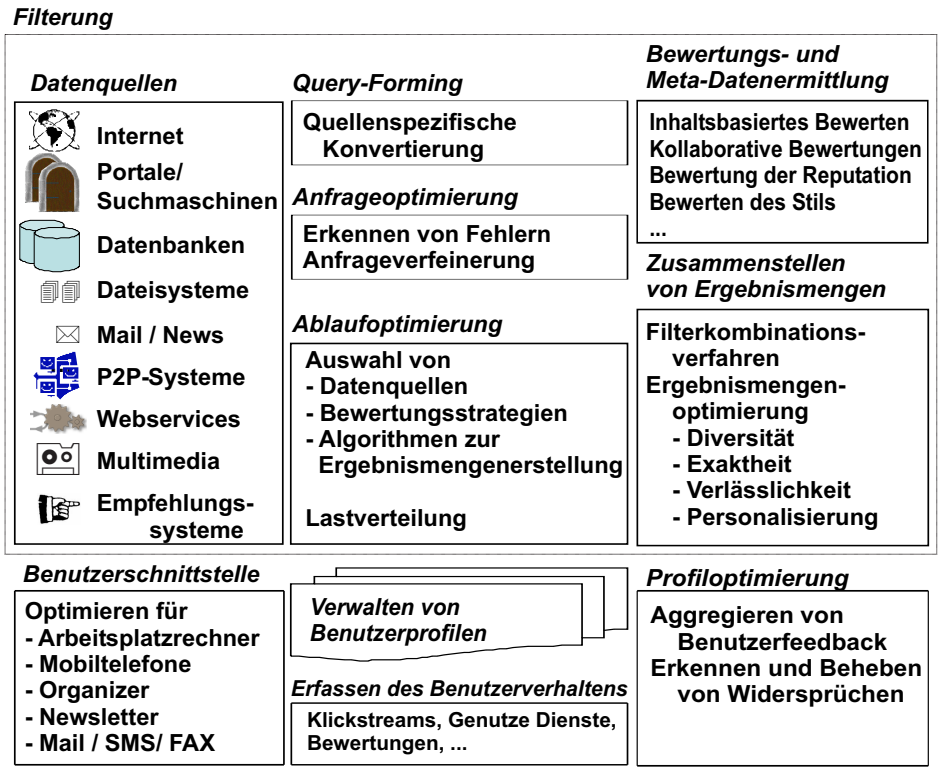


Abbildung 4.1: Die Abbildung zeigt die Aufgabenbereiche eines Informationssystems sowie die den Bereichen jeweils zugeordneten Funktionen.

Bevor die Filterung im Detail erläutert wird, werden zunächst die Komponenten für die Bereitstellung einer Benutzerschnittstelle und zum Verwalten von Benutzerprofilen erklärt.

Die Benutzerschnittstelle

Die Benutzerschnittstelle ist der für den Benutzer sichtbare Teil eines Informationssystems und muss in Abhängigkeit von der Zielgruppe des Systems und dem typischen Nutzungsszenario realisiert werden. Bei der Realisierung müssen die folgenden Anforderungen berücksichtigt werden:

- Die Benutzerschnittstelle muss dem Benutzer die Möglichkeit bieten, seinen Informationsbedarf zu beschreiben und Suchanfragen zu formulieren.
- Es müssen die Besonderheiten und der Funktionsumfang der vom Benutzer verwendeten Geräte (z. B. Displaygröße) beachtet werden.
- Die Benutzeroberfläche sollte es ermöglichen, (explizites und implizites) Benutzerfeedback zu erfassen.
- Neben der Möglichkeit zur interaktiven Nutzung des Systems, sollte die Benutzerschnittstelle auch Push-Dienste (z. B. Newsletter, SMS) anbieten.
- Es sollte die Möglichkeit bestehen, die Benutzerschnittstelle an die individuellen Präferenzen des Benutzers anzupassen (implizite und explizite Personalisierung).

An die Benutzerschnittstelle werden bei modernen personalisierten Informationssystemen hohe Anforderungen gestellt. Eine auf die Präferenzen und Kenntnisse des Benutzers abgestimmte Benutzerschnittstelle schafft die Basis für zum Informationsbedürfnis des Benutzers passende Suchanfragen und die intuitive Nutzung der ermittelten Filterergebnisse.

Benutzerprofile

Eine Voraussetzung für das Ermitteln personalisierter, auf die individuellen Wünsche des Benutzers abgestimmter Filterergebnisse ist es, die Präferenzen des Benutzers zu verwalten. Zum Speichern der Vorlieben und Erfahrungen des Benutzers können Benutzerprofile verwendet werden, an die in der Domäne der personalisierten Informationssysteme die folgenden Anforderungen gestellt werden:

- Das Benutzerprofil muss die Speicherung unterschiedlicher Arten von Präferenzen ermöglichen, wie z. B. Suchbegriffe, semantisch annotierte Terme (z. B. Autoren) und bevorzugte Datei-Formate.

- Um in einem Informationssystem vielfältige Algorithmen zum Bewerten von Dokumenten einsetzen zu können, muss das Benutzerprofil unterschiedliche Arten von Daten unterstützen, wie z. B. Trainingsdaten für Klassifikationsalgorithmen, Regelmengen, Benutzerbewertungen und Parameterkonfigurationen (z. B. für künstliche neuronale Netze). Um sicherzustellen, dass neue Filteralgorithmen in das Informationssystem integriert werden können, muss das Benutzerprofil erweiterbar gestaltet sein.

- Zusätzlich zu den direkt für die Filterung notwendigen Daten muss das Benutzerprofil Informationen zum Benutzer (Anmeldeberechtigungen, Abrechnungsmodell) sowie die Präferenzen bezüglich der Benutzerschnittstelle und der Konfiguration von Assistenten erhalten (z. B. für Zustellung von Push-Nachrichten oder die regelmäßige Überwachung von Informationsquellen).

- Um die Akzeptanz eines Systems zu verbessern, sollte der Benutzer die im Profil gespeicherten Informationen einsehen können. Zudem sollten Änderungen am Benutzerprofil protokolliert werden, damit der Anwender die Herkunft der Daten nachvollziehen kann („Transparenz und Nachvollziehbarkeit“).

Die zur Beschreibung der Interessen des Benutzers verwendeten Profile können unterschiedlich detaillierte Daten aufweisen. Da ein Benutzer im Allgemeinen mehrere Interessen besitzt, wird meist für jedes Interessensgebiet ein separates Profil („Anfrageprofil“) erstellt. Benutzer mit ähnlichen Interessen können zu einer Gemeinschaft („Community“) zusammengefasst werden, der ein eigenes Profil („Community-Profil“) zugeordnet wird. Das Profil eines Benutzers setzt sich daher aus verschiedenen Teilkomponenten zusammen („Anfrageebene“, „Benutzerebene“, „Communityebene“, „Gesamtheit der Benutzer“). Die Beziehung zwischen den Teilkomponenten eines Benutzerprofils visualisiert die Abbildung 4.2.

Um den Benutzer beim Vervollständigen und Optimieren seines Benutzerprofils zu unterstützen, sollte ein Informationssystem über eine Komponente verfügen, die Hilfestellung für das Erkennen und Beseitigung von Fehlern anbietet sowie dem Benutzer beim Verfeinern assistiert. Grundlage für die Adaption des Benutzerprofils und die Optimierung von Informationsanfragen bildet die Erfassung und Auswertung des Benutzerverhaltens.

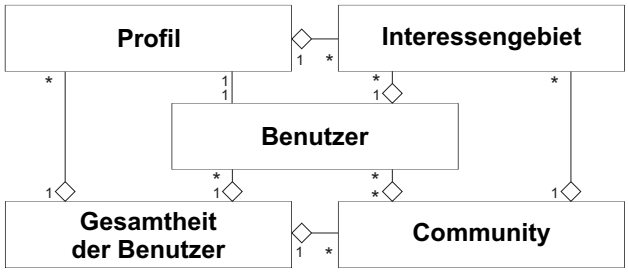


Abbildung 4.2: Die Abbildung zeigt die Komponenten und die Beziehungen zwischen den Elementen eines Benutzerprofils.

Auswertung des Benutzerverhaltens

Aus dem Verhalten des Benutzers lassen sich Rückschlüsse auf die Interessen und Präferenzen des Benutzers ableiten. Ein Informationssystem sollte daher über die Fähigkeit verfügen, das Benutzerverhalten zu erfassen und anhand der erfassten Daten Wissen zu den Interessen des Benutzers zu extrahieren. In einem Informationssystem kann dabei beispielsweise protokolliert werden, welche Dokumente der Benutzer öffnet, wie lange der Benutzer vorgeschlagene Dokumente geöffnet lässt, welche Aktionen der Benutzer mit den Dokumenten ausführt (z. B. Drucken, Speichern, als E-Mail-Versenden) und welche Geräte der Benutzer bevorzugt einsetzt. Die erhobenen Daten können für die Generierung von Vorschlägen zur Optimierung des Benutzerprofils sowie für die Berechnung kollaborativer Bewertungen eingesetzt werden.

4.2 Ablauf einer personalisierten Suche

Das Ermitteln personalisierter Filterergebnisse stellt einen komplexen Prozess dar, bei dem verschiedene Verfahren kombiniert und mehreren Aufgabenbereichen zugeordnete Komponenten integriert werden müssen. Im Folgenden werden die für eine Suche notwendigen Schritte erläutert und beispielhaft der Ablauf des Filterprozesses dargestellt:

- **Definition des Informationsbedarfes:** Als Ausgangspunkt für eine Suche dient eine Anfrage, mit der der Benutzer seinen Informationsbedarf beschreibt. Dies kann z. B. durch Suchbegriffe oder die Auswahl von Kategorien erfolgen.
- **Auswahl der Informationsquellen:** Auf Basis der Anfrage, dem Benutzerprofil und der aktuellen Auslastung des Systems wählt das Informationssystem die am besten passenden Datenquellen aus.

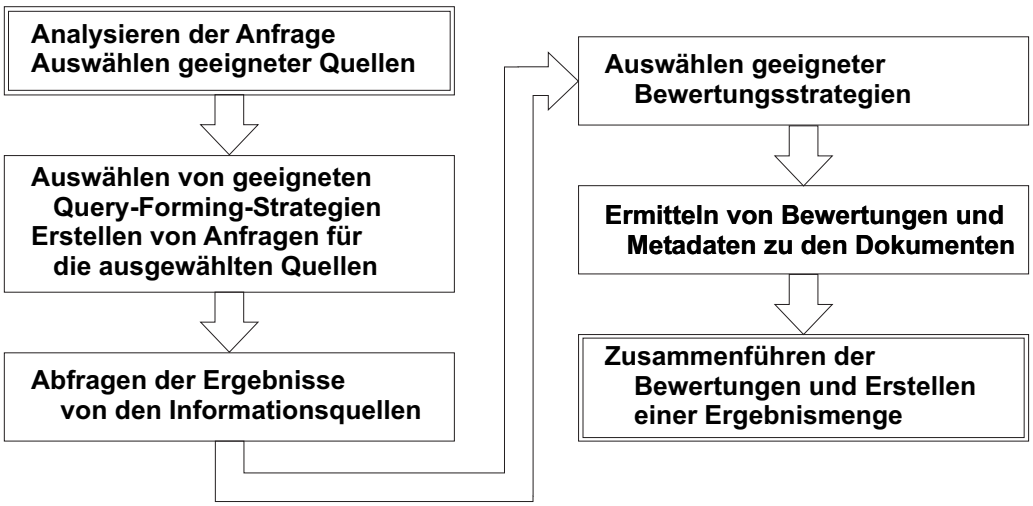


Abbildung 4.3: Die Abbildung zeigt die für eine personalisierte Informationssuche notwendigen Schritte.

- **Ansteuerung der Informationsquellen („Query-Forming“):** Für jede ausgewählte Quelle wird analysiert, welche Teile des Benutzerprofils für die Bearbeitung der Anfrage relevant sind. Unter Berücksichtigung der definierten Datenschutzrichtlinien werden die relevanten Teile des Benutzerprofils in das für Informationsquelle passende Datenformat konvertiert.

- **Abfragen der Dokumente von den Informationsquellen:** Die von den Informationsquellen ermittelten Dokumente werden in ein einheitliches Datenformat überführt. Dieses Format kann zusätzlich zum ermittelten Dokument auch Informationen über das Dokument enthalten (z. B. Relevanzbewertung durch die Datenquelle).

- **Ermitteln von Bewertungen und Metadaten:** In Abhängigkeit von der Informationsanfrage und dem Benutzerprofil werden die von den Informationsquellen gelieferten Dokumente auf Basis verschiedener Kriterien einheitlich bewertet. Außerdem werden Metadaten zu den Dokumenten ermittelt.

- **Zusammenstellen der Ergebnismenge:** Aus der Menge der potentiell relevanten Dokumente werden auf Basis der ermittelten Bewertungen und Metadaten die am besten zum Informationsbedarf des Benutzers passenden Dokumente ermittelt. Dabei können verschiedene Optimierungs- und Bewertungskombinationsverfahren (Heuristiken, Constraint-Solver) eingesetzt werden.

- **Präsentation der Ergebnisse:** Die ermittelten Dokumente werden dem Benutzer präsentiert. Das System analysiert, wie der Benutzer die gelieferten Dokumente verwendet (implizites Feedback). Zusätzlich werden die vom Benutzer abgegebenen Bewertungen erfasst (explizites Feedback).

- **Auswerten des Benutzerverhaltens:** Auf Basis der gesammelten (expliziten und impliziten) Bewertungen können Vorschläge zur Adaption des Benutzerprofils und zur Optimierung der Suchanfragen generiert werden.

In Abhängigkeit vom jeweils betrachteten Szenario kann der Ablauf angepasst werden, indem einzelne Berechnungsschritte parallelisiert, übersprungen oder wiederholt ausgeführt werden. Die für die jeweiligen Schritte verwendeten Systemkomponenten müssen jeweils so realisiert werden, dass sie an die spezifischen Anforderungen einer Suchanfrage angepasst werden können und neue Strategien (z. B. zusätzliche Informationsquellen, Bewertungs- und Auswahlstrategien) integriert werden können. Des Weiteren muss sichergestellt werden, dass die für die einzelnen Schritte eingesetzten Komponenten optimal aufeinander abgestimmt werden. Schwächen in einem Schritt oder bei der Verknüpfung zweier Schritte können die Güte des gesamten Prozesses beeinträchtigen.

4.3 Schichtenmodell

Anhand der für eine personalisierte Informationssuche notwendigen Schritte und Aufgabenbereiche wird eine dreischichtige Architektur für das Informationssystem vorgeschlagen (Abbildung 4.4).

Die oberste Ebene des Systems umfasst die Komponenten zur Optimierung von Anfragen, zur Integration von Datenquellen und zur Anpassung der Anfragen an die Besonderheiten der jeweiligen Quellen. Ausgehend von einer Suchanfrage wird in dieser Schicht eine Menge von potentiell relevanten Dokumenten (z. B. Texte, Musik, Filme) ermittelt.

Die zweite Ebene des Systems beinhaltet Komponenten zum Berechnen von Metadaten und von Bewertungen für die als potentiell relevant ermittelten Dokumente. Dabei müssen unterschiedliche Arten von Dokumentenverarbeitungs- und Bewertungsstrategien berücksichtigt werden. Darüber hinaus muss sichergestellt werden, dass neue Algorithmen zum Ermitteln von Bewertungen und Metadaten integriert werden können. Als Eingabe für diese Schicht dient eine Menge von Dokumenten; die ermittelten Bewertungen werden als Metadaten den Dokumenten hinzugefügt.

In der dritten Ebene des Systems werden auf Basis der ermittelten Metadaten und Bewertungen optimierte Ergebnismengen zusammengestellt. Dazu werden die in der zweiten Ebene ermittelten Bewertungen entsprechend der individuellen Präferenzen des Benutzers und dem jeweilige Suchkontext kombiniert.

Für jede Ebene existiert jeweils eine Managementkomponente, die die Entitäten der jeweiligen Schicht steuert und überwacht, z. B. indem sie die Zuordnung von Filteraufgaben zu Komponenten übernimmt, Dienste für die Registrierung von Komponenten anbietet und das Protokollieren des Laufzeitverhaltens übernimmt.

Zum Ermitteln der Ergebnisse für eine Suchanfrage durchläuft der Filterprozess die drei Schichten des Informationssystems. Die in jeder Ebene verwendeten Komponenten („Agenten“) werden jeweils dynamisch ausgewählt. Im Allgemeinen erfolgt die Bearbeitung der Anfrage innerhalb jeder Ebene parallel durch mehrere Agenten: So werden beispielsweise jeweils mehrere Datenquellen parallel abgefragt und verschiedene Bewertungskriterien nebenläufig berechnet. Um die mehrfache Ausführung von Berechnungen zu vermeiden, können Abhängigkeitsbeziehungen zwischen Entitäten definiert werden. So ist beispielsweise das Zerlegen eines Dokumentes in Wörter und das Durchführen einer Wortstammreduzierung („Stemming“) Voraussetzung für das Berechnen der Ähnlichkeit einer Anfrage zu einem Dokument auf Basis einer Tf − Idf-Bewertung und auch für das Ermitteln einer Bewertung mit dem BM25-Algorithmus [RWB98]. Das Bereitstellen von Zwischenergebnissen (z. B. Dokumente als eine Menge von Wörtern) ermöglicht es, doppelte Berechnungen zu vermeiden und so Ressourcen zu sparen.

Für die Bearbeitung einer Anfrage können die drei Ebenen des Informationssystems sequentiell durchlaufen werden. Allerdings ist es in Szenarien, bei denen es auf die besonders schnelle Bereitstellung von Ergebnissen ankommt, auch möglich, dass eine Anfrage auf mehreren Ebenen parallel bearbeitet wird. So können beispielsweise aus der Menge von potentiell relevanten Dokumenten auch ohne umfangreiche Bewertungen und Metadaten vorläufige Filterergebnismengen zusammengestellt werden. Vorläufige Filterergebnisse bieten eine Möglichkeit zum Überbrücken der Wartezeit, die für das Berechnen einer unter Berücksichtigung verschiedener Kriterien optimierte Dokumentenmenge erforderlich ist. Einen beispielhaften Ablauf der Bearbeitung einer Anfrage zeigt das in Abbildung 4.5 dargestellte Sequenzdiagramm.

Das gewählte Schichtenmodell ordnet die Komponenten (Agenten) des Systems jeweils einem Aufgabenbereich zu und definiert eine Managementstrategie für jede Ebene. Jede Schicht wird jeweils durch einen Agenten verwaltet, der die in der Schicht verfügbaren Agenten kennt. Anhand von Daten

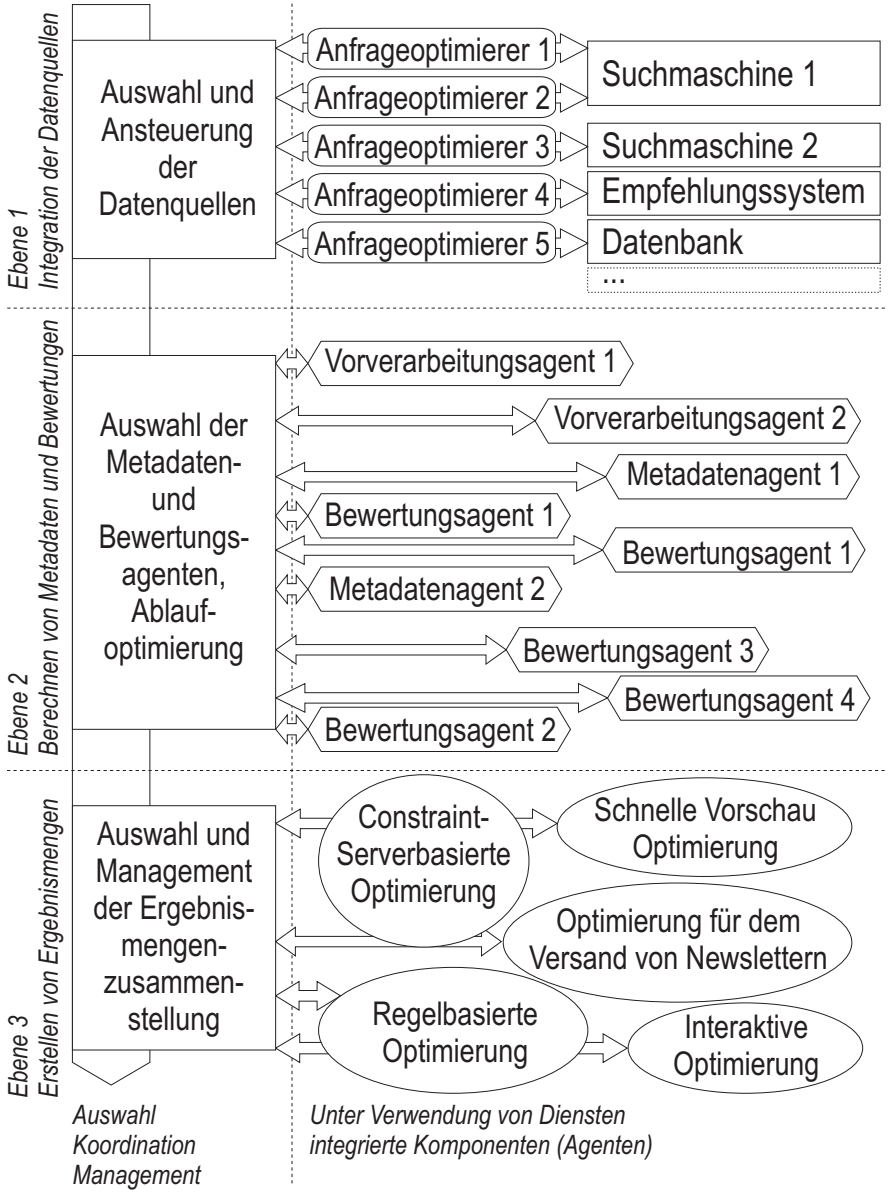


Abbildung 4.4: Die Abbildung zeigt die Architektur für ein personalisiertes Informationssystem. Es wird eine aus drei Ebenen (Extraktion, Bewertungs- und Metadatenberechnung, Ergebnismengenerstellung) bestehende Architektur verwendet, in dem jede Ebene durch eine Managementkomponente verwaltet wird. In Abhängigkeit von der Anfrage und den verfügbaren Ressourcen werden die jeweils am besten geeigneten Komponenten ausgewählt. Die flexible Verknüpfung von Komponenten bietet die Möglichkeit, neue Komponenten zu integrieren und ausgefallene Komponenten zu ersetzen.

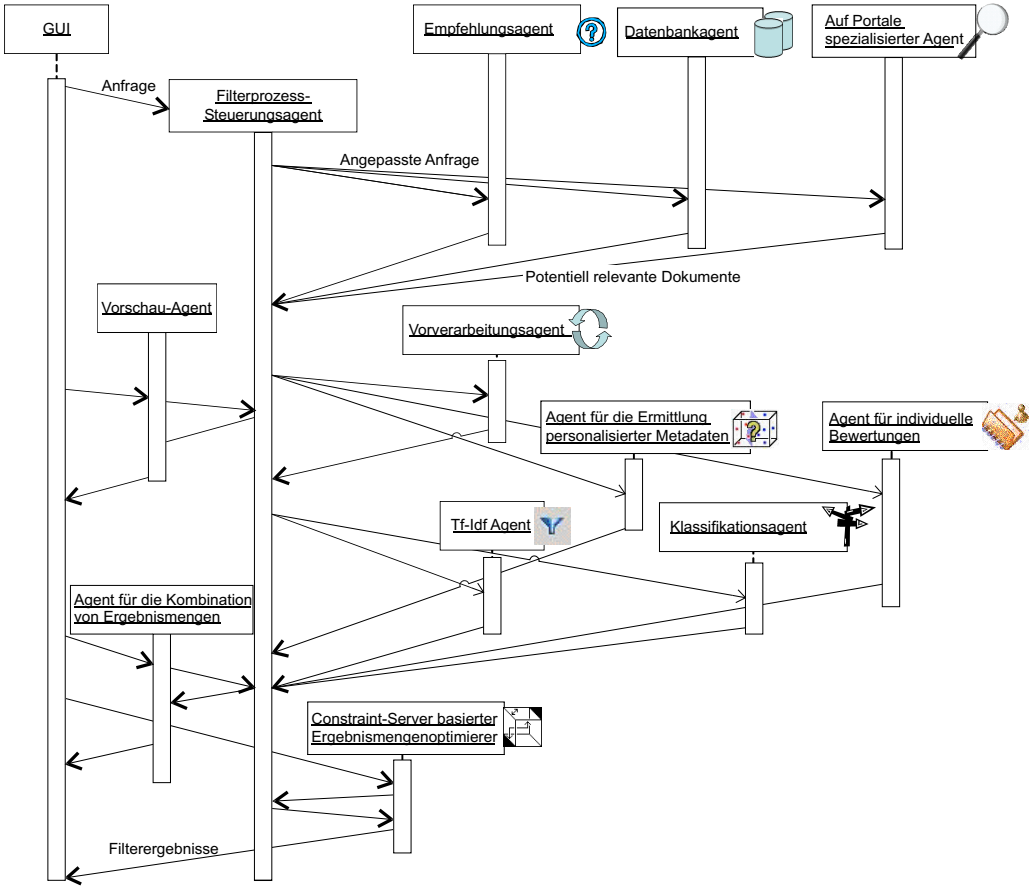


Abbildung 4.5: Die Abbildung zeigt beispielhaft den Ablauf einer Informationssuche. Das Sequenzdiagramm veranschaulicht, wie eine vom Benutzer gestellte Anfrage zunächst durch die Filter-Management-Komponente analysiert, für verschiedene Quellen adaptiert und dann zur parallelen Bearbeitung an die drei Datenquellen weitergeleitet wird. Zu den von den Datenquellen ermittelten Dokumenten werden durch mehrere Agenten nebenläufig Bewertungen und Metadaten berechnet. Um doppelte Verarbeitungsschritte einzusparen, können die Ergebnisse eines Agenten (z. B. des Vorverarbeitungsagenten) von anderen Agenten genutzt werden. Auf Basis der potentiell relevanten Dokumente und der Bewertungen können durch spezialisierte Agenten Ergebnismengen (z. B. Vorschau-Ergebnismenge, mit Constraint-Solver erstellte Ergebnismenge) zusammengestellt werden.

über die Fähigkeiten der Agenten und der Merkmale der Anfrage werden die am besten für die Anfrage geeigneten Agenten ausgewählt.

Die Stärken der entwickelten Architektur liegen in der hohen Flexibilität und der guten Nachvollziehbarkeit der Filterprozesse: In jeder Ebene können neue Entitäten hinzugefügt werden und die jeweils verwendeten Koordinations- und Delegationsstrategien an das aktuelle Szenario angepasst werden. So können vielfältige Filter- und Empfehlungsstrategien untereinander kombiniert und neue Bewertungsalgorithmen integriert werden. Durch die Implementierung der Komponenten als autonome intelligente Entitäten kann eine Verteilung auf mehrere Computer realisiert werden. Dies ermöglicht die Verwendung heterogener Hardware und den Einsatz von Lastverteilungsstrategien [LMK07].

4.4 Abgrenzung zu bestehenden Systemen

Nachdem in den vorangegangenen Abschnitten die Architektur des entworfenen Informationssystems erklärt wurde, werden im Folgenden die Merkmale des Systems erläutert. Dabei werden die Gemeinsamkeiten und Unterschiede mit bestehenden Informationssystemen diskutiert.

Metasuchmaschinen Metasuchmaschinen kombinieren die Ergebnisse von mehreren Suchmaschinen. Die meisten Metasuchmaschinen konzentrieren sich dabei auf die Integration von Standard-Text-Suchmaschinen. Metasuchmaschinen bieten keine intelligenten Strategien zur Ermittlung einheitlicher, personalisierter Bewertungen für Filterergebnisse. Im Allgemeinen konzentrieren sich Metasuchmaschinen auf das Weiterleiten von Suchanfragen an eine große Anzahl von Suchmaschinen und das Sammeln der gelieferten Ergebnisse. Eine intelligente Auswahl von Quellen und ein einheitliches Bewerten von Dokumenten werden nicht vorgenommen. Im Gegensatz dazu ermöglicht es die entwickelte Architektur, verschiedene Arten von Informationsquellen von Informationsquellen anzusteuern und Anfrage an die Besonderheiten der einzelnen Quellen anzupassen. Darüber hinaus bietet die entwickelte Architektur die Möglichkeit, Dokumente aus verschiedenen Quellen einheitlich zu bewerten und die Bewertung auf die individuellen Präferenzen des Benutzers abzustimmen.

Empfehlungssysteme Empfehlungssysteme (wie z. B. MovieLens[1], Amazon[2]) versuchen aus abgegebenen Bewertungen die Interessen eines Benutzers abzuleiten und Bewertungen für bisher noch unbewerteten Objekten vorherzusagen. Die meisten Systeme beschränken sich dabei jedoch auf eine eng begrenzte Menge von zu empfehlenden Objekt-Typen. Die Algorithmen zur Generierung von Empfehlungen sind meist starr definiert und daher nur in geringem Maße erweiterbar und kontextsensitiv. Die entwickelte Architektur ermöglicht es, Empfehlungssysteme als Quellen zu integrieren und die empfehlenden Dokumente bezüglich einer großen Anzahl von Kriterien zu bewerten.

Kombination von Filterstrategien Klassische Suchmaschinen bieten meist nur statische Methoden zur Kombination von Filterverfahren. Populäre Ansätze sind dabei die gewichtete Summe aus der Ähnlichkeit eines Dokumentes mit einer Suchanfrage und der Anzahl der Referenzen auf das Dokument („PageRank"). Bei der Kombination bleiben Suchkontext und individuelle Präferenzen des Benutzers meist unberücksichtigt. Die entwickelte Architektur ermöglicht es, eine Vielzahl von Filterstrategien zu kombinieren. Die Auswahl und Gewichtung der Filterstrategien kann in Abhängigkeit vom Suchszenario und den individuellen Präferenzen des Benutzers erfolgen.

Multi-Agentensysteme: Eine Möglichkeit für die Realisierung flexibler, adaptiver Systeme besteht in der Verwendung von Multi-Agentensystemen. Durch die dienstbasierte Koppelung autonomer Agenten können Funktionalitäten zur Laufzeit erweitert und neu kombiniert werden. Der Einsatz von Multi-Agentenframeworks für die Realisierung von Informationssystemen schafft eine hohe Flexibilität und Offenheit. Dies ermöglicht es, das Informationssystem auf individuelle Benutzerpräferenzen abzustimmen sowie neue Informationsquellen und Bewertungsstrategien zu integrieren. Bereits verfügbare Komponenten (z. B. Suchmaschinen, Empfehlungssysteme, Filterstrategien) können als Agenten in das Informationssystem aufgenommen werden.

4.5 Zusammenfassung

Die vielfältigen Anforderungen an personalisierte Informationssysteme erfordern eine flexible offene Architektur, die es erlaubt, verschiedene Algorithmen

[1] `http://movielens.umn.edu/`, `http://www.grouplens.org/`
[2] `http://www.amazon.com/`

und Verfahren in Abhängigkeit von der Anfrage und dem jeweils betrachteten Szenario zu kombinieren. Es muss die Möglichkeit geben, neue Algorithmen zu integrieren und bestehende Komponenten an die sich verändernden Anforderungen anzupassen. Für ein effektives Management der Systemkomponenten wird eine dreistufige Architektur vorgeschlagen, bei der jede Ebene auf einen Aufgabenbereich spezialisiert ist. Durch die Verwendung einer serviceorientierten Architektur mit autonomen, intelligenten Komponenten (Agenten) können Filteranfragen nebenläufig durch mehrere (ggf. über verschiedene Computer verteilte) Agenten bearbeitet werden, so dass Ressourcen optimal ausgenutzt werden, die für das Filtern erforderliche Zeit minimiert und eine hohe Zuverlässigkeit des Systems erreicht wird.

Die zusätzliche Flexibilität ist mit einem erhöhten Aufwand zur Auswahl und Kombination von Strategien verbunden. In den folgenden Kapiteln werden daher Algorithmen zur Auswahl und Kombination von Bewertungsalgorithmen erläutert, sowie gezeigt, wie Ensemble Learning-Strategien effektiv zur Optimierung von Suchanfragen eingesetzt werden können.

Kapitel 5

Ermitteln relevanter Dokumente und Dokumentenbewertungen

Die Grundlage für das Zusammenstellen personalisierter Filterergebnisse bildet das Ermitteln von potentiell relevanten Dokumenten. Dazu müssen im ersten Schritt geeignete Informationsquellen ausgewählt sowie die Suchanfrage für die Informationsquellen angepasst (und ggf. optimiert) werden.

Das folgende Kapitel gibt zunächst einen Überblick über wichtige Datenquellen und erläutert die jeweiligen Besonderheiten. Es werden Algorithmen zum Optimieren von schlüsselwortbasierten Suchanfragen vorgestellt. Im Anschluss daran werden Verfahren zum Bewerten von Dokumenten und zum Ermitteln von Metadaten erläutert.

5.1 Auswahl von Informationsquellen

Ein Informationssystem sollte unterschiedliche Informationsquellen integrieren, um möglichst viele für den Benutzer relevante Dokumente bei der Suche zu berücksichtigen. Die wichtigsten Typen der für die Suche relevanten Datenquellen und deren Besonderheiten werden im Folgenden erläutert.

Internet-Suchmaschinen Internet-Suchmaschinen (z. B. GOOGLE) bzw. Metasuchmaschinen (z. B. METAGER) ermitteln zu schlüsselwortbasierten Suchanfragen relevante Dokumente. Suchmaschinen stellen dabei ergänzend zu der Adresse eines Dokumentes eine kurze Erklärung zur Relevanz des Dokumentes (z. B. ein „Snippet“, Position in der Ergebnismenge) bereit. Die Güte der von einer Suchmaschine ermittelten Dokumente hängt stark von der verwendeten Anfrage ab. Aus diesem Grund ist es sinnvoll, Algorithmen zum Optimieren von Suchanfragen einzusetzen und den Benutzer beim Ermitteln

von geeigneten Suchbegriffen zu unterstützen. Bei der Verwendung von Suchmaschinen muss berücksichtigt werden, dass die ermittelte Dokumentenmenge oftmals irrelevante Dokumente (z. B. „Spam“) enthält und ein Teil der von einer Suchmaschine ermittelten Dokumente im Allgemeinen nicht mehr verfügbar ist.

Verzeichnisdienste Verzeichnisdienste (wie z. B. DMoz[1]) ordnen Internetseiten in Kategorien ein. Da die Zuordnung von Webseiten zu Kategorien durch menschliche Redakteure vorgenommen wird, weisen die von einem Verzeichnisdienst zu einer Anfrage ermittelten Dokumente im Allgemeinen eine geprüfte Mindestqualität auf und liefern zur ausgewählten Kategorie passende Daten. Ein Problem bei der Verwendung von Verzeichnisdiensten besteht darin, dass Verzeichnisse im Allgemeinen veraltete oder nicht mehr verfügbare Einträge enthalten. Bei der Integration von Verzeichnisdiensten in ein Informationssystem sollten daher die ermittelten Dokumente auf ihre Verfügbarkeit geprüft werden.

Internet-Portale Für viele Themenbereiche existieren spezialisierte Informationsportale. Diese bieten aktuelle, auf ein begrenztes Themengebiet abgestimmte Informationen. Portale sind daher besonders gut geeignet, wenn neue Entwicklungen innerhalb eines Themenbereiches verfolgt werden sollen oder aktuelle Informationen aus einer Domäne benötigt werden. Bei der Integration von Portalen in ein Informationssystem muss sichergestellt werden, dass für eine Anfrage jeweils die am besten geeigneten Portale ausgewählt werden.

Nachschlagewerke Um grundlegende Informationen zu Begriffen oder Abkürzungen zu ermitteln, können Nachschlagewerke (z. B. Wörterbücher, Lexika) eine wichtige Quelle sein. Online verfügbare Nachschlagewerke (wie z. B. Wikipedia[2]) bieten oftmals als Ergänzung zu Begriffserklärungen (ähnlich wie Verzeichnisdienste) Referenzen auf weiterführende Dokumente und relevante Informationsquellen.

Empfehlungssysteme Empfehlungssysteme (z. B. Netflix [BL07], GroupLens [RIS+94]) liefern zu einen Interessenprofil potentiell relevante Dokumente. Bei der Integration von Empfehlungssystemen in ein Informationssystem ist zu berücksichtigen, dass einige Empfehlungssysteme als Anfrage eine Menge von Bewertungen erwarten, um Methoden des

[1] `http://www.dmoz.org/`
[2] `http://www.wikipedia.org/`

kollaborativen Filterns anwenden zu können. Zudem muss beachtet werden, dass die von Empfehlungssystemen ermittelten Bewertungen nicht direkt mit inhaltsbasierten Bewertungen verglichen werden können. Informationssysteme müssen daher verschiedene Arten von Benutzerprofilen (d. h. auf Suchbegriffen basierende Profile und auf Bewertungen basierende Profile) verwalten können und in der Lage sein, aus heterogen Quellen stammende Dokumente einheitlich zu bewerten.

Datenbanken Zum Verwalten großer (einheitlich strukturierter) Datenmengen, werden Datenbanken eingesetzt. Im Rahmen eines Informationssystems sind Personen- und Publikationsdatenbank (z. B. für Bücher, Zeitschriften, wissenschaftliche Veröffentlichungen) von besonderer Bedeutung. Bei der Integration von Datenbanken in Informationssysteme muss berücksichtigt werden, dass Datenbanken oft nur eine einfache (mengenbasierte) Suchfunktion anbieten. Um komplexe Suchanfragen ausführen zu können, muss im Allgemeinen die Suchfunktion an die Anforderungen der Anwendung angepasst werden (z. B. durch das Erstellung eines Volltext-Index).

Datendienste im Rahmen von Rundfunk und Fernsehen Fernseh- und Rundfunksender stellen oft aktuelle Daten zur Verfügung, wie z. B. Programminformationen in Form eines elektronischen Programmführers („EPG“). Diese Dienste können als Datenquelle in einem Informationssystem genutzt werden. Zu berücksichtigen ist, dass diese Datenquellen keine Suchfunktion anbieten, sondern dass ein Informationssystem eine geeignete Komponente zum Filtern der Daten bereitstellen muss.

Die Integration vielfältiger Datenquellen in ein Informationssystem ermöglicht es, in Abhängigkeit von der Anfrage und den Benutzerpräferenzen die am besten geeigneten Quellen auszuwählen. Der Benutzer muss sich nicht mit den Besonderheiten der Datenquellen auskennen und spart Zeit, da das beschwerliche Abfragen mehrerer Quellen durch das Informationssystem automatisch vorgenommen wird. Darüber hinaus analysiert das Informationssystem die von den Datenquellen bereitgestellten Bewertungen, ergänzt diese durch Metadaten und Bewertungen, so dass unabhängig von der Quelle die Relevanz eines Dokumentes zuverlässig beurteilt werden kann.

Die Auswahl der für eine Anfrage berücksichtigten Datenquellen kann entweder durch den Benutzer oder automatisch durch das Informationssystem erfolgen. Als Ausgangspunkt für die automatisierte Auswahl von Datenquellen können expertendefinierte Regelmengen eingesetzt werden, die durch

die Auswertung des Benutzerverhaltens (z. B. die abgegebenen Bewertungen) an die individuellen Präferenzen des Benutzers angepasst werden. Kriterien, die für die Auswahl einer Quelle herangezogen werden, sind beispielsweise die Antwortzeit der Datenquelle, der Themenbereich zu dem die Datenquelle Informationen zur Verfügung stellt, die bereits vom Benutzer vergebenen Bewertungen und die Aktualität der durch die Datenquelle bereitgestellten Dokumente.

Regeln für die Bestimmung geeigneter Informationsquellen können beispielsweise sein:

- Ist das Themengebiet bekannt, in dem nach Dokumenten gesucht werden soll, werden als Quellen die auf das gewünschte Themengebiet spezialisierten Portale bevorzugt. Als Ergänzung werden Dokumente von Suchmaschinen und Metasuchmaschinen abgefragt.

- Startet der Benutzer die Suche in einem neuen Themengebiet, so werden bei der Auswahl von Datenquellen Nachschlagewerke und Verzeichnisdienste bevorzugt. Die Dokumentenmenge wird um aktuelle Dokumente von Portalen ergänzt.

- Arbeitet ein Benutzer bereits längere Zeit in einem Bereich, so werden vor allem Dokumentenquellen berücksichtigt, die aktuelle Daten bereitstellen (z. B. Nachrichtenportale). Dabei sind solche Quellen zu bevorzugen, die der Benutzer in der Vergangenheit positiv bewertet und häufig genutzt hat.

- Verwendet der Benutzer das Informationssystem interaktiv, so werden Datenquellen mit sehr langen Antwortzeiten ausgeschlossen.

Das Auswählen von Datenquellen für eine Anfrage hat einen starken Einfluss auf die Güte der von einem Informationssystem ermittelten Filterergebnisse. Es muss sichergestellt werden, dass jeweils alle für die Anfrage relevanten Quellen einbezogen werden (um nicht relevante Dokumente durch eine ungeeignete Auswahl von Datenquellen auszuschließen). Zugleich sollten aber die Datenquellen unberücksichtigt bleiben, die keine zur Anfrage passenden Dokumente bereitstellen, um so Systemressourcen zu sparen und das Bearbeiten von Anfragen zu beschleunigen.

5.2 Optimieren von Anfragen

Einen entscheidenden Einfluss auf die Dokumentenmenge, die eine Datenquelle zu einer Anfrage des Benutzers liefert, hat die verwendete Suchanfrage. Das Informationssystem sollte daher den Benutzer beim Erstellen von

Suchanfragen unterstützen, indem es auf schlecht formulierte Anfragen hinweist und Tipps zur Optimierung gibt. Bei der Optimierung und Anpassung von Suchanfragen werden vor allem drei unterschiedliche Zielsetzungen verfolgt:

- Die Anpassung der Anfragen an die Besonderheiten der jeweiligen Datenquelle.
- Das Erkennen und Beheben von Fehlern und Widersprüchen in Suchanfragen.
- Das Verfeinern bzw. Erweitern von Anfragen.

Die für die verschiedenen Ziele eingesetzten Algorithmen und Methoden werden im Folgenden diskutiert.

Anpassen an die Besonderheiten der Datenquelle

Im Bereich der Suchmaschinen und Dokumentenquellen kommen unterschiedliche Verfahren und Sprachen zur Definition von Suchanfragen zum Einsatz. Um verschiedene Datenquellen zu integrieren, muss daher die Anfrage des Benutzers an die Besonderheiten der jeweiligen Quelle angepasst werden. Dabei müssen beispielsweise die Syntax zur booleschen Verknüpfung von Suchbegriffen (z. B. durch „AND“, „UND“ oder „+“), die von der Datenquelle unterstützten Feature-Schlüsselwörter (z. B. „site:“ oder „filetype:“) und die jeweils zulässige Größe von Suchanfragen („maximale Anfragelänge“) beachtet werden. Zusätzlich ist zu berücksichtigen, dass nicht jeder Anfragetyp von allen Datenquellen unterstützt wird. In Abhängigkeit von der Anfrage kann es daher sinnvoll sein, eine geeignete Ersetzung von Nebenbedingungen vorzunehmen: Ist beispielsweise in einer Anfrage das gewünschte Dateiformat spezifiziert, so kann oft mit Wissen darüber, welche Datenquelle welches Dateiformat unterstützt, das gewünschte Dateiformat durch eine geeignete Quellenauswahl erreicht werden.

Beheben von Fehlern und Unklarheiten

Fehler und Widersprüche in Suchanfragen können zu unerwarteten, nur schlecht zu den Interessen des Benutzers passenden Suchergebnissen führen. Ein Informationssystem sollte daher Verfahren zum Erkennen von Fehlern und zum Optimieren von Anfragen bieten.

Schreibfehler

Bei der Eingabe von Suchbegriffen können verschiedene Arten von Fehlern auftreten, wie beispielsweise Tippfehler (z. B. Vertauschen von Buchstaben), fehlerhaft kodierte Zeichen (z. B. weil Daten aus einer Fremdanwendung kopiert wurden), Rechtschreibfehler oder veraltete Schreibweisen für einen Suchbegriff. Zum Erkennen dieser Fehler können Rechtschreibkorrektur-Algorithmen [Whi04, Knu73, Lev66] eingesetzt werden, die die Eingaben unter Verwendung eines Abstandsmaßes gegen eine Menge bekannter Begriffe prüfen (z. B. unter Verwendung von Sprachwörterbüchern).

Entdecken von logischen Fehlern

Suchmaschinen unterstützen im Allgemeinen die Verknüpfung von Suchbegriffen durch boolesche Operatoren. Dabei können verschiedene Fehler auftreten, wie z. B. fehlerhafte Klammerung, Tautologien (`a OR NOT a`) und redundante Einträge (`a AND a`). Zur Behandeln dieser Fälle kann entweder der Benutzer auf die Fehler hingewiesen werden oder versucht werden, durch das Entfernen von fehlerhaften Ausdrücken eine gültige Suchanfrage abzuleiten.

Homographe Begriffe

Bei der Verwendung schlüsselwortbasierter Suchanfragen müssen Wörter, die unterschiedliche Bedeutungen besitzen können (Homographen), besonders berücksichtigt werden. Homographe Begriffe können dazu führen, dass Dokumente als relevant eingestuft werden, die nicht zu den Interessen des Benutzers passen, weil sie zwar den gewünschten Suchbegriff enthalten, nicht aber zum gewünschten Themengebiet gehören. In diesem Fall sollte der Benutzer auf die Mehrdeutigkeit des Suchbegriffes hingewiesen werden. Verfahren zum Verfeinern und Optimieren von Suchanfragen, mit denen auch homographe Begriffe geeignet behandelt werden können, werden im folgenden Abschnitt diskutiert.

Anfrageoptimierungsstrategien

Ziel der Anfrageoptimierung ist es, eine Anfrage so zu modifizieren, dass damit besser zu den Interessen des Benutzers passende Ergebnisse ermittelt werden können. Dies kann beispielsweise erfolgen, indem eine ungenau definierte Anfrage auf einen Teilbereich eingeschränkt wird oder eine zu speziell definierte Anfrage auf einen größeren Themenbereich erweitert wird. Das Optimieren von Suchanfragen kann sowohl vollautomatisch durch das Informationssystem als auch iterativ im Dialog mit dem Benutzer erfolgen.

Erweitern der Anfrage durch Synonyme

Da bei der schlüsselwortbasierten Suche Dokumente nur dann als relevant eingestuft werden, wenn sie den gewünschten Suchbegriff enthalten, werden Dokumente, die lediglich Synonyme des Suchbegriffes enthalten, aber sehr gut zum Interessenbereich des Nutzers passen, als irrelevant eingestuft. Um Dokumente, die Synonyme des definierten Suchbegriffes enthalten als Ergebnisse zu erhalten, erweitert man die Anfrage um geeignete Synonyme. Eine Übersicht über die Typen von Synonymen gibt die Tabelle 5.1.

Ein möglicher Ansatz zum Berücksichtigen von Synonymen in Suchanfragen besteht in der Verwendung von Synonymwörterbüchern. Synonymwörterbücher stellen zu einem Begriff eine Menge von verwandten Wörtern zur Verfügung. Zusätzlich stellen diese Wörterbücher meist Informationen zur Verfügung, um welche Art von Synonym es sich handelt und in welchen Themenbereich das Synonym verwendet wird. Bei der Adaption von Informationsanfragen unter Verwendung von Synonymwörterbüchern müssen jeweils aus der Gesamtmenge der Synonyme die Wörter ermittelt werden, die zum Themenbereich der Anfrage passen und in Abhängigkeit von Suchkontext die Anfrage thematisch einschränken oder generalisieren.

Umfangreiche Synonymwörterbücher, die für die Optimierung von Suchanfragen herangezogen werden können, sind beispielsweise WordNet[3] (englisch) und GermaNet[4] (deutsch). Diese stellen neben Informationen zur Wortart von Begriffen Daten über die Beziehungen zwischen Begriffen zur Verfügung. Diese Informationen können genutzt werden, um eine Anfrage zu verallgemeinern (z. B. durch das Erweitern einer Anfrage durch Hyperonyme) bzw. auf einen Themenbereich einzuschränken (z. B. durch das Ergänzen von Hyponymen).

Unter Verwendung von Synonymwörterbüchern können für häufig verwendete Begriffe Synonyme ermittelt werden. Der Nachteil von Nachschlagewerken besteht darin, dass Spezial-Themengebiete nur unzureichend berücksichtigt und aktuelle Begriffe nur verzögert aufgenommen werden.

Erweitern von Anfragen durch Kollokationsanalyse

Ein Ansatz zum dynamischen Ermitteln von semantisch verwandten Begriffen besteht in der Durchführung einer Kollokationsanalyse. Man analysiert dabei für eine zu einem Themenbereich gehörende Menge von Dokumenten, welche Begriffe häufig zusammen auftreten. Die anhand der Kollokationsanalyse ermittelten miteinander semantisch verwandten Begriffe können (ähnlich

[3] http://wordnet.princeton.edu/

[4] http://www.sfs.uni-tuebingen.de/lsd/

Wortart	Art des Synonyms	Erklärung
Substantiv	Hyperonym (Oberbegriff)	A ein Hyperonym von B, wenn jedes B auch ein A ist (Verallgemeinerung).
	Hyponym (Unterbegriff)	A ein Hyponym von B, wenn jedes A auch ein B ist (Spezialisierung).
	Gemeinsamer Oberbegriff	Die Substantive A und B besitzen einen gemeinsamen Oberbegriff, wenn A und B ein gemeinsames Hyperonym besitzen.
	Holonym (Umkehrung von Meronym)	A ist ein Holonym von B, wenn B ein Teil von A ist.
	Meronym (Umkehrung von Holonym)	A ist ein Meronym von B, wenn A ein Teil von B ist.
Verben	Hyperonym (Oberbegriff)	Ein Verb A ist ein Hyperonym des Verbs B, wenn die Aktivität B eine Spezialisierung (a kind of) von A ist.
	Troponym	Das Verb A ist ein Troponym des Verbs B, wenn das Verb B eine Ausprägung von A ist.
	Erforderung	Das Verb A ist ein Teil von B, wenn man B ausführen muss, um auch A auszuführen (z. B. schnarchen – schlafen).
	Gemeinsamer Oberbegriff	Zwei Verben A und B besitzen ein gemeinsames Verb, das ein Oberbegriff von A und B ist.
Adjektive	Verwandte Substantive, Partizipien	Von Verben abgeleitete Partizipien können adjektivisch verwendet werden.
Adverbien	Zugehörende Adjektive	Adverbien leiten sich oft von Adjektiven ab. Adverbien und Adjektive erfüllen unterschiedliche Funktionen im Satz, besitzen aber eine ähnliche Semantik.

Tabelle 5.1: Übersicht über die Arten von Synonymen

wie Synonyme) zum Erweitern von Suchanfragen eingesetzt werden. Der Vorteil des Ansatzes liegt darin, dass die Kollokationsanalyse automatisiert auf Basis von Dokumentensammlungen durchgeführt werden kann. Daher können bei diesem Verfahren aktuelle Entwicklungen schneller berücksichtigt und spezielle Themenbereiche einbezogen werden. Das Verfahren kann zudem unter Verwendung semantischer Daten erweitert werden, indem man in Dokumenten Abkürzungen und Entitäten (z. B. Personen) identifiziert und die semantische Nähe zu den Anfragebegriffen ermittelt.

Erweitern von Anfragen durch Domänen-Schlüsselwörter

Bei der Verwendung von auf Schlüsselwörtern basierenden Suchanfragen kann das Problem auftreten, dass ein Suchbegriff in mehreren Themenbereichen verwendet wird. Dies kann dazu führen, dass Suchmaschinen Dokumente aus Bereichen als relevant einstufen, die nicht zum Interessenprofil des Benutzers passen. Um Dokumente auszuschließen, die zu für den Benutzer irrelevanten Themengebieten gehören, erweitert man die Suchanfrage um Begriffe, die die Anfrage auf den gewünschten Themenbereich eingrenzen. Man greift dabei im Allgemeinen auf Domänenwissen zurück, welches beispielsweise durch Verzeichnisdienste (z. B. DMOZ) oder Nachschlagewerke (z. B. WORDNET) verfügbar ist. Zum Erkennen von Mehrdeutigkeiten können auch Enzyklopädien (Sachwörterbücher) herangezogen werden, die verschiedene Bedeutungen eines Wortes erläutern und zusätzlich eine kurze Beschreibung des Begriffes sowie Referenzen auf verwandte Konzepte bereitstellen.

Kollaborative Anfrageoptimierung

Bei der kollaborativen Anfrageoptimierung werden aus der Analyse der Ähnlichkeiten zwischen Benutzern Vorschläge für die Verbesserung von Suchanfragen abgeleitet. Man folgt dabei dem Ansatz, dass für Benutzer mit ähnlichen Interessen auch ähnliche Suchbegriffe relevant sind. Die Ähnlichkeit zwischen Benutzern kann dabei anhand von Anfragen, Bewertungen oder dem Verhalten bestimmt werden.

Bei der kollaborativen Anfrageoptimierung muss sichergestellt werden, dass die zum Generieren von Vorschlägen verwendeten Benutzerdaten anonymisiert werden und sich aus den Vorschlägen keine Rückschlüsse auf personenbezogene Daten ziehen lassen. Des Weiteren besteht bei der kollaborativen Anfrageoptimierung das Risiko, dass fehlerhaft geschriebene Begriffe zum Adaptieren einer Anfrage vorgeschlagen werden und dass Benutzer versuchen, anderen Benutzer unpassende Begriffe (z. B. Werbung, „Spam“) vor-

zuschlagen. Verfahren zur kollaborativen Anfrageoptimierung sollten daher stets mit alternativen Anfrageoptimierungsverfahren kombiniert werden.

Eine Erweiterung zum Berücksichtigen von Anfragen von Benutzern mit ähnlichen Interessen, besteht in der Integration von **Social Tagging**-Diensten. Social Tagging-Dienste ermöglichen es Benutzern, Internet-Adressen zusammen mit einer Menge von Schlüsselwörtern zu speichern. Auf Basis der definierten Begriffe und der Verknüpfungen kann ein semantisches Netz aufgebaut werden, das als Grundlage für die Ermittlung häufig zusammen auftretender Begriffe verwendet werden kann [SGMB08].

Durch die hohe Akzeptanz von Social Tagging-Diensten stellen diese Dienste umfangreiche Datensammlungen bereit, durch die aktuelle Trends erfasst werden können. Im Vergleich zur Extraktion von Begriffen (wie sie bei der Kollokationsanalyse auf Basis von Texten durchgeführt wird) haben Social Tagging-Dienste den Vorteil, dass die Begriffe durch Benutzer definiert werden und gut für die schlüsselwortbasierte Suche geeignet sind [WZB08].

Anfrageoptimieren auf Basis von Benutzerbewertungen

Hat ein Benutzer Dokumente bewertet, so lassen sich damit Rückschlüsse auf die Benutzerinteressen ziehen und Vorschläge für die Optimierung von Anfragen ermitteln. Dabei können sowohl inhaltsbasierte Methoden (z. B. die Extraktion von charakteristischen Begriffen aus bewerteten Dokumenten) als auch kollaborative Verfahren (z. B. Ermitteln von Begriffen, die Benutzer mit ähnlichen Bewertungen benutzen) eingesetzt werden. In Abhängigkeit davon, ob der Benutzer Dokumente positiv oder negativ bewertet hat, können die ermittelten Begriffe in der Anfrage eine positive oder eine negative Gewichtung erhalten.

Hat der Benutzer Dokumente positiv als auch Dokumente negativ bewertet, so können unter Verwendung von Klassifikationsalgorithmen Merkmale ermittelt werden, die besonders gut für eine Trennung von guten und schlechten Dokumenten geeignet sind. Die ermittelten Merkmale werden zum Optimieren von Anfragen verwendet. Dabei können beispielsweise Entscheidungsbäume (wie z. B. C4.5 [Qui93]) oder der ROCCHIO-Algorithmus [Joa97] erfolgreich eingesetzt werden. Das Ableiten von Profiloptimierungsvorschlägen anhand von Bewertungen hat den Vorteil, dass detaillierte Erklärungen zu Vorschlägen bereitgestellt werden können, was die Nachvollziehbarkeit und die Akzeptanz der generierten Vorschläge erhöht. Eine detaillierte Beschreibung der Anfrageoptimierungsstrategien erfolgt im Rahmen der Vorstellung der implementierten Informationssysteme (Kapitel 8).

5.3 Kombinieren von Anfrageoptimierungsverfahren

Beim Auswählen von Vorschlägen zur Anfrageoptimierung müssen (analog zur Bestimmung von Dokumenten zu einer Anfrage) verschiedene Kriterien berücksichtigt werden. Man kombiniert daher mehrere Algorithmen zur Anfrageoptimierung, um die Stärken der jeweiligen Verfahren zu nutzen und die Schwächen der einzelnen Verfahren zu überwinden. Die Architektur der Komponente zum Ermitteln von Anfrageoptimierungsvorschlägen (Abbildung 5.1) orientiert sich an der zum Filtern von Dokumenten entwickelten Architektur.

Zusammenfassen von Anfrageoptimierungsvorschlägen

Um die von den verschiedenen Anfrageoptimierungsalgorithmen ermittelten Vorschläge zu kombinieren, werden Verfahren der Schnittmengenanalyse sowie des Votings eingesetzt.

Bei der Schnittmengenanalyse ermittelt man, inwiefern die von den Agenten zur Anfrageoptimierung ermittelten Vorschläge untereinander Übereinstimmungen aufweisen. Vorschläge, die sowohl von kollaborativ als auch von inhaltsbasiert arbeitenden Algorithmen ermittelt wurden, sind für die Anfrageoptimierung besonders geeignet, da diese Vorschläge unterschiedliche Aspekte berücksichtigen. Um bei der Schnittmengenanalyse die Güte von Vorschlägen berechnen zu können, sollten die Agenten zur Anfrageoptimierung jeweils mehrere Vorschläge liefern sowie die Güte eines Vorschlages beschreiben. Um bei der Schnittmengen verschiedene Schreibweisen von Begriffsvorschlägen zu erkennen, können beim Analysieren von Vorschlägen, die von verschiedenen Agenten stammen, Methoden des unscharfen Abgleichs (z. B. LEVENSHTEIN-Abstand, Wortstammanalyse) eingesetzt werden.

Da die von unterschiedlichen Algorithmen (Agenten) ermittelten Vorschläge zur Anfrageoptimierung nicht immer Überschneidungen aufweisen, können die Optimierungsvorschläge auch durch die Verwendung von Voting-Algorithmen kombiniert werden. Es werden dazu unter Verwendung unterschiedlicher Algorithmen Begriffe für die Anfrageoptimierung generiert. Die von einem Agenten ermittelten Vorschläge werden jeweils von alternativen Agenten bewertet. Auf Basis dieser Bewertungen werden dann die Vorschläge ausgewählt, die für die Optimierung verwendet werden.

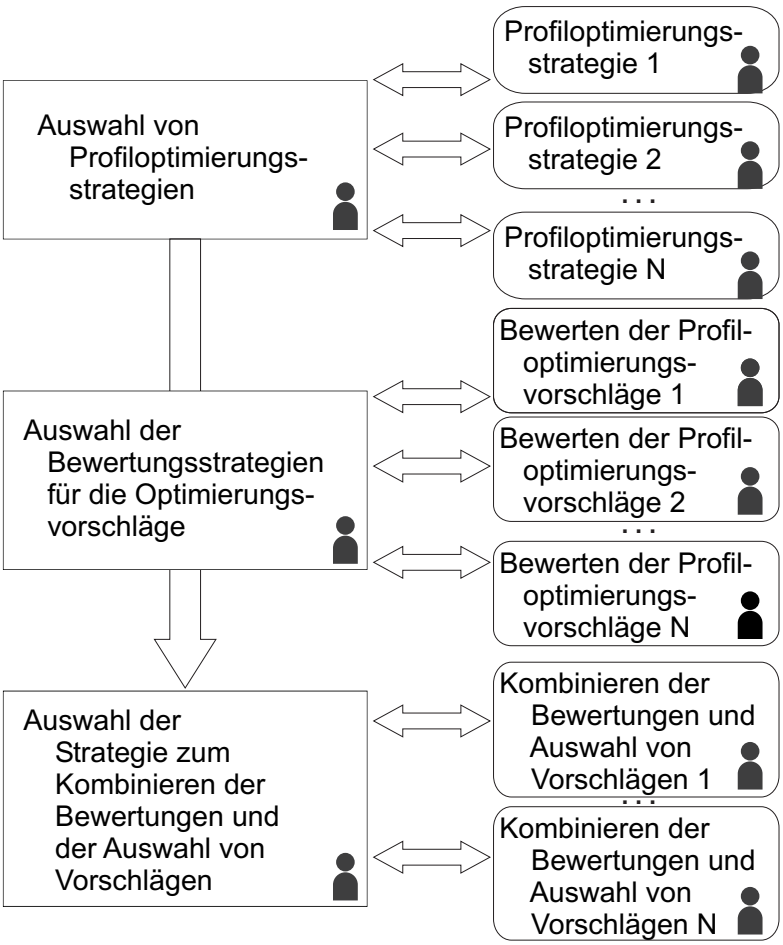


Abbildung 5.1: Die Abbildung zeigt die Architektur der Anfrageoptimierungskomponente. Es wird eine dreischichtige Struktur verwendet, die sich aus einer Schicht zur Ermittlung von Vorschlägen, einer Schicht zur Bewertung der Vorschläge und der Schicht zur Auswahl der am besten geeigneten Profiloptimierungsvorschläge zusammensetzt. Die einzelnen Ebenen sind jeweils so realisiert, dass neue Strategien (Agenten) hinzugefügt bzw. bestehende Strategien ausgetauscht werden können.

Beispiele für die Kombination von Optimierungsalgorithmen

Anhand der Schwächen von einzelnen Anfrageoptimierungsalgorithmen wird gezeigt, wie durch eine geeignete Kombination alternativer Strategien, die Qualität der Anfrageoptimierungsvorschläge verbessert werden kann.

- Das Hauptproblem bei der Verwendung von Wörterbüchern zur Fehlerkorrektur ist, dass aktuelle oder fachspezifische Begriffe oft von Wörterbüchern nicht erfasst werden und daher vom Informationssystem möglicherweise als fehlerhaft eingestuft werden. Um die Exaktheit der Bewertung von aktuellen Begriffen zu erhöhen, werden Vorschläge zur Profiloptimierung, die in Wörterbüchern nicht auftreten, mit Hilfe von Web 2.0-Diensten (z. B. Social Tagging-Diensten) oder mit Hilfe von Suchmaschinen („Wie oft wird der betreffende Begriff in einer Domäne verwendet?“) überprüft.

- Bei der Verwendung von Synonymwörtern besteht eine Schwierigkeit darin zu erkennen, welche der Synonyme zu der für den Benutzer interessanten Domäne gehören. Eine Analyse, welche Wörter in welchem Themenbereich relevant sind, kann anhand von Lexika durchgeführt werden. Alternativ kann eine Überprüfung auch unter Anwendung von kollaborativen Verfahren vorgenommen werden. Die Idee dabei ist, dass insbesondere die Synonyme für den Benutzer relevant sind, die in Dokumenten auftreten, die Benutzer mit ähnlichen Interessen positiv bewertet haben.

- Bei der Analyse von Dokumenten auf Basis einer Tf – Idf-Statistik (Termfrequenz-Analyse) werden in einem Dokument häufig auftretende Rechtschreibfehler und bei gemischt sprachlichen Texten Stoppwörter als besonders relevant eingestuft. Um diese Schwachstelle zu überwinden, sollten die vorgeschlagenen Begriffe anhand von Wörterbüchern und Stoppwortlisten überprüft werden, um wenig hilfreiche Vorschläge zu erkennen und zu eliminieren.

- Bei der Verwendung von kollaborativen Verfahren besteht die Gefahr, dass für die Suche ungeeignete Begriffe aus dem Profil eines Benutzers anderen Benutzern vorgeschlagen werden. Um dieses Problem zu lösen, sollten die auf Basis kollaborativer Verfahren ermittelten Begriffe mit Hilfe von Wörterbüchern auf ihre korrekte Schreibung überprüft werden. Eine Rechtschreibkontrolle ist insbesondere bei geringer Anzahl von Benutzern notwendig, da in diesem Fall die Vorschläge anhand von

wenigen Ähnlichkeitsmerkmalen ermittelt werden. Um das Vorschlagen von domänenfremden Begriffen zu verhindern, sollten die auf Basis kollaborativer Verfahren ermittelten Begriffe zusätzlich unter Verwendung der Verfahren „Begriffe von Suchmaschinen“ oder „Wörter aus Enzyklopädien“ überprüft werden.

Die Beispiele zeigen, dass durch eine geeignete Verknüpfung von Profiloptimierungsstrategien die wichtigsten Fehler erkannt und behoben werden können.

5.4 Ermitteln von Metadaten und Bewertungen für Dokumente

Um die von Agenten der Extraktionsschicht als potentiell relevant ermittelten Dokumente einheitlich beurteilen zu können, wird die Güte bzw. die Relevanz der Dokumente durch eine Menge von Bewertungsagenten berechnet. Da die Menge von potentiell relevanten Dokumenten (für die zusätzliche Bewertungen ermittelt werden) im Allgemeinen klein ist im Vergleich zu den Dokumentensammlungen, die von den Datenquellen berücksichtigt werden, können in dieser Stufe des Filterprozesses komplexe, personalisierte Bewertungen berechnet werden. Dies ermöglicht es, die Filterergebnismengen auf die individuellen Präferenzen des Benutzers und das jeweilige Szenario abzustimmen.

Zusätzlich zu Bewertungen können in dieser Schicht des Informationssystems auch die Metadaten berechnet werden, die für eine optimale Präsentation der Dokumente benötigt werden (z. B. Zusammenfassungen).

Arten von Bewertungsstrategien

Für eine umfassende Beurteilung von Dokumenten müssen vielfältige Kriterien berücksichtigt werden. Die wichtigsten Strategien zum Bewerten von Dokumenten werden im Folgenden erläutert.

Von der Anfrage abhängige Merkmale

Eine wichtige Gruppe von Kriterien zur Bewertung der Relevanz eines Dokumentes bilden die anfragespezifischen Kriterien. Man berechnet dabei die Ähnlichkeit zwischen der Anfrage und dem Dokument. Bei Verwendung von schlüsselwortbasierten Anfragen analysiert man dazu die Häufigkeit des Auftretens der Anfragebegriffe in den zu bewertenden Dokumenten. Die Häufig-

keit des Auftretens wird mit dem Teil des Dokumentes (z. B. Titel, Zusammenfassung, Hauptteil) gewichtet, in der ein Suchbegriff auftritt. Um zu bestimmen, ob ein Suchbegriff mit einer Zeichenkette in einem Dokument übereinstimmt, können verschiedene Techniken des unscharfen Matchings eingesetzt werden, z. B. Reduzierung von Wörtern auf ihre Stammform [Por97], Vergleichen der übereinstimmenden Zeichenfolgen („n-Gram-Matching auf Zeichenebene"), Übereinstimmung von Wortfolgen („n-Gram-Matching auf Wortebene") oder das Berechnen der LEVENSTHEIN-Distanz [Lev66]. Unter Verwendung der Übereinstimmungsmaße lassen sich verschiedene Relevanzbewertungsstrategien definieren:

- **Termfrequenz**: Es wird ermittelt, wie oft die Anfragebegriffe im Dokument auftreten [BYRN99].

- **Tf – Idf:** Es wird das Verhältnis der Termfrequenz im Dokument zur Termfrequenz der Anfragebegriffe in der Dokumentensammlung ermittelt [BYRN99].

- **Semantische Übereinstimmungen:** Stehen zu den Dokumenten semantische Annotationen zur Verfügung, so können zum Berechnen der Relevanz eines Dokumentes Verfahren des semantischen (graphbasierten) Matching eingesetzt werden [GS03].

Stehen für ein Dokument Kategorie-Schlüsselwörter oder Schlagwörter zur Verfügung („index terms"), so wird die Übereinstimmung von Anfragebegriffen mit den zum Dokument definierten Schlagwörtern geprüft.

Dokumentenspezifische Kriterien

Neben den anfragespezifischen Kriterien können dokumentenspezifische Kriterien berücksichtigt werden. Merkmale, die als Grundlage für die Bewertung eines Dokumentes dienen können, sind z. B. die Autoren, der Verlag, das Dateiformat und der Umfang des Dokumentes.

Algorithmen, die auf Basis dokumentenspezifischer Merkmale Bewertungen berechnen, werden im Folgenden erläutert.

PageRank Es wird analysiert, welche Dokumente auf das zu bewertende Dokument verweisen. Man geht dabei davon aus, dass es auf hochqualitative Dokumente eine große Zahl von Referenzen gibt, wobei insbesondere die Verweise berücksichtigt werden, die von populären Dokumenten stammen [PBMW98, Ber05].

Klassifizieren des Anbieters des Dokumentes Für im Internet bereitgestellte Dokumente wird unter Verwendung von Datenbanken und Heuristiken ermittelt, welche Organisation bzw. welcher Autor das Dokument veröffentlicht hat. Dabei wird Dokumenten eine hohe Qualität zugeordnet, die von vertrauenswürdigen Quellen (z. B. anerkannten Hochschulen oder Wissenschaftsverlagen) stammen.

Domäne des Dokumentes Um den Themenbereich zu ermitteln, dem ein Dokument zuzuordnen ist, wird unter Verwendung von Verzeichnisdiensten und Klassifikatoren eine Menge von Themenbereichen ermittelt, in die ein zu bewertendes Dokument eingeordnet werden kann. Die Klassifikation kann dabei beispielsweise anhand von Entscheidungsbäumen (z. B. C4.5) oder anhand von Bayes-Klassifikatoren erfolgen [LLS08, WAB+07].

Erkennen der Sprache des Dokumentes Die Bestimmung der Sprache eines Dokumentes kann durch die statistische Analyse von Wort- oder Buchstabenhäufigkeiten erfolgen [AMSWH06]. Sind die zu analysierenden Texte sehr kurz (z. B. nur eine Zusammenfassung), so dass keine aussagekräftige Statistik berechnet werden kann, so kann die Sprache des Dokumentes auch durch eine Analyse der auftretenden Stoppwörter vorgenommen werden.

Die Algorithmen zur Berechnung von dokumentenspezifischen Merkmalen sind teilweise mit hohem Aufwand verbunden (wie z. B. die Berechnung der PageRank-Bewertung [Ber05]). Da die dokumentenspezifischen Merkmale aber unabhängig von konkreten Anfragen sind, müssen die Bewertungen nur einmalig berechnet werden und können dann für alle Anfragen verwendet werden.

Bewerten auf Basis von Benutzerprofilen

Um bei der Bewertung von Dokumenten die individuellen Präferenzen des Benutzers zu berücksichtigen, bestimmt man, wie gut ein Dokument zum Profil des Benutzers und insbesondere zu den bisher vom Benutzer abgegebenen Bewertungen passt. Man analysiert dabei, wie ähnlich ein Dokument zu den bisher vom Benutzer bewerteten Dokumenten ist, und ermittelt anhand der berechneten Ähnlichkeit eine Bewertung für das neue Dokument. Dabei können Merkmale des Dokumentes (z. B. Autoren, Sprache, Verleger) entsprechend der individuellen Präferenzen des Benutzers gewichtet werden.

In Abhängigkeit von den im Benutzerprofil gespeicherten Daten (vgl. Abschnitt 4.1) können für die individuelle Bewertung von Dokumenten auch im-

plizite Bewertungen sowie die Profile von Benutzer mit ähnlichen Interessen herangezogen werden.

Ablaufplanung für die Berechnung von Zusatzinformationen und Bewertungen

Die Berechnung der verschiedenen Bewertungskriterien kann im Allgemeinen nebenläufig erfolgen, wobei jedes Bewertungskriterium durch einen spezialisierten Agent ermittelt wird. Es wird dabei die Eigenschaft von Multi-Agentensystemen genutzt, dass Agenten transparent auf verschiedene Computer verteilt werden können. Um doppelte Berechungen zu vermeiden, werden grundlegende Vorverarbeitungsschritte (z. B. das Zerlegen des Dokumentes in Wörter, das Entfernen von Stoppwörtern) von Agenten vorgenommen, deren Ergebnisse die anderen Agenten für die eigenen Berechnungen benutzen können.

Bei der Ablaufplanung für die Berechnung der Bewertungen und Metadaten muss analysiert werden, welche der Berechungen parallel und welche Berechnungen aufgrund von Abhängigkeiten sequentiell ausgeführt werden müssen. Darüber hinaus sollte beim Planen eines Berechnungsdurchlaufes die Wichtigkeit der einzelnen Bewertungskriterien für die Gesamtbewertung berücksichtigt werden. Damit wird erreicht, dass die Kriterien, die unbedingt notwendig zur Beurteilung eines Dokumentes sind, Vorrang vor den Kriterien haben die eine Verbesserung der Bewertungsqualität ermöglichen aber ggf. auch durch mit geringem Aufwand zu berechnende Kriterien ersetzt werden können.

Für das Ermitteln optimaler Anfrageabarbeitungsstrategien sowie die Auswahl der jeweils am besten geeigneten Agenten können vordefinierte Regeln oder die Planungskomponente der Agentenplattform (wie sie z. B. von JIAC IV bereitgestellt wird) genutzt werden.

5.5 Zusammenfassung

In diesem Kapitel wurden Strategien zur Auswahl von Informationsquellen und zur Optimierung von Suchanfragen vorgestellt. Es wurde ein Überblick zu den häufigsten Schwierigkeiten bei der Formulierung schlüsselwortbasierter Anfragen gegeben und gezeigt, wie durch die geeignete Kombination von Profiloptimierungsalgorithmen, Suchanfragen erweitert werden können.

Die Strategien zur Optimierung von Suchanfragen können sowohl im Hintergrund (für den Benutzer transparent) als auch im Dialog mit dem Benutzer angewendet werden. Eine iterative Optimierung von Anfragen hat dabei den

Vorteil, dass die jeweils vorgenommenen Optimierungsschritte dem Benutzer erläutert werden können und so vom Benutzer besser verstanden und akzeptiert werden. Eine Evaluation der verschiedener Anfrageoptimierungsstrategien im Rahmen eines realisierten Informationssystemen wird im Kapitel 8 beschrieben.

Neben den Methoden zur Anfrageoptimierung wurden in diesem Kapitel Strategien zum Ermitteln von Metadaten und zum Bewerten von Dokumenten vorgestellt. Es wurde erläutert, welche Nebenbedingungen beim Berechnen der Bewertungskriterien und beim Erstellen eines Ablaufplanes beachtet werden müssen. Bei der Realisierung einer Komponente zur Integration von Bewertungsalgorithmen muss darüber hinaus berücksichtigt werden, dass ein Informationssystem offen für neue Bewertungsmethoden sein muss, so dass neue Strategien (bzw. Agenten) in das System integriert werden können. Um Bewertungskriterien flexibel kombinieren zu können, sollte ein Bewertungsagent jeweils auf die Berechung eines Kriteriums spezialisiert sein. Durch die (auf die Präferenzen des Benutzers abgestimmte) Kombination mehrerer „schwacher Bewertungen“ (weak ratings) können hochqualitative Bewertungen abgeleitet werden. Die dabei eingesetzten Algorithmen und Methoden werden im Kapitel 6 ausführlich erläutert.

Die Analyse, welche Bewertungsagenten besonders gut mit den Bewertungen des Benutzers korrelieren und welche Gemeinsamkeiten zwischen den Bewertungsstrategien bestehen, ermöglicht es, die Leistungsfähigkeit der Bewertungsagenten zu evaluieren. Die Stärken und Schwächen von Bewertungsstrategien können dabei in Abhängigkeit vom jeweils betrachteten Szenario ermitteln werden, so dass Rückschlüsse darauf gezogen werden können, welche Agenten besonders leistungsfähig sind und welche Agenten nachtrainiert werden sollten.

Kapitel 6

Zusammenstellen von Ergebnismengen

Nachdem Verfahren erläutert wurden, mit denen Anfragen optimiert, Dokumente von heterogenen Datenquellen abgefragt und bezüglich verschiedener Gütekriterien bewertet werden können, stellt das folgende Kapitel Algorithmen zum Zusammenstellen optimaler Ergebnismengen vor. Es wird analysiert, welche Algorithmen zum Kombinieren von Bewertungen besonders leistungsfähig sind und wie geeignete Gewichtungen für die berücksichtigten Kriterien beim Zusammenführen der Bewertungen ermittelt werden können. Besondere Berücksichtigung finden dabei die Besonderheiten agentenbasierter Informationssysteme in Bezug auf Offenheit, Ausfallsicherheit und Adaptivität. Die Effektivität der entwickelten Algorithmen wird anhand mehrerer Benchmarkdatensätze und Szenarien gezeigt.

Der Aufbau des Kapitels gestaltet sich folgendermaßen: Zunächst werden die grundlegenden Fragestellungen zur Kombination von Filterstrategien diskutiert:

- Warum kombiniert man Filter? Welche Anwendungsfälle müssen berücksichtigt werden?
- Welche Strategien können zum Erstellen hochqualitativer Ergebnismengen angewendet werden?
- Wie wählt man die Agenten aus, die in ein „Expertengremium“ zum Bereitstellen hochqualitativer Ergebnisse aufgenommen werden sollen?
- Wie kann die Güte der Filterergebnisse evaluiert werden?
- Wie stark kann die Güte eines Filtersystems durch das Kombinieren von Bewertungen mehrerer Agenten verbessert werden?

Im Anschluss daran wird gezeigt, wie die entwickelten Verfahren im Rahmen eines agentenbasierten Informationssystems für die personalisierte Informationsbereitstellung eingesetzt werden können.

6.1 Warum kombiniert man Filterstrategien?

Bei der Kombination von Bewertungen von Filterstrategien wird zwischen dem Zusammenführen von disjunkten Ergebnismengen und dem Zusammenführen von Bewertungen für eine gegebene Dokumentenmenge unterschieden.

6.1.1 Collection Fusion

Strategien zum Erstellen einer Filterergebnismenge auf Basis disjunkter (meist aus verschiedenen Quellen stammenden) Dokumentenmengen bezeichnet man als **Collection Fusion**. Die Herausforderung beim Zusammenführen von Dokumenten aus disjunkten Quellen besteht darin, dass im Allgemeinen jede Dokumentenquelle eine spezifische Bewertungsstrategie verwendet und die Bewertungen somit nicht unmittelbar miteinander vergleichbar sind.

Collection Fusion-Strategien kommen in der Praxis vor allem in Meta-Suchmaschinen zum Einsatz, bei denen Dokumente aus verschiedenen Datensammlungen (die oft unterschiedliche Strukturen aufweisen und auf mehrere Computer verteilt sind) zusammengeführt werden sollen und das Zusammenführen auf Basis der Bewertungen der Quellen erfolgt. Die Heterogenität und Verteilung der Datenquellen wird vor dem Benutzer verborgen, so dass diesem der Eindruck vermittelt wird, dass alle Dokumente aus einer umfassenden einheitlichen Datenquelle stammen [LLC03].

Zum Zusammenführen von Dokumenten, die jeweils von nur einer Quelle bewertet wurden, können die folgenden Strategien herangezogen werden:

- Es werden jeweils die besten Dokumente von jeder Quelle in die Ergebnismenge übernommen. Der Vorteil dieses Ansatzes besteht darin, dass alle Dokumentenquellen in der Ergebnismenge berücksichtigt werden. Der Nachteil des Ansatzes besteht darin, dass keine einheitliche Bewertung der Dokumente erfolgt und somit eine Optimierung bezüglich spezifischer Gütekriterien (z.B. Exaktheit, Diversität, Verlässlichkeit) nicht möglich ist. Die Präsentation der Ergebnisse, sollte die Unterschiedlichkeit der für die Bewertung der Dokumente verwendeten Gütefunktionen berücksichtigen.

- Man definiert eine Funktion, die das Überführen von quellenspezifischen Bewertungen in ein standardisiertes Gütemaß ermöglicht. Es wird ana-

lysiert, wie zu einander ähnliche Dokumente, die aus unterschiedlichen Quellen stammen, durch die jeweiligen Quellen bewertet werden. So kann beispielsweise für den Fall, dass zwei Dokumentenquellen dieselbe Bewertungsstrategie (d. h. sie verwenden die gleichen Merkmale zur Berechnen der Dokumentengüte) aber eine unterschiedliche Skalierung für die Bewertungen verwenden, eine geeignete Transformationsfunktion für die Bewertungen abgeleitet werden. Verwendet jede der berücksichtigten Dokumentenquellen eigene Merkmale zum Bewerten von Dokumenten (z. B. eine Dokumentenquelle PageRank, eine andere Quelle aber Tf – Idf) kann im Allgemeinen keine Funktion ermittelt werden, die eine hinreichende Vergleichbarkeit der Bewertungen herstellt.

- Man greift beim Zusammenstellen von Ergebnismengen nicht auf die von den Dokumentenquellen ermittelten Bewertungen zurück, sondern verwendet auf die Bewertung von Dokumenten spezialisierte Komponenten. Der Nachteil dieses Ansatzes besteht darin, dass zusätzlicher Aufwand zum Berechnen der Bewertungskriterien notwendig ist. Durch das Ermitteln von quellenunabhängigen Bewertungen kann aber eine einheitliche, auf das Szenario und die Präferenzen des Benutzers abgestimmte Beurteilung der Dokumente erreicht werden. Stehen für alle Dokumente Bewertungen zur Verfügung, die durch eine einheitliche Menge von Bewertungsagenten ermittelt wurden, können **Data Fusion**-Verfahren angewendet werden.

Da in Hinsicht auf die gewählten Szenarien vor allem das Bereitstellen von personalisierten Dokumenten von Bedeutung ist, wird im Folgenden das Zusammenstellen von Filterergebnismengen anhand von durch Bewertungsagenten berechneten Kriterien analysiert.

6.1.2 Data Fusion

Das Zusammenfassen von Bewertungen, die durch eine Menge von Experten (bzw. Agenten) ermittelt wurden, zu *einer* Beurteilung wird als Data Fusion bezeichnet. Das Ziel ist es dabei, durch das Berücksichtigen mehrerer Expertenmeinungen (d. h. Bewertungskriterien), die Güte und die Verlässlichkeit der ermittelten Filterergebnisse zu verbessern. Die Grundidee ist, dass Dokumente, die von unterschiedlichen Bewertungsstrategien (d. h. Agenten) als relevant betrachtet werden, mit einer höheren Wahrscheinlichkeit zu den Interessen des Benutzers passen als Dokumenten, die nur von einem einzigen Agenten als relevant bewertet werden. Durch das Berücksichtigen unterschiedlicher Relevanzbewertungsstrategien können zudem fehlerhafte Bewertungen eines einzelnen Agenten durch andere Agenten überstimmt wer-

den. Es wird dabei das Vorgehen simuliert, das in vielen Bereichen beim Treffen komplizierter Entscheidungen angewendet wird (z. B. Diagnose einer schweren Krankheit), wobei jeweils ein aus mehreren Experten bestehendes Gremium eine gemeinsame Beurteilung finden muss [Pol06].

6.1.3 Delegieren von Anfragen

Eine Alternative zum Kombinieren von Bewertungen, die durch eine Menge von Experten (Agenten) ermittelt wurden, ist das Delegieren von Anfragen an den jeweils für eine Anfrage am besten geeigneten Experten. Um die Vor- und Nachteile des Delegierens von Anfragen im Vergleich zum Kombinieren von Bewertungen zu verdeutlichen, werden die Merkmale und die Unterschiede der beiden Herangehensweisen gegenübergestellt.

Für das Delegieren von Anfragen an einen einzelnen Agenten sprechen die folgenden Argumente:

- **Geringerer Bedarf an Ressourcen:** Da anstatt eines Gremiums von Agenten nur ein einzelner Agent die Relevanz der Dokumente bewertet, werden weniger Ressourcen benötigt. Außerdem entfällt das Zusammenführen der von den einzelnen Agenten ermittelten Bewertungen, das beim „Data Fusion“ zusätzlich Ressourcen benötigt.

- **Reduzierter Managementaufwand:** Bei der Verwendung eines Agentengremiums müssen die einzelnen Agenten untereinander synchronisiert werden. Erst wenn jeder Agent des Gremiums eine Bewertung ermittelt hat, können die Einzelbewertungen zu einer Gesamtbewertung zusammengeführt werden. Da dies bei einer Delegation an einen einzelnen Agenten entfällt, reduziert sich der Koordinationsaufwand, wodurch Anfragen etwas schneller als durch ein Agentengremium bearbeitet werden können [Mil05].

Für die Verwendung eines Expertengremiums anstatt eines einzelnen Experten sprechen die folgenden Argumente:

- **Verbesserung der Ergebnisqualität:** Durch die Kombination mehrerer Expertenmeinungen lässt sich eine Ergebnisqualität erreichen, die über der Ergebnisqualität aller einzelnen Experten liegt [Fre99, Edl07, AM01].

- **Höhere Zuverlässigkeit:** Bei Ausfall eines Agenten kann im Allgemeinen auf Basis der Ergebnisse der verbliebenen Experten (Agenten) weiterhin ein hochqualitatives Ergebnis ermittelt werden. Der Ausfall eines Filteragenten bleibt dem Benutzer verborgen.

- **Statistische Gründe:** Einige Lernalgorithmen (wie z. B. künstliche neuronale Netze) liefern in Abhängigkeit von der Initialisierung unterschiedlich gute Entscheidungsstrategien (bzw. verfügen über verschiedene Generalisierungsfähigkeit). Durch die geeignete Kombination mehrerer Bewertungsstrategien (z. B. Klassifikatoren) kann das Risiko, für eine Anfrage nicht zufriedenstellende Ergebnisse zu erhalten, reduziert werden [Pol06].

- **Höhere Flexibilität:** Auf einem Gremium von Agenten basierende Systeme ermöglichen das Integrieren neuer Agenten (Bewertungsstrategien), ohne dass das Risiko besteht, dass ein fehlerhafter (z. B. unzureichend getesteter) Agent die Ergebnisqualität stark negativ beeinflusst (vgl. „Höhere Zuverlässigkeit“). Dies ermöglicht es, auf veränderte Anforderungen zu reagieren und neue Algorithmen zu berücksichtigen.

- **Auswahl des für eine Anfrage geeigneten Agenten:** Soll die Relevanz eines Dokumentes nur durch einen einzelnen Agenten (anstatt durch ein Agentengremium) ermittelt werden, so muss eine Strategie zum Bestimmen eines geeigneten Agenten definiert werden. Dies ist insbesondere bei begrenzten Informationen über die jeweilige Anfrage (z. B. bei Verwendung des LETOR-Datensatzes [LXQ+07]) eine schwierige Aufgabe, so dass das Risiko besteht, dass Anfragen an weniger geeignete Agenten delegiert werden. Gremien von Agenten sollten jeweils so zusammengestellt werden, dass für möglichst viele Arten von Anfragen hochqualitative Ergebnisse garantiert werden können.

In Abhängigkeit vom jeweils betrachteten Szenario, haben beide Strategien (Auswahl eines geeigneten Agenten, Zusammenstellen eines Gremiums) spezifische Vorteile. Zum zuverlässigen Erreichen einer optimalen Ergebnisqualität sollte die Verwendung eines Agentengremiums gegenüber dem Auswählen eines einzelnen Agenten bevorzugt werden.

6.2 Wie evaluiert man die Ergebnisse?

Um die Güte von Filterverfahren und von Algorithmen zum Kombinieren von Dokumentenbewertungen zu beurteilen, müssen geeignete Verfahren und Maße definiert werden. Darüber hinaus müssen geeignete Benchmarkdatensätze ausgewählt werden, anhand derer die Güte der Filteralgorithmen berechnet werden kann.

6.2.1 Kriterien zum Bewerten von Filteralgorithmen

Zum Bewerten der Güte von Filterergebnissen können verschiedene Kriterien eingesetzt werden:

- **Exaktheit:** Die von einem Filteragenten ermittelte Ergebnismenge sollte einen hohen Anteil von relevanten Dokumenten enthalten.
- **Vollständigkeit der Ergebnismenge:** Die Filterergebnismenge sollte alle relevanten Dokumente enthalten.
- **Verlässlichkeit:** Die Güte der Ergebnisse sollte unabhängig von der Anfrage gleichbleibend hoch sein. Der Benutzer sollte sich darauf verlassen können, dass er für eine neu gestellte Anfrage gute Ergebnisse erhält.
- **Personalisierung:** Beim Zusammenstellen der Filterergebnisse sollten die individuellen Kenntnisse und Wünsche des Benutzers berücksichtigt werden. Es sollten bei der Bewertung von Dokumenten die im Benutzerprofil gespeicherten und die durch Analyse des Benutzerverhaltens (z. B. explizites und implizites Benutzerfeedback, „Click-Stream-Analyse“) abgeleiteten Daten einbezogen werden.
- **Diversität der Ergebnismenge:** Die Dokumente der Ergebnismenge sollten untereinander nicht zu stark übereinstimmen und verschiedene Aspekte abdecken. Duplikate sowie leicht modifizierte Versionen desselben Dokumentes sollten vermieden werden.

Weitere Kriterien, die in bestimmten Suchszenarien beachtet werden müssen, sind:

- **Anzahl der Ergebnisse:** Die Größe der Ergebnismenge sollte auf das jeweilige Szenario abgestimmt sein und insbesondere die zur Nutzung des Informationssystems verwendeten Geräte (z. B. Mobiltelefone) berücksichtigen.
- **Ressourcenanforderungen und Zeitbedarf:** Die Filterergebnisse sollten innerhalb einer angemessenen Zeit bereitstehen. Dabei muss das jeweilige Suchszenario (z. B. interaktive Nutzung, zeitgebergesteuerte Suche) berücksichtigt werden. Der zum Berechnen der Filterergebnisse notwendige Ressourcenbedarf sollte angemessen sein.
- **Strukturierung der Ergebnismenge:** Die Ergebnismenge sollte in Abhängigkeit von der verwendeten Benutzerschnittstelle so aufbereitet werden, dass der Benutzer schnell die Informationen findet, die er benötigt.

Die Besonderheiten der einzelnen Kriterien sowie Strategien zur automatisierten Evaluation dieser Kriterien werden in den folgenden Abschnitten erläutert.

6.2.2 Die Exaktheit

Eines der wichtigsten Kriterien zur Bewertung der Leistungsfähigkeit eines Filters ist die **Exaktheit**. Die Exaktheit gibt an, wie groß der Anteil von relevanten Dokumenten in der vom Filter ermittelten Ergebnismenge ist. Exaktheitsmaße werden im Bereich des Information Retrieval verwendet, wenn die Anzahl der für eine Anfrage relevanten Dokumente groß im Vergleich zur erwarteten Anzahl von Dokumenten ist. Da in diesem Fall nicht alle relevanten Dokumente Teil der Filterergebnismenge sein können („begrenzte Vollständigkeit"), sollte die Ergebnismenge keine irrelevanten Dokumente enthalten.

In Abhängigkeit von der Art der vom Filter bereitgestellten Ergebnismenge (unsortierte Dokumentenmenge, sortierte Liste von Dokumenten, Anzahl der berücksichtigten Relevanzlevel) werden unterschiedliche Exaktheitsmaße verwendet.

Precision

Um die Exaktheit einer unsortierten Filterergebnismenge zu beschreiben, verwendet man als Maß die **Precision**. Die Precision eines Filters F wird definiert als der Quotient zwischen der Anzahl der relevanten Dokumente in der Ergebnismenge R und der Größe der Ergebnismenge:

$$\text{Precision}_F(R) = \frac{|\text{Rel}(R)|}{|R|}$$

Dabei beschreibe $|\text{Rel}(R)|$ die Anzahl der relevanten Dokumente in vom Filter F ermittelten Ergebnismenge R.

Stellt ein Filter eine geordnete Ergebnisliste bereit, die mehr Dokumente enthält, als in der Anwendung angezeigt werden können, berechnet man die Precision nur für die ersten n Dokumente der Ergebnismenge. Man geht dabei davon aus, dass die Precision nur für die Dokumentenmenge berechnet werden sollte, die in der Anwendung angezeigt wird. Oft berücksichtigt man nur die ersten 10 oder 30 Elemente der Ergebnismenge, da viele Benutzer sich nur für die ersten Dokumente der Ergebnismenge interessieren und die Dokumente auf den hinteren Rängen ignorieren [Vor04, CHT99].

Für eine sortierte Liste von Filterergebnissen wird die Precision für einen Evaluationspunkt n folgendermaßen definiert:

$$\text{Precision}_F(R)@n = \frac{1}{n}\sum_{i=1}^{n}\text{Rel}(r_i)$$

Dabei sei $\text{Rel}(r_i) = \begin{cases} 1, & \text{wenn das } i\text{-te Dokument in } R \text{ relevant} \\ 0, & \text{wenn das } i\text{-te Dokument in } R \text{ irrelevant.} \end{cases}$

Enthält die Ergebnismenge R weniger als n Dokumente, so wird die Berechnung der Precision@n modifiziert. Dafür gibt es die folgenden Strategien:

- Man ermittelt $m = \min(|R|, n)$ und berechnet die Precision mit der Formel:

  $$\text{Precision}_F(R)@n = \frac{1}{m}\sum_{i=1}^{m}\text{Rel}(R_i)$$

  Für Ergebnismengen mit weniger als n Elementen berechnet man somit die Precision unter Verwendung eines Evaluationspunktes, der kleiner ist als n. Man folgt dabei dem Grundsatz, dass ein Filter nicht deshalb eine reduzierte Precision zugeordnet werden sollte, wenn er weniger als die erwarteten n Dokumente bereitstellt. Problematisch bei diesem Ansatz ist, dass die Vergleichbarkeit von Strategien durch die Verwendung unterschiedlicher Evaluationspunkte beeinträchtigt wird.

- Man fügt an die vom Filter gelieferte Ergebnisliste zufällig ausgewählte Dokumente an, so dass die sich ergebene Ergebnisliste mindestens n Dokumente besitzt. Dieses Vorgehen hat den Vorteil, dass die Formel zur Berechung der Precision@n nicht angepasst werden muss und dass immer Ergebnismengen identischer Größe miteinander verglichen werden [LXQ+07, TRE04].

Welche der beiden Strategien angewendet werden sollte, ist in Abhängigkeit vom jeweils betrachteten Szenario zu entscheiden. In der folgenden Arbeit wird bei der Berechnung der Precision@n die Ergebnismenge jeweils mit zufällig gewählten Dokumenten aufgefüllt (um eine Vergleichbarkeit zu mit `trecEval` [TRE04] generierten Bewertungen sicherzustellen).

Um für theoretische Überlegungen die Precision für alle Filterergebnismengen verwenden zu können, definiert man für die Precision der leeren Filterergebnismenge $\text{Precision}(\emptyset) = 1$. Eine umfassende Diskussion zu den Merkmalen der Precision findet man in [Rij79], [Far81] und [RBJ89b].

Verallgemeinerung der Precision

Bei der Berechnung der Precision wird die Anordnung der Dokumente in der Ergebnismenge nicht berücksichtigt. Informationssysteme stellen Ergebnisse aber oft in Form einer sortierten Liste dar. Um der Erwartung des Benutzers Rechnung zu tragen, auf den ersten Rängen der Ergebnisliste besonders relevanten Dokumente zu finden, definiert man die Exaktheitsmaße **Average Precision** und die **Mean Average Precision**, welche die Besonderheiten von sortierten Ergebnislisten berücksichtigen.

Man definiert die **Average Precision** (AP) für eine Anfrage q:

$$AP(R_q) = \frac{\sum_{i=1}^{n} \text{Precision}_F(R_q)@i * \text{ Rel}(r_i)}{\sum_{i=1}^{n} \text{Rel}(r_i)}$$

Die Average Precision ergibt sich somit als Mittelwert der Precision an den Auswertungspunkten, an denen durch den Filter ein relevantes Dokument ermittelt wurde.

Soll die Güte eines Filters über eine Menge von Anfragen berechnet werden, verwendet man die **Mean Average Precision** (MAP). Diese ist definiert als der Mittelwert der **Average Precision** über die Elemente einer Menge von Anfragen Q.

$$MAP(R_Q) = \frac{1}{|Q|} \sum_{q \in Q} AP(R_q)$$

Da für die Precision@n-Berechnung jeweils die ersten n Dokumente betrachtet werden, ist der Wert der **Average Precision** (und der **Mean Average Precision**) hoch, wenn die relevanten Dokumenten auf den vorderen Rängen der Ergebnisliste stehen [BYRN99, BV04].

Ein weiteres verbreitetes Maß zum Bewerten der Exaktheit einer geordneten Filterergebnisliste ist der **Mean Reciprocal Rank** (MRR). Man definiert zunächst für eine Anfrage q den **Reciprocal Rank** (RR):

$$\text{RR}(R_q) = \frac{1}{p}$$

wobei p der Rang in der Ergebnismenge ist, auf dem das erste relevante Dokument zu finden ist. So ergibt sich für eine Ergebnisliste $[R]$ zur Anfrage q, bei der das erste relevante Dokument auf Rang 8 zu finden ist: $\text{RR}(R_q) = \frac{1}{8} = 0.125$. Für eine Ergebnisliste $[\text{R}_q]$, die keine relevanten Dokumente enthält, definiert man $\text{RR}(R_q) = 0$. Mittelt man den Reciprocal Rank für alle Anfragen einer Anfragenmenge Q so erhält man den Mean Reciprocal Rank.

$$\text{MRR}(R_Q) = \frac{1}{|Q|} \sum_{q \in Q} \text{RR}(R_q)$$

Die Maße RR und MRR sollten immer dann verwendet werden, wenn es bei der Bewertung der Exaktheit eines Filters nur darauf ankommt, dass ein relevantes Dokument möglichst auf einem der vorderen Ränge der Ergebnismenge platziert ist [NZT07]. Für die Bewertung der Qualität von Suchmaschinen, bei denen der Benutzer eine Menge von relevanten Dokumenten erwartet ist MRR ungeeignet.

Normalized Discounted Cumulative Gain

Die bisher vorgestellten Maße unterscheiden lediglich zwischen relevanten und irrelevanten Dokumenten. In einigen Szenarien (z. B. bei der Auswertung von Benutzerbewertungen) wird oft zwischen mehreren Relevanzleveln unterschieden (z. B. bei Verwendung von 1 bis 5 Sternen zum Bewerten von Dokumenten). In [JK02] wird dazu für sortiere Dokumentenlisten, deren Elemente mehreren Relevanzleveln zugeordnet sind, als Bewertungsmaß NDCG (**Normalized Discounted Cumulative Gain**) vorgeschlagen.

Man definiert zunächst, den durch ein Dokument auf Position n der Ergebnisliste erreichten Informationsgewinn durch den Relevanzlevel des Dokumentes.

$$\mathrm{G}(R)@n = \mathrm{RL}(R_n)$$

Dabei beschreibt $\mathrm{RL}(R_n)$ den Relevanzlevel des Dokumentes auf Rang n der sortierten Ergebnisliste R. Wird in der Dokumentenmenge nur zwischen sehr relevanten (Relevanzlevel 2), relevanten (Relevanzlevel 1) und irrelevanten Dokumenten (Relevanzlevel 0) unterschieden, ergibt sich somit $G(n) = 2$, wenn das n-te Dokumente der Ergebnismenge *sehr relevant* ist, $G(n) = 1$, wenn das n-te Dokument *relevant* ist, und $G(n) = 0$, wenn das Dokumente auf Rang n *irrelevant* ist.

Auf der Basis des Gain $\mathrm{G}(R)@n$ definiert man den kumulierten Gewinn CG (**Cumulated Gain**):

$$\begin{aligned} \mathrm{CG}(R)@n &= \mathrm{CG}(R)@(n-1) + \mathrm{G}(R)@n \text{ , falls } n > 1 \text{ , und} \\ \mathrm{CG}(R)@1 &= \mathrm{G}(R)@1 \text{ , falls } n = 1 \end{aligned}$$

Da relevante Dokumente auf den vorderen Rängen einer sortierten Dokumentenmenge als wichtiger angesehen werden als die Dokumente auf den hinteren Rängen, modifiziert man die Definition des Cumulated Gain, indem man einen Gewichtungsterm (**Discount Faktor**) ergänzt. Den Discountfak-

tor wählt man meist als $log_b(n)$, wobei $b = 2$ [JK02, TCH+06]. Man definiert den **Discounted Cumulated Gain** (DCG):

$$\begin{aligned} DCG(R)@n &= CG(R)@n && \text{, wenn } n < b \\ DCG(R)@n &= DCG(R)@(n-1) + \frac{1}{\log_b n} && \text{, wenn } n \geq b \end{aligned}$$

Zusammengefasst in einer Gleichung erhält man:

$$\mathrm{DCG}(R)@n = \sum_{1 \leq i < b} r(i) + \sum_{b \leq i \leq n} \frac{1}{log_b i} r(i)$$

Um das Maß DCG für verschiedene Filteranfragen vergleichbar zu machen, führt man eine Normierung auf das Intervall $[0, 1]$ durch. Man definiert den Normierungsfaktor $Z(R)@n = \mathrm{DCG}_{FP}(R)@n$. Dabei sei FP der perfekte Filter, der die Dokumente entsprechend ihrer tatsächlichen Relevanz anordnet. Das Maß NDCG (**Normalized Discounted Cumulative Gain**) wird folgendermaßen definiert:

$$\begin{aligned} \mathrm{NDCG}(R)@n &= Z(R)@n * \mathrm{CG}(R)@n \\ &= Z(R)@n * \left(\sum_{1 \leq i < b} r(i) + \sum_{b \leq i \leq n} \frac{1}{log_b i} r(i) \right) \end{aligned}$$

Das Maß NDCG ist gemäß Definition genau dann 1, wenn die Dokumente nach ihrer Relevanz in der Ergebnisliste geordnet sind. Für eine Filterergebnismenge, in der alle Dokumente zum Relevanzlevel 0 gehören, ist das NDCG-Maß null. Falls $n = |R|$, schreibt man anstatt von $\mathrm{NDCG}(R)@n$ oft $\mathrm{NDCG}(R)$.

Das Maß NDCG ist darauf optimiert, die Exaktheit von sortierten Ergebnisliste mit mehreren Relevanzleveln zu messen, wie sie beispielsweise als Ergebnisse von Suchmaschinen auftreten, bei denen der Benutzer die Güte der Dokumente bewerten kann. Da die im Information Retrieval verwendeten Benchmarkdatensätze oft nur zwei Relevanzlevel verwenden (relevant, irrelevant) und die evaluierten Suchalgorithmen nur eine unsortierte Ergebnismenge bereitstellen, ist das Maß NDCG weniger stark verbreitet als die Precision.

Ein Beispiel für die Berechnung verschiedener Exaktheitsmaße ist in Abbildung 6.1 dargestellt. Die Graphik zeigt am Beispiel der gewichteten Kombination von zwei Filtern, dass die Maße Precision@10, $NDCG@10$ und MAP stark miteinander korrelieren.

Für die folgenden Analysen wird als Gütemaß vorrangig MAP verwendet, da dieses Kriterium auf die Bewertung von Dokumentenlisten mit zwei Relevanzleveln optimiert ist und kein Evaluationspunkt gewählt werden muss.

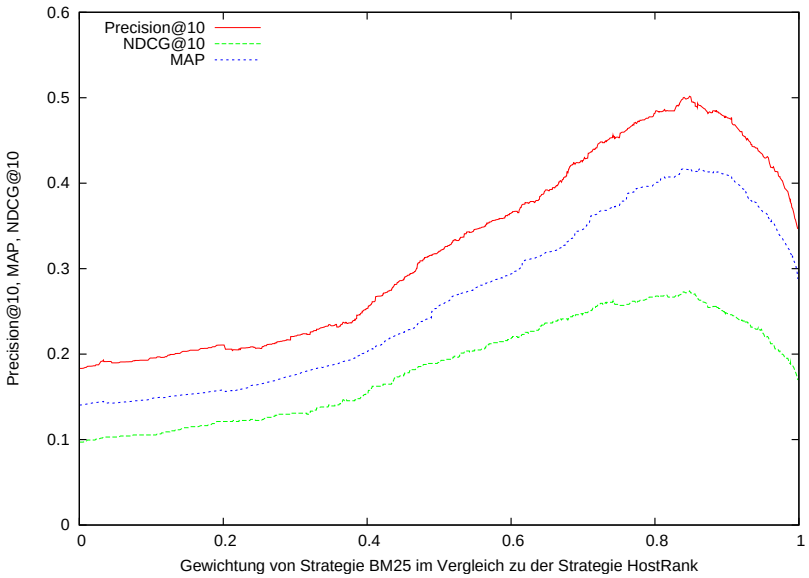


Abbildung 6.1: Die Graphik zeigt verschiedene Exaktheitsmaße am Beispiel der gewichteten Kombination der Filterstrategien '0' (BM25) und '7' (HostRank) des LETOR-TD2004-Datensatzes (vgl. Abschnitt 6.3). Die Kurven zeigen, dass die Maße Precision@10, $NDCG@10$ und MAP miteinander stark korrelieren.

6.2.3 Die Vollständigkeit

Um zu beschreiben, welcher Anteil der relevanten Dokumente einer Dokumentenmenge D in der von einem Filter ermittelten Ergebnismenge R enthalten ist, verwendet man das Maß **Recall**:

$$\text{Recall}(R, D) = \frac{|\text{Rel}(R)|}{|\text{Rel}(D)|}$$

$\text{Rel}(R)$ bezeichnet dabei die Menge der relevanten Dokumente in der ermittelten Ergebnismenge und $\text{Rel}(D)$ die Anzahl der relevanten Dokumente in der Menge, auf der der Filter arbeitet. Enthält die Ergebnismenge eines Filters alle Dokumente, so ist der Recall 1. Den Recall für eine Ergebnismenge, die ausschließlich irrelevante Dokumente enthält, sowie der Recall für die leere Ergebnismenge definiert man als null.

Im Bereich von Internet-Suchmaschinen, bei der für viele Anfragen die Anzahl der der relevanten Dokumente $|D|$ groß im Vergleich zur in der Anwendung präsentierbaren Filterergebnismenge ist, wird die Vollständigkeit zur Bewertung eines Filterergebnisses nur selten eingesetzt.
Ist $|\text{Rel}(D)| \gg |R| \geq |\text{Rel}(R)|$, so ist der Recall selbst bei sehr exakt arbeitenden Filtern (Filter mit hoher Precision) klein.

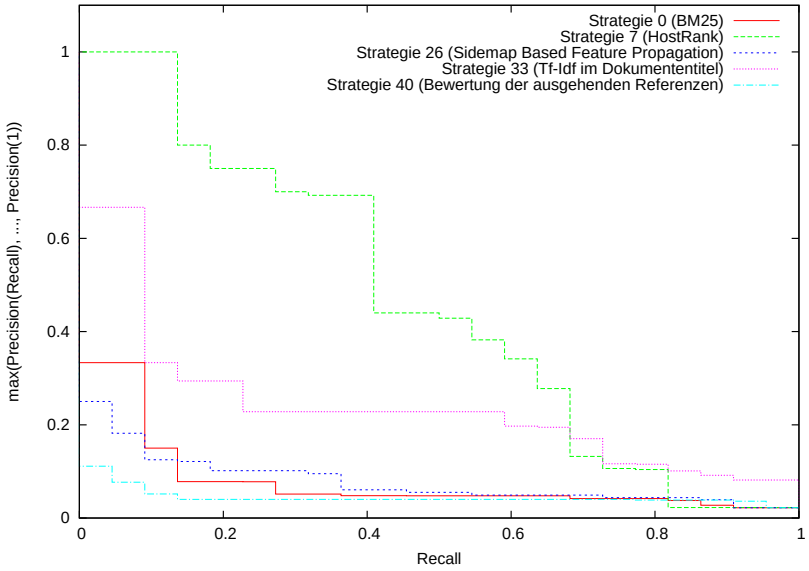


Abbildung 6.2: Die Abbildung zeigt den Recall-Precision-Graph für die Filterstrategien '0' (BM25), '7' (HostRank), '26' (Sidemap Based Feature Propagation), '33' (Tf − Idf im Dokumententitel) und '40' (Bewertung der ausgehenden Referenzen) des LETOR-TD2004-Datensatzes für die Anfrage mit der ID `160`. Um die Leistungsfähigkeit von Filterstrategien zu vergleichen, wählt man zunächst einen Recall-Level aus und analysiert dann die durch die Filter erreichte Precision.

Der **Recall** wird oft in Kombination mit der **Precision** für die Bewertung von Filterstrategien eingesetzt. Man verwendet dabei Recall-Precision-Graphen, die das charakteristische Verhalten eines Filters beschreiben (Graphik 6.2).

Um die Maße Precision und den Recall in einer Größe zusammenzufassen, bildet man das gewichtete harmonische Mittel aus den beiden Größen.

$$\mathrm{F}_\alpha(R, D) = \frac{\mathrm{Prec}(R) \cdot \mathrm{Recall}(R, D)}{(1-\alpha) \cdot \mathrm{Prec}(R) + \alpha\ \mathrm{Recall}(R, D)}\ ;\ \alpha \in [0, 1]$$

Die so definierte Größe wird **F-Measure** genannt [Rij79, MKSW99].

6.2.4 Auf einer Referenzstrategie basierende Kriterien

Ein weiteres wichtiges Kriterium zum Bewerten einer Filterstrategie ist (neben der Exaktheit und der Vollständigkeit) die Verlässlichkeit des Filters. Man analysiert dazu das Risiko, für eine gestellte Anfrage „schlechte" Ergebnisse zu erhalten. Im Folgenden werden verschiedene Maße und Strategien

zum Bewerten der Verlässlichkeit eines Filters vorgestellt. Es wird untersucht, wie die erreichte Filterpräzision über die Menge der Anfragen variiert und wie stark die Ergebnisqualität von der Qualität einer Referenzfilterstrategie abweicht.

Varianz und Standardabweichung

Zur Beschreibung der Streuung einer Zufallsvariable X wird die Varianz $\mathrm{Var}(X)$ bzw. die Standardabweichung $\sigma_X = \sqrt{\mathrm{Var}(X)}$ verwendet. Die Varianz ist definiert als das zweite zentrale Moment einer quadratisch integrierbaren Zufallsvariable und kann sowohl für diskrete als auch für stetige Zufallsvariablen verwendet werden.

$$Var(X) = \mathrm{E}\left((X - \mathrm{E}\,(X))^2\right)$$

Im diskreten Fall (z. B. für eine endliche Menge von Messwerten X) berechnet man die Varianz, indem man die Abstände der Messwerte vom Mittelwert quadriert, addiert und durch die Anzahl der Messwerte teilt.

$$\mathrm{Var}(X) = \sum_{x_i \in X} (x_i - \mathrm{E}(X))^2 \;;\; \mathrm{E}(X) = \frac{1}{|X|} \sum_{x_j \in X} x_j$$

Im Bereich des Information Retrieval kann die Varianz verwendet werden, um zu beschreiben, wie groß die Streuung der erreichten Filtergüte ist. Um die Verlässlichkeit einer Filterstrategie zu bewerten, ist die Varianz nur bedingt geeignet, da bei der Berechung nicht die unterschiedlichen Schwierigkeitsgrade der Anfragen berücksichtigt werden. Um die für eine Anfrage erzielte Ergebnisqualität besser einschätzen zu können, sollte die Filtergüte jeweils mit der von einem Referenzfilter erreichten Ergebnisqualität in Beziehung gesetzt werden.

Definition von Referenzfilterstrategien

Um die Leistungsfähigkeit und die Streuung einer Filterstrategie unabhängig vom Schwierigkeitsgrad einer Anfrage bewerten zu können, definiert man Filterstrategien, die als Bezugsstrategie für die Bewertung von Filtern dienen können. Eine Referenzstrategie sollte dabei die folgenden Anforderungen erfüllen:

- Die Referenzstrategie sollte geeignet sein, die spezifischen Besonderheiten des Information Retrieval-Problems zu beschreiben (z. B. den Schwierigkeitsgrad der Anfragen).

- Die Referenzfilterstrategie sollte unabhängig von der Anzahl und der Art der zu bewerteten Filter sein, da sich diese in offenen Filtersystemen relativ leicht ändern (z. B. durch das Klonen von Filterstrategien).

- Die Strategie sollte deterministisch berechenbar sein.

Es werden verschiedene Ansätze für das Definieren von Referenzfilterstrategien analysiert.

- **Der Filter, der alle Dokumente als relevant bewertet:** Eine sehr einfache Strategie besteht darin, alle Dokumente einer Dokumentenmenge D als relevant zu bewerten. Diese Filterstrategie erzielt den Recall 1 und die Precision $\frac{|\mathrm{Rel}(D)|}{|D|}$. Das Verhältnis der relevanten Dokumente zu der Gesamtmenge der Dokumente kann dabei genutzt werden, um den Schwierigkeitsgrad einer Anfrage zu bewerten, da der Anteil von relevanten Dokumenten die Wahrscheinlichkeit angibt, mit der ein zufällig ausgewähltes Dokument relevant ist.
  Der Vorteil dieser Strategie besteht darin, dass sie effizient zu berechnen ist und kein zusätzliches Wissen über die Anfragen und die Dokumente erfordert. Der Nachteil dieser Strategie ist, dass diese lediglich die Wahrscheinlichkeit berücksichtigt, ein relevantes Dokument zufällig zu finden, die Art und Formulierung der Anfragen aber unberücksichtigt lässt.

- **Zufällige Bewertungen für die Dokumente:** Wird von einem Filter erwartet, dass er Dokumente als sortierte Liste zurückgibt, so ist es nicht ausreichend Dokumente als relevant oder irrelevant zu bewerten, sondern es muss auch eine Ordnung auf der Dokumentenmenge definiert werden. Eine mögliche Strategie für einen Referenzfilter besteht darin, die Dokumente der Ergebnismenge zufällig anzuordnen. Analysiert man die Eigenschaften einer Filterergebnismenge R bestehend aus den ersten n Elementen einer sortierten Dokumentenmenge D, so erhält man die Precision $\mathrm{Prec}(D)@n = \mathrm{Prec}(R) = \frac{|\mathrm{Rel}(R)|}{n}$. Der Vorteil dieser Filterstrategie besteht (wie bei der Strategie, die alle Dokumente als relevant bewertet) darin, dass kein zusätzliches Wissen über die Dokumentenmenge benötigt wird und anhand dieser Strategie ermittelt werden kann, wie groß die Wahrscheinlichkeit ist, zufällig in einer Dokumentenmenge ein relevantes Dokument zu finden. Problematisch bei dieser Strategie ist, dass durch die Verwendung von Zufallszahlen die Berechnung nicht deterministisch ist.

- **Perfekte Bewertung für die Dokumente:** Als Referenzstrategie kann ein Filteragent dienen, der alle Dokumente entsprechend ihrer

tatsächlichen Relevanz bewertet. Der Vorteil dieser Strategie besteht darin, dass diese einfach berechenbar ist und eine Analyse ermöglicht, wie nahe ein realer Filteragent dem perfekten Filter kommt. Der Nachteil dieses Ansatzes ist, dass die perfekte Filterstrategie nur dann verwendet werden kann, wenn Benchmarkdaten zur Verfügung stehen. Im Kontext der Kombination existierender Filterstrategien zu einer neuen Filterstrategie wird bei der Verwendung des perfekten Filters als Referenzstrategie die Leistungsfähigkeit der gegebenen Filteragenten nicht berücksichtigt.

- **Kombination aus verfügbaren Filtern:** Ein Ansatz zum Definieren einer Referenzfilterstrategie besteht darin, eine gewichtete Kombination aus den verfügbaren Filtern zu verwenden. Anhand einer solchen Referenzstrategie kann ermittelt werden, ob eine neue Filterstrategie sich besser oder schlechter als die „durchschnittliche" Filterstrategie verhält. Der Vorteil dieser Definition für eine Referenzfilterstrategie ist, dass der Schwierigkeitsgrad von Anfragen unter Berücksichtigung der zur Verfügung stehenden Filterstrategien ermittelt wird und zusätzlich zur Wahrscheinlichkeit, zufällig ein relevantes Dokument herauszufiltern, weitere Aspekte (z. B. wie gut eine Anfrage zum Ermitteln relevanter Dokumente geeignet ist) berücksichtigt werden. Problematisch bei diesem Ansatz ist, dass die Güte der Referenzstrategie davon abhängt, wie gut die Gewichtungen für die Berechnung der Filterkombination gewählt werden (Berücksichtigen der Korrelation zwischen Filtern). Im Allgemeinen muss die Berechnung der Referenzstrategie angepasst werden, sobald neue Filterstrategien dem Informationssystem hinzugefügt werden. Ist die Menge der berücksichtigten Filterstrategien unveränderlich und sollen vor allem Strategien zur Kombination von Filterstrategien (basierend auf den existierenden Filtern) bewertet werden, so ist das gewichtete Mittel aus den verfügbaren Basis-Filtern eine gute Referenzstrategie. Ein Beispiel für die Verwendung einer über mehreren Filterstrategien gemittelten Präzision als Referenzfiltergüte zeigt das Diagramm 6.3.

- **Auswahl des jeweils besten Einzel-Filters:** Eine Alternative zum Definieren einer Referenzstrategie für eine gegebene Anfragemenge Q als gewichtetes Mittel aus den vorhandenen Filterstrategien besteht darin, eine Strategie zu definieren, die für jede Anfrage $q \in Q$ jeweils die beste bekannte Strategie anwendet. Im Gegensatz zum Ansatz die Referenzstrategie als gewichtetes Mittel aus den verfügbaren Strategien zu definieren, bei der die Gewichtungen unter Verwendung von

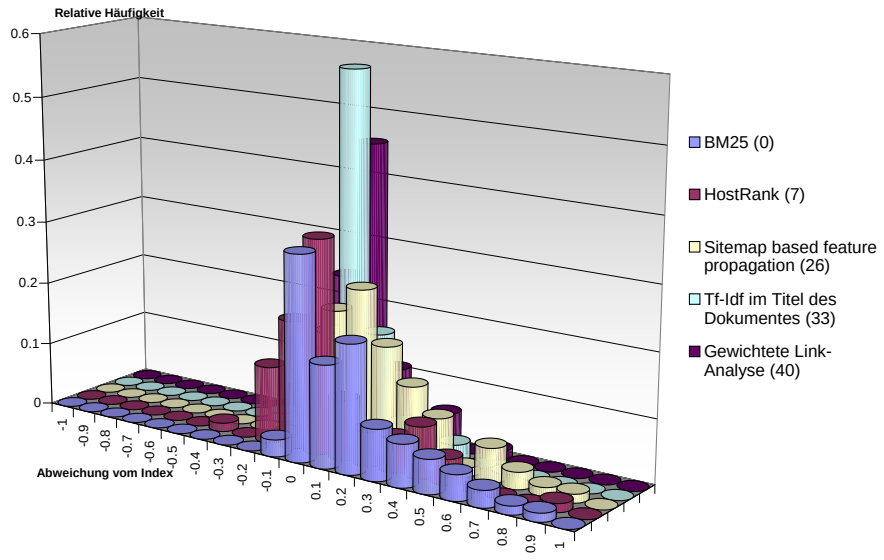


Abbildung 6.3: Das Diagramm zeigt die Häufigkeitsverteilung der Abweichung der Average Precision (AP) gemittelt über alle Filterstrategien und alle Anfragen des LETOR-TD2004-Datensatzes. Anhand der Graphik ist zu erkennen, dass die Häufigkeitsverteilungen rechtsschief sind, d. h. ein starkes Unterschreiten der gemittelten **Average Precision** selten ist. Der Fall, dass ein Filter für eine Menge von Anfragen deutlich besser als der Durchschnitt ist, tritt dagegen deutlich häufiger auf. Dies deutet darauf hin, dass Filter auf bestimmte Anfragetypen spezialisiert sind.

Zusatzwissen gewählt werden müssen, ist die Auswahl der jeweils besten Einzelstrategie einfach zu realisieren. Ein weiterer Vorteil ist, dass diese Referenzstrategie durch das Klonen von Filterstrategien nicht beeinflusst wird. Ein Nachteil dieses Ansatzes zum Definieren einer Referenzstrategie ist, dass diese aktualisiert werden muss, wenn neue Filterstrategien dem Dokumenten-Retrieval-System hinzugefügt werden. Die Referenzstrategie, die jeweils für jede Anfrage den besten Einzelfilter auswählt, ist insbesondere dann gut geeignet, wenn Algorithmen zur Kombination von Filterverfahren analysiert werden sollen und sich dabei die Menge der „Elementarfilter“ nicht verändert. Diese Filterstrategie ist dann eine geeignete Referenzstrategie, wenn analysiert werden soll, wie sich eine durch Kombination von Einzelfiltern definierter Filter im Vergleich zu einer Filterstrategie verhält, die für jede Anfrage

jeweils einen Filter auswählt (vgl. Abschnitt 6.1.3).

Auf Basis der durchgeführten Analyse erscheinen zum Bewerten von Filterkombinationsstrategien die Strategien „Mittelwert aus verfügbaren Filtern“ und „Auswahl des jeweils besten Einzel-Filters“ am besten geeignet, da diese auf Grundlage der vorhandenen Filter ermittelt werden können und zeigen, welche Verbesserung der Filterqualität durch die gewählte Filterkombinationsstrategien erreicht werden kann.

Bewerten des Risikos von Filtern

Unter Verwendung der Referenzfilterstrategien werden Maße abgeleitet, die das Risiko beschreiben, dass ein Filter für Anfragen „schlechte“ Ergebnisse liefert. Durch den Bezug auf einen Referenzfilter wird das durch die Komplexität der Anfrage gegebene Risiko („unspezifisches Risiko“) vom durch die jeweils verwendete Filterstrategie bedingte („filterspezifische“, „systematische“) Risiko voneinander getrennt.

Für die Analyse des Risikos eines Filters werden die folgenden Maße betrachtet:

- Die Wahrscheinlichkeit, dass der Filter für eine Anfrage schlechter als die Referenzstrategie ist.

- Der Betrag, mit der eine Filterstrategie die Güte der Referenzstrategie maximal unterschreitet („maximaler Schaden“).

- Das Produkt aus der Wahrscheinlichkeit, dass die Ergebnisse für eine Anfrage schlechter sind als die Referenzstrategie, mit der dabei im Mittel auftretenden Abweichung („Schaden“). Um starke Abweichungen von der Referenzstrategie stärker zu gewichten, kann anstatt der einfachen Abweichung vom Index die p-Norm zwischen der durch die Referenzstrategie und zu bewertenden Filterstrategie verwendet werden.

- Korrelation mit dem Vergleichsindex: Die Korrelation mit der Vergleichsstrategie gibt an, wie gleichgerichtet sich eine Filterstrategie und die Referenzstrategie verhalten. Werden Filterstrategien im Vergleich zu unterschiedlichen Referenzstrategien betrachtet, muss dabei die Varianz der Referenzstrategie berücksichtigt werden. Anstatt der Korrelation verwendet man den *Beta-Faktor* $\mathrm{Beta}(F, I) = \frac{\mathrm{Cov}(F,I)}{\mathrm{Var}(I)}$ (wobei I die verwendete Referenzstrategie und F die zu bewertende Filterstrategie beschreibt)[FKPT02].

Welches Maß zum Beschreiben des Risikos in einer Anwendung eingesetzt werden sollte, hängt vom jeweils betrachteten Szenario und den Erwartungen der Benutzer ab.

6.2.5 Personalisierung

Bei der Realisierung von Informationssystemen sollten zur Bewertung von Dokumenten Strategien berücksichtigt werden, welche die individuellen Erwartungen und Kenntnisse der Benutzer einbeziehen. Um zu beurteilen, wie gut Dokumente zu den Präferenzen eines Benutzers passen, analysiert man die Ähnlichkeit zwischen den zu einer Anfrage ermittelten Dokumenten und den bereits von Benutzer bewerteten Dokumenten.

Die Grundlage für die personalisierte Bewertung von Dokumenten bildet ein Benutzerprofil, in dem die Daten über den Benutzer verwaltet werden (vgl. Abschnitt 4.1). Von besonderer Bedeutung für die personalisierte Filterung sind dabei (implizite und explizite) Benutzerbewertungen sowie die Erwartungen der Nutzer [HJJ+07].

Um zu bestimmen, wie gut die von einem Agenten ermittelten Filterergebnisse zu den Präferenzen eines Benutzers passen, berechnet man, wie ähnlich die ermittelten Dokumente zu den bisher vom Benutzer bewerteten Dokumenten sind. Darüber hinaus können aus dem Kontext der Dienstnutzung (z. B. benutztes Endgerät, Internetprovider, bevorzugte Sprache) Informationen über den Benutzer gewonnen werden, anhand derer Dokumente bewertet werden können. Dies ermöglicht es, auch für neue Benutzer (die bisher noch keine Bewertungen abgegeben haben) die Beurteilung von Dokumenten zu personalisieren.

Stehen zur Evaluation eines Informationssystems Benchmarkdaten zur Verfügung, welche die Bewertungen von Nutzern zu einer Menge von gespeicherten Anfragen beinhalten, können diese verwendet werden, um zu analysieren, wie gut sich das Informationssystem an die Präferenzen eines Benutzers anpassen kann. Man trainiert dazu das Informationssystem mit den Bewertungen eines Benutzers und analysiert, wie gut die für Testanfragen ermittelten Dokumente zu den Benutzerbewertungen passen.

6.2.6 Vielfältigkeit der Ergebnismenge

Ein weiteres Kriterium, das zum Beurteilen der Güte einer Ergebnismenge herangezogen werden kann, ist die Vielfältigkeit der Dokumente in einer Ergebnismenge. Man analysiert dabei, inwiefern die Dokumente der Ergebnismenge unterschiedliche Aspekte berücksichtigen und wie groß die Ähnlichkeit zwischen den ermittelten Dokumenten ist. Unter Verwendung des

Kriteriums **Diversität** kann das Vorliegen von Duplikaten (d. h. inhaltlich übereinstimmenden Dokumenten, die unterschiedliche Formatierungen aufweisen) bestraft sowie das Vorliegen von Dokumenten aus unterschiedlichen Themenbereichen gefördert werden. Um die Diversität einer Dokumentenmenge zu berechnen, definiert man ein Maß, das die Ähnlichkeit von Dokumenten beschreibt. Zur Bestimmung der Ähnlichkeit können die folgenden Attribute und Algorithmen eingesetzt werden:

- **Textbasierte Attribute:** Levenshtein-Distanz zwischen den Texten, Ähnlichkeit der n-Gram-Statistik auf Wort- oder Buchstabenebene
- **Datumsbasierte Attribute:** Die Zeitdifferenz zwischen den Datumsangaben (z. B. Veröffentlichungsdatum)
- **Semantische Attribute:** Auf Basis von semantischen Attributen (z. B. Verlag, Autor oder Sprache des Textes) wird die Ähnlichkeit der Entitäten berechnet (z. B. anhand von graphbasierten Algorithmen).

Bei der Verwendung der Diversität als Gütekriterium muss beachtet werden, dass damit nicht die Qualität eines einzelnen Dokumentes, sondern einer Menge von Dokumenten beschrieben wird.

6.2.7 Diskussion der Evaluationsstrategien

Im diesem Abschnitt wurden Kriterien zur Evaluation von Filterstrategien vorgestellt. Unter Verwendung von Benchmarkdatensätzen (Abschnitt 6.3) können damit Filter- und Filterkombinationsverfahren automatisiert bewertet werden.

Evaluiert man die Güte von bestehenden Filtersystemen, muss beachtet werden, dass neben dem der objektiv messbaren Güte einer Dokumentenmenge auch die subjektive Zufriedenheit des Benutzers berücksichtigt werden muss. Bewertungen für Dokumente sind immer subjektiv, so dass jeder Benutzer andere Dokumente als relevant einstuft. Daher erreichen Benutzer beispielsweise beim Bewerten von Benchmarkdaten nicht immer die optimale Precision bzw. den maximalen Recall.

6.3 Verwendete Benchmarkdatensätze

Um die Güte von Filterstrategien und Algorithmen zur Kombination von Bewertungsstrategien zu messen, verwendet man Benchmarkdatensätze. Auf

Basis der Benchmarkdatensätze wird ein reproduzierbares IR-Szenario definiert, das eine praxisnahe Ermittlung der Leistungsfähigkeit von Filterstrategien ermöglicht und die Grundlage für eine Vergleichbarkeit der von verschiedenen Forschergruppen entwickelten Algorithmen bildet.

Um die für ein offenes personalisiertes Filterframework entwickelten Algorithmen zu testen, wird der LETOR-Datensatz [QLL07, LXQ+07] verwendet.

Der LETOR-Datensatz („**Le**arning **To R**ank“) ist ein Datensatz mit dem Algorithmen zum Kombinieren von Dokumentenbewertungsstrategien in Hinsicht auf ihre Leistungsfähigkeit analysiert werden können. Der Datensatz besteht dabei aus einer Menge von Anfragen (ausschließlich beschrieben durch eine `ID`). Jeder Anfrage ist eine feste Menge von Dokumenten zugeordnet, wobei für jedes Dokument gegeben ist, inwieweit es für die Anfrage relevant ist. Darüber hinaus stehen für jede Anfrage die Relevanzbewertungen mehrerer Bewertungsstrategien („Filteragenten“) zur Verfügung, die als Grundlage für die Kombination mehrerer Bewertungsverfahren dienen. Anhand der gegebenen Relevanzbewertungen lassen sich die ursprünglich verwendeten Anfragen nicht mehr rekonstruieren, so dass neue Bewertungsstrategien ausschließlich durch die geeignete Kombination der gegebenen „Basisfilterstrategien“ abgeleitet werden können.

In Verbindung mit dem LETOR-Datensatz wird ein Evaluationsprogramm zur Verfügung gestellt, das eine Vergleichbarkeit der mit verschiedenen Algorithmen erzielten Ergebnisse sicherstellen soll und die Maße NDCG@n, MAP und Precision@n ($n \in \{0, 1, \ldots 15\}$) berechnet. Der LETOR-Datensatz besteht aus drei Teildatensätzen, die sich jeweils durch die Art der gegebenen Dokumente und Filterstrategien unterscheiden.

Der LETOR-OHSUMED-Datensatz [XLL07] basiert auf Dokumenten des OHSUMED-Korpus [HBLH94]. Für die Bewertung der Relevanz der Dokumente werden drei Level („irrelevant“, „vielleicht relevant“ und „relevant“) unterschieden. Es stehen die Bewertungen von 25 Filterstrategien zur Verfügung.

Der LETOR-TD2003- und der LETOR-TD2004-Datensatz basieren auf Dokumenten des Web-Tracks der TREC2003- und TREC2004-Korpora, die auf Grundlage von zur Toplevel-Domäne `.gov` gehörenden Webseiten erstellt wurden. Aus dem TREC-Korpus (bestehend aus ca. einer Million HTML-Dokumenten und ca. 11 Millionen Verweisen zwischen den Dokumenten) wurden für jede Anfrage jeweils ca. 1000 potentiell relevante Dokumente (unter Verwendung eines auf dem Algorithmus BM25 [RWB98] basierten Verfahrens) ausgewählt („**T**opic **D**estilation“). Für die ausgewählten Dokumente wurden die Relevanzbewertungen sowohl mit Hilfe von allgemein akzeptierten Filterstrategien (wie z. B. Tf – Idf-basierten Strategien)

als auch mit komplexen Filterstrategien (z.B. HostRank, Feature propagation und Topical PageRank) bewertet. Die von Menschen definierte Referenz-Relevanzbewertung unterscheidet zwischen relevanten und irrelevanten Dokumenten (zwei Relevanzlevel). Der LETOR-TD2003-Datensatz umfasst 50, der LETOR-TD-2004-Datensatz 75 Anfragen. Für den LETOR-TD2003/TD2004-Datensatz stehen 44 Bewertungsstrategien zur Verfügung. Einen Überblick über die Strategien gibt die Tabelle A.1.

6.4 Wie stellt man Ergebnismengen zusammen?

Für das Zusammenstellen von Dokumentenmengen auf Basis einer Menge von (durch Bewertungsagenten ermittelten) Bewertungen können unterschiedliche Typen von Algorithmen eingesetzt werden. Bevor zwei Ansätze (Sozialwahlstrategien, die ratingbasierte Kombination von Bewertungen) im Detail erläutert werden, wird zunächst ein Überblick über Algorithmen zum Kombinieren von Bewertungen gegeben.

6.4.1 Wahlverfahren

Um Bewertungen mehrerer Filterstrategien zusammenzufassen, können Wahlverfahren eingesetzt werden. Dabei werden die Dokumente als die Kandidaten und die Bewertungsagenten als Wähler betrachtet. In Abhängigkeit von der Art der verwendeten Bewertungen (z. B. numerische Bewertungen – „ratings“, sortierte Dokumentenliste – „rankings“, booleschen Bewertungen) muss ein geeignetes Wahlverfahren ausgewählt werden.

Der Vorteil von Wahlverfahren besteht darin, dass diese im Allgemeinen gut nachvollziehbar sind und ihre Leistungsfähigkeit in vielfältigen Anwendungsbereichen nachgewiesen wurde. Wahlverfahren ermöglichen es meist, Gewichtungen für jeden Wähler (d. h. Filterstrategie) zu berücksichtigen und so den Einfluss eines Agenten auf das Endergebnis zu steuern. Die Wahlverfahren skalieren im Allgemeinen gut mit der Anzahl der berücksichtigten Bewertungsstrategien („Wähler“). Die Bewertungen werden separat für jedes Dokument (bzw. „Kandidaten“) ermittelt, so dass Merkmale unberücksichtigt bleiben, die die Ergebnismenge im Ganzen beschreiben (wie z. B. die Diversität).

Eine ausführliche Diskussion der Merkmale und Besonderheiten von Wahlverfahren erfolgt im Abschnitt 6.5.

6.4.2 Regeln und Heuristiken

Eine weitere Möglichkeit Bewertungen unterschiedlicher Bewertungsagenten zu kombinieren, besteht in der Verwendung von durch Experten definierten Heuristiken. Es werden dabei Regeln definiert, anhand derer in Abhängigkeit vom Vorliegen bestimmter Voraussetzungen jeweils eine geeignete Methode zum Kombinieren von Bewertungen ausgewählt wird. Kriterien, die als Grundlage zur Auswahl einer geeigneten Methode dienen können, sind beispielsweise das jeweils betrachtete Szenario oder die Interessenschwerpunkte des Benutzers. Das Vorgehen orientiert sich dabei an den in der betriebswirtschaftlichen Entscheidungstheorie [GT05] verwendeten Entscheidungsbäumen. Im Gegensatz zu der Verwendung eines fest definierten Verfahrens zum Zusammenfassen von Bewertungen, erlaubt der Einsatz von Heuristiken, die Besonderheiten von Dokumenten in ausgewählten Szenarien besser zu berücksichtigen. Allerdings erfordert das Erstellen von guten Regeln Erfahrung und Expertenwissen.

6.4.3 Strategien der globalen Optimierung

Anstatt von Heurisiken können zum Zusammenstellen von Filterergebnismengen Algorithmen zur globalen Optimierung eingesetzt werden, wie z. B. Constraint Solver [BHZ06, BNW06] oder genetische Algorithmen. Man definiert dabei Bedingungen, die das Ergebnis erfüllen muss, sowie ein Gütefunktional, das die Qualität einer gefundenen Lösung beschreibt. Im Gegensatz zur Kombination von Bewertungen auf Basis von Wahlverfahren, bei denen die Güte jedes Dokumentes separat betrachtet wird, erlauben die Methoden zur globalen Optimierung auch Merkmale der Ergebnismenge (wie z. B. Ähnlichkeit zwischen Dokumenten und die Diversität von Informationsquellen) zu berücksichtigen. In Informationssystemen können dadurch benutzer- und anwendungsfallspezifische Anforderungen für das Zusammenstellen einer optimalen Ergebnismenge herangezogen werden. Ein System zum Erstellen von Filterergebnismengen auf Basis von ganzzahligen Optimierungsverfahren wird in [Kun06] vorgestellt; der Einsatz eines Constraint Solvers zum Zusammenstellen von Filterergebnismengen wird in [Edl07] erläutert.

6.4.4 Gruppieren von Ergebnissen

Anstatt aus einer Dokumentenmenge nur die am besten passenden Dokumente herauszufiltern, kann man alternativ auch die Menge von potentiell relevanten Dokumenten so aufbereiten, dass der Benutzer die für ihn interessanten Dokumente leicht finden kann. Der erfahrene Benutzer kann dann

selbst entscheiden, welche Teile der Dokumentenmenge für ihn relevant sind. Er muss sich nicht auf automatisiert berechnete Relevanzbewertungen verlassen. Zur Beschreibung der ermittelten Cluster werden wichtige Begriffe von den im Cluster zusammengefassten Dokumenten ermittelt.

Eine Methode, Dokumentenmengen automatisiert zu strukturieren, bieten Cluster-Algorithmen. Man definiert dazu ein Ähnlichkeitsmaß für Dokumente (vgl. Abschnitt 6.2.6 und fasst jeweils ähnliche Dokumente unter Verwendung von Algorithmen zum hierarchischen Clustern in Gruppen zusammen. Details zum Clustern von Dokumentenmengen sowie deren Anwendung in Informationssystemen findet man in [MKA07, KLR+04, Hec06]. Ein Beispiel für das Clustern einer Ergebnismenge im Informationssystem PIA+COMM zeigt Abbildung 6.4.

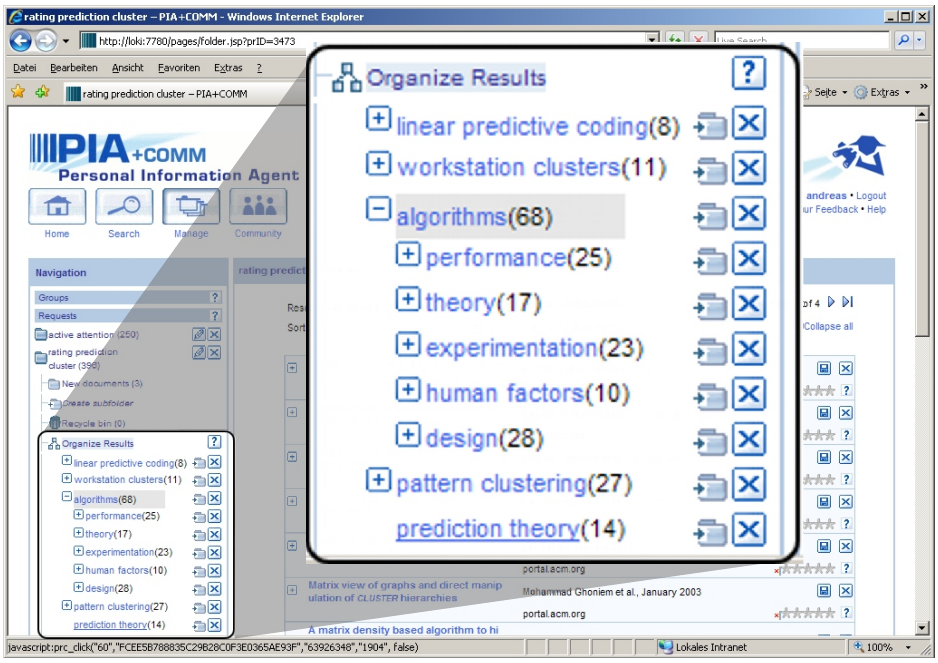


Abbildung 6.4: Die Abbildung zeigt das thematische (hierarchische) Gruppieren von den zu einer Anfrage als relevant ermittelten Dokumenten im Informationssystem PIA+COMM.

6.4.5 Diskussion der Strategien zum Zusammenstellen von Ergebnismengen

Zum Erstellen von Filterergebnismengen können sehr unterschiedliche Algorithmen eingesetzt werden, die sich in der Berechnungskomplexität und der

dabei erreichten Ergebnisqualität unterscheiden. In Abhängigkeit von der jeweiligen Anwendung müssen daher jeweils geeignete Algorithmen ausgewählt werden. In den folgenden Abschnitten werden schwerpunktmäßig die wahlbasierten Algorithmen zur Kombination von Filterbewertungen im Detail diskutiert, da Wahlverfahren eine effiziente und zuverlässige Kombination verschiedener Filterstrategien ermöglichen und kein zusätzliches Domänenwissen benötigen [FS94].

6.5 Wie können Bewertungen mehrerer Filterstrategien kombiniert werden?

Sollen die von unterschiedlichen Filteragenten stammenden Bewertungen unter Verwendung von Wahlverfahren miteinander kombiniert werden, so muss in Abhängigkeit von der Art der von den einzelnen Agenten bereitgestellten Bewertungen (boolesch, sortierte Liste, numerische Ratings) ein geeigneter Algorithmus gewählt werden. Die für die Kombination der unterschiedlichen Typen von Bewertungen eingesetzten Algorithmen werden im Folgenden erläutert.

6.5.1 Kombination boolescher Bewertungen

Für das Zusammenfassen boolescher Dokumentenbewertungen („relevant“/„irrelevant“) kommen im Allgemeinen mengenbasierte Kombinationsstrategien und Voting-Verfahren zum Einsatz. In der Praxis haben Methoden zur Kombination von booleschen Bewertungen nur eine begrenzte Bedeutung, da meist Filterstrategien verwendet werden, die detaillierter Relevanzbewertungen (z. B. in Form von Rankings oder Ratings) zur Verfügung stellen. Die Verfahren zur Kombination von booleschen Bewertungen demonstrieren aber gut die grundlegenden Strategien zum Kombinieren von Filtern und sollen daher aus Gründen der Vollständigkeit hier erläutert werden.

Kombination von Bewertungen durch Schnittmengenbildung

Ein Verfahren zur Kombination mehrerer Dokumentenbewertungen besteht darin, die von allen Filterstrategien als relevant bewerteten Dokumente zu ermitteln. Ist $F_i(D)$ die vom Agenten i als relevant ermittelte Menge der Dokumentenmenge D so ergibt sich als Ergebnis der Kombination der Filter $1, \ldots, n$:

$$\mathrm{Comb}_{\cap}(F_1(D), \ldots, F_n(D)) = F_1(D) \cap \ldots \cap F_n(D)$$

Kombination von Filtern durch Vereinigungsmengenbildung

Ein alternatives Verfahren zur Kombination von Filtern ist, jeweils die Dokumente als relevant einzustufen, die von mindestens einem Filter als relevant bewertet wurden. Man definiert dazu:

$$\text{Comb}_{\cup}(F_1(D), \ldots, F_n(D)) = F_1(D) \cup \ldots \cup F_n(D)$$

Kombination von Mengen durch Voting

Als Verallgemeinerung der Strategien $\text{Comb}_{\cap}$ und $\text{Comb}_{\cup}$, definiert man die Strategie Comb_m, die als Ergebnismenge alle Dokumente enthält, die von mindestens m Filtern als relevant bewertet wurden:

$$\text{Comb}_m(F_1(D), \ldots, F_n(D)) = \bigcup_{J \subseteq I,\ |J| \geq m} \left(\bigcap_{j \in J} F_j(D) \right)$$

Für $m = 1$ erhält man $\text{Comb}_{\cup}$, für $m = n$ erhält man $\text{Comb}_{\cap}$. Der Kombinationsalgorithmus kann als eine Voting-Strategie angesehen werden, wobei ein Dokument als relevant eingestuft wird, wenn mindestens m Filteragenten das Dokument als relevant bewerten.

Analyse von Precision und Recall

Untersucht man die Eigenschaften der Strategien $\text{Comb}_{\cap}$ und $\text{Comb}_{\cup}$, so stellt man fest:

- Bei der Bildung der Schnittmenge von Filterergebnismengen vergrößert sich der Recall nicht. Es gilt:

$$\text{Recall}(\text{Comb}_{\cap}(F_1(D), \ldots, F_n(D))) \leq \min_{i \in \{1,\ldots,n\}} (\text{Recall}\,(F_i\,(D)))$$

- Bei der Bildung der Vereinigungsmenge von Filterergebnismengen verringert sich der Recall nicht. Es gilt:

$$\text{Recall}(\text{Comb}_{\cup}(F_1(D), \ldots, F_n(D))) \geq \max_{i \in \{1,\ldots,n\}} (\text{Recall}\,(F_i\,(D)))$$

- Die Precision kann sich sowohl bei der Bildung von Schnittmengen als auch bei der Bildung von Vereinigungsmengen vergrößern, verringern oder gleichbleiben.

Durch die Kombination von Filtern lässt sich der Recall zielgerichtet steuern. Über die Precision lassen sich keine allgemeingültigen Aussagen ableiten.

Beweise

Beweis (Schnittmengenbildung vergrößert den Recall *nicht).* Gegeben seien n Filterstrategien $F_1, \ldots, F_n$ und eine Menge von Dokumenten D. Weiterhin bezeichne $\mathrm{Rel}(D)$ die Menge der relevanten Dokumente zu einer Anfrage.
Aus $F(D) \subseteq D$ folgt:
$F_i(F_j(D)) \subseteq F_j(D)$ sowie $\mathrm{Rel}(F_i(F_j(D))) \subseteq \mathrm{Rel}(F_j(D))$. Es gilt:

$$\mathrm{Recall}(\mathrm{Comb}_{\cap}(F_i(D), F_j(D))) = \frac{|\mathrm{Rel}(F_i(F_j(D)))|}{|\mathrm{Rel}(D))|}$$

Mit $F_i(F_j(D)) \subseteq F_j(D)$ folgt $\mathrm{Rel}(F_i(F_j(D))) \subseteq \mathrm{Rel}(F_j(D))$ und damit auch

$$\begin{aligned}\mathrm{Recall}(\mathrm{Comb}_{\cap}(F_i(D), F_j(D))) = \frac{|\mathrm{Rel}(F_i(F_j(D)))|}{|\mathrm{Rel}(D))|} &\leq \frac{\mathrm{Rel}(F_j(D))}{|\mathrm{Rel}(D))|} \\ \frac{\mathrm{Rel}(F_j(D))}{|\mathrm{Rel}(D))|} &= \mathrm{Recall}\,(F_j(D))\end{aligned}$$

□

Beweis (Vereinigungsmengenbildung verringert den Recall nicht). Analog zum vorigen Beweis gilt:

$$\mathrm{Recall}(\mathrm{Comb}_{\cup}(F_i(D), F_j(D))) = \frac{|\mathrm{Rel}(F_i(D) \cup F_j(D))|}{|\mathrm{Rel}(D))|}$$

Mit $F_i(D) \cup F_j(D) \supseteq F_j(D)$ folgt $\mathrm{Rel}(F_i(D) \cup F_j(D)) \supseteq \mathrm{Rel}(F_j(D))$ und damit auch

$$\begin{aligned}\mathrm{Recall}(\mathrm{Comb}_{\cup}(F_i(D), F_j(D))) = \frac{|\mathrm{Rel}(F_i(D) \cup F_j(D))|}{|\mathrm{Rel}(D))|} &\geq \frac{|\mathrm{Rel}(F_j(D))|}{|\mathrm{Rel}(D))|} \\ \frac{|\mathrm{Rel}(F_j(D))|}{|\mathrm{Rel}(D))|} &= \mathrm{Recall}(F_j(D))\end{aligned}$$

□

Beispiele: Die Änderung der Precision bei $\mathrm{Comb}_{\cap}$ *und* $\mathrm{Comb}_{\cup}$. Die folgenden Beispiele zeigen, dass beim Bilden von Vereinigungsmengen und Schnittmengen die Precision sowohl steigen als auch sinken kann.

Beispiel 1: Das Beispiel zeigt, dass bei Verwendung von $\mathrm{Comb}_{\cap}$ Precision und Recall sinken sowie dass bei Verwendung von $\mathrm{Comb}_{\cup}$ Precision und Recall ansteigen.

DokumentID	1	2	3	4			
Relevanz	Rel	Rel	Irrel	Irrel		Precision	Recall
F_1	x	-	x	-		1/2	1/2
F_2	-	x	x	-		1/2	1/2
$F_1 \cap F_2$	-	-	x	-		0/1	0/1
$F_1 \cup F_2$	x	x	x	-		2/3	1

Beispiel 2: Das Beispiel zeigt, dass bei Verwendung von $\text{Comb}_\cap$ die Precision steigt (bei unverändertem Recall) und bei Verwendung von $\text{Comb}_\cup$ die Precision sinkt (bei unverändertem Recall).

DokumentID	1	2	3	4			
Relevanz	Rel	Rel	Irrel	Irrel		Precision	Recall
F_1	x	-	x	-		1/2	1/2
F_2	x	-	-	x		1/2	1/2
$F_1 \cap F_2$	x	-	-	-		1	1/2
$F_1 \cup F_2$	x	-	x	x		1/3	1/2

Beispiel 3: Das Beispiel zeigt, dass bei Verwendung von $\text{Comb}_\cup$ der Recall steigt (bei unveränderter Precision) sowie bei Verwendung von $\text{Comb}_\cap$ Recall und Precision unverändert bleiben.

DokumentID	1	2	3	4			
Relevanz	Rel	Rel	Irrel	Irrel		Precision	Recall
F_1	x	-	x	-		1/2	1/2
F_2	x	x	x	x		1/2	1
$F_1 \cap F_2$	x	-	x	-		1/2	1/2
$F_1 \cup F_2$	x	x	x	x		1/2	1

□

6.5.2 Kombination numerischer Bewertungen

Liegen die zu kombinierenden Relevanzeinschätzungen in Form von numerischen (d. h. reellwertigen) Bewertungen („Ratings“) vor, so kann das Kombinieren der Bewertungen für ein Dokument durch das arithmetische Verknüpfen der Einzelbewertungen erfolgen. Die verwendeten Algorithmen werden meist als „Comb∗“-Strategien bezeichnet [FS94, KR04, Lee95, Lee97]. Um die von den Filteragenten $\{F_1, \ldots, F_n\}$ stammenden Bewertungen $\{r_{F_1}(d), \ldots, r_{F_n}(d)\}$ zu einer neuen Bewertung zusammenzufassen, wählt man

eine geeignete Verknüpfungsfunktion $\mathrm{V}(r_{F_1}, \ldots, r_{F_n})$ und berechnet damit die Bewertung für das Dokument d:

$$r_{\mathrm{CombV}}\left((r_{F_1}, \ldots, r_{F_n}), d\right) = \mathrm{V}\left(r_{F_1}(d), \ldots, r_{F_n}(d)\right)$$

Häufig verwendete Vertreter der Comb$*$-Strategien sind:

- CombMIN**:** Die Relevanzbewertung des Dokumentes ist das Minimum der Einzelbewertungen:

  $$r_{\mathrm{CombMIN}}\left((F_1, \ldots, F_n), d\right) = \min_{i \in \{1,\ldots,n\}} r_{F_i}(d)$$

  Diese Kombinationsstrategie verfolgt die Idee, dass ein Dokument nur so relevant sein kann, wie es von dem Agenten eingestuft wird, der die niedrigste Bewertung für das Dokument abgibt. Damit kann ein Agent, der das Dokument als irrelevant einstuft, die Relevanzbewertungen aller anderen Agenten überstimmen.

- CombMAX**:** Man berechnet die neue Bewertung des Dokumentes, indem man das Maximum der einzelnen Bewertungen ermittelt:

  $$r_{\mathrm{CombMAX}}\left((F_1, \ldots, F_n), d\right) = \max_{i \in \{1,\ldots,n\}} r_{F_i}(d)$$

  Die Idee dabei ist, dass die Dokumente besonders relevant sind, die mindestens ein Agent als besonders relevant bewertet hat. Dieses Verfahren kann sowohl angewendet werden, wenn die zu kombinierenden Filter auf derselben Dokumentenmenge arbeiten als auch dann wenn die Filter auf unterschiedlichen Dokumentenmengen filtern (Collection Fusion).

- CombSUM**:** Die Relevanzbewertung des Dokumentes ergibt sich als die Summe der Einzelbewertungen:

  $$r_{\mathrm{CombSUM}}\left((F_1, \ldots, F_n), d\right) = \sum_{i=1}^{n} r_{F_i}(d)$$

- CombAVG**:** Die Relevanzbewertung des Dokumentes berechnet sich als der Mittelwert der Einzelbewertungen:

  $$r_{\mathrm{CombAVG}}\left((F_1, \ldots, F_n), d\right) = \frac{1}{n} \sum_{i=1}^{n} r_{F_i}(d)$$

  Die beim Verknüpfen der Einzelbewertungen erhaltene Bewertung ist das arithmetische Mittel der Ratings aller Filteragenten. Jeder Filteragent hat denselben Einfluss auf die abgeleitete Bewertung.

- CombANZ: Die Relevanzbewertung des Dokumentes ergibt sich als die Summe der Einzelbewertungen geteilt durch die Anzahl der Filteragenten, die das Dokument als relevant bewerten:

$$r_{\text{CombANZ}}\left((F_1,\ldots,F_n),d\right) = \frac{1}{m} \sum_{i\in\{1,\ldots,n\},\ r_{F_i}(d)>t} r_{F_i}(d)$$

  Dabei sei m die Anzahl der Filteragenten, deren Bewertung des Dokumentes größer als ein Schwellwert t ist. Diese Kombinationsmethode folgt dem Grundssatz, dass Agenten, die ein Dokument nicht als relevant einschätzen, nicht bei der Berechnung der Relevanz berücksichtigt werden sollten.

- CombMNZ: Die Relevanzbewertung des Dokumentes wird berechnet, indem die Summe der Einzelbewertungen mit der Anzahl der Filteragenten multipliziert, die das Dokument als relevant bewerten:

$$r_{\text{CombMNZ}}\left(F_1,\ldots,F_n\right),d) = m * \sum_{i=1}^{n} r_{F_i}(d)$$

  Dabei bezeichne m die Anzahl der Strategien bei denen die Bewertung größer als eine Schwelle t ist. Diese Strategie verfolgt die Idee, dass ein Dokument dann relevant ist, wenn es sowohl eine hohe Relevanzbewertung erhalten hat als auch von vielen Strategien als relevant bewertet wurde.

Anstatt wie bei den Filterstrategien CombMNZ und CombANZ jeweils Schwellwerte t zu definieren, ab denen Bewertungen einen besondere Gewichtung erhalten, können die Bewertungen auch durch die Bildung eines **f-Mittelwertes** kombiniert werden.

$$r_{\text{CombAVG}}\left((F_1,\ldots,F_n),d\right) = f^{-1}\left(\frac{1}{n}\sum_{i=1}^{n} f\left(r_{F_i}(d)\right)\right)$$

Dabei sei f eine streng monotone stetige (und daher invertierbare) Funktion. Wählt man die Funktion f rechtsgekrümmt, so erhalten kleine Bewertungen einen höheren Einfluss auf den Mittelwert; ist f linksgekrümmt, so wird der Mittelwert stärker durch große Bewertungen bestimmt. Für $f(x) = x$ ergibt sich CombAVG.

Um den Einfluss der Agenten bei Verwendung der Comb$*$-Strategien zu beeinflussen, können die zu kombinierenden Bewertungen unterschiedlich gewichtet werden. Die Gewichtung der Filteragenten wird meist unter Verwendung von Trainingsdaten bezüglich eines gewählten Gütekriteriums (z. B.

MAP) ermittelt [RS03]. Abschnitt 6.6 erläutert das Ermitteln geeigneter Skalierungs- und Gewichtungsstrategien im Detail.

$$r_{\text{CombWeightedV}}\left((F_1,\ldots,F_n),d\right) = V\left(w_1 * r_{F_1}(d),\ldots,w_n * r_{F_n}(d)\right)$$

6.5.3 Kombination rangbasierter Bewertungen

Bewerten Agenten die Relevanz von Dokumenten nicht durch numerische Ratings sondern durch das Sortieren einer Liste von Dokumenten nach Relevanz, spricht man von rangbasierten Bewertungen. Methoden zum Kombinieren von rangbasierten Bewertungen werden unter dem Begriff **Sozialwahlverfahren** (bzw. **Social choice theory**) zusammengefasst. Sozialwahlverfahren führen die Präferenzen mehrerer Personen („Experten“, „Agenten“) zu einer gemeinsamen „kollektiven“ Präferenz zusammen, um eine Entscheidung abzuleiten, die den Nutzen der Gemeinschaft maximiert. Die zwei bekanntesten Social Choice-Strategien sind die **Borda**-Wahl [TS99] und das **Condorcet**-Verfahren [Ris05].

Borda-Wahl

Die Borda-Wahl ist ein nach dem Franzosen Jean-Charles Borda (1733-1799) benanntes Verfahren zur Kombination von Ranglisten. Das Verfahren basiert darauf, dass entsprechend der von den Wählern (in Form einer Rangliste) gegebenen Präferenzen Punkte für jeden Kandidaten vergeben werden. Der Kandidat auf dem letzen Rang erhält null Punkte, der Kandidat auf dem vorletzten Platz einen Punkt usw. Der Kandidat auf dem ersten Platz erhält $n-1$ Punkte (wobei n die Anzahl der Kandidaten bezeichne). Man summiert die durch die Wähler für jeden Kandidaten vergebenen Punkte und sortiert auf Basis der Punktesumme die Kandidaten.

$$\text{points}_{\text{Borda}}(d) = \sum_{i\in\{1,\ldots,n\}} \text{points}(F_i,d)$$

Dabei bezeichne $\text{points}(F_i,d)$ die Anzahl der Punkte, die ein Filteragent F_i für ein Dokument d vergibt. Durch Sortieren der Kandidaten nach den bei der Borda-Wahl ermittelten Punkte $\text{points}_{\text{Borda}}(d)$ wird das Ergebnisranking abgeleitet.

Um bei der Borda-Wahl Gewichtungen für Wähler zu berücksichtigen, modifiziert man das Verfahren zur Berechnung der Punkte, die sich aus den Präferenzen der Wähler ergeben. Dazu multipliziert man die Anzahl der

Punkte, die sich aus dem Rangplatz des Wählers ergeben mit der Gewichtung w_i des Wählers F_i.

$$\text{points}_{\text{weightedBorda}}(d) = \sum_{i \in \{1,\ldots,n\}} w_i * \text{points}(F_i, d)$$

Die Borda-Wahl bietet für taktische Wähler verschiedene Schwachstellen. Besonders empfindlich ist das Verfahren gegen eine Manipulation der Kandidaten-Aufstellung: Durch das Duplizieren des Kandidaten, der von einem Wähler als besonders gut eingestuft wird, kann anderen Kandidaten ein Vorteil verschafft werden. Im Bereich der agentenbasierten Informationssysteme bedeutet dies, dass durch das Hinzufügen von Duplikaten zur Menge der potentiell relevanten Dokumente, die Wahl beeinflusst werden kann.

Ein Vorteil der Borda-Wahl ist, dass diese einfach und effizient implementiert werden kann und daher auch bei einer großen Anzahl von Filteragenten und Dokumenten durchführbar ist (Aufwandsklasse $\mathcal{O}(n * |D|)$).

Condorcet-Methode

Eine Alternative zur Borda-Wahl bietet die Condorcet-Methode, die 1785 vom Marquis de Condorcet in ihren Grundzügen beschrieben wurde[1].

Die Condorcet-Methode analysiert für jeden Wähler F_i die Präferenzen für jedes Paar von zwei Kandidaten (d_i, d_j). Ein Kandidat d_i wird gegenüber einem Kandidaten d_j bevorzugt, falls es mehr Wähler gibt, die d_i gegenüber d_j vorziehen, als Wähler, die d_j gegenüber d_i favorisieren. Den Kandidaten, der gegenüber allen anderen Kandidaten vorgezogen wird, nennt man Condorcet-Sieger. Die Condorcet-Methode kann auf verschiedene Arten implementiert werden. Die bekanntesten Arten der Condorcet-Methode sind die Kemeny-Young-Methode, das Ranked Pairs-Verfahren und die Schulze-Methode.

Im Gegensatz zur Borda-Wahl ist die Condorcet-Methode nicht so anfällig für Manipulationen (z. B. das Duplizieren von Kandidaten). Allerdings ist die Condorcet-Methode nur mit deutlich höheren Aufwand berechenbar (Aufwandsklasse $\mathcal{O}\left(n * |D|^2\right) + \mathcal{O}\left(|D|^3\right)$). Da beim Dokumenten-Retrieval die Anzahl der Dokumente (d. h. Kandidaten) oft groß ist (im Gegensatz zum Szenario der Wahl eines Expertengremiums, bei dem die Anzahl der Kandidaten klein, die Anzahl der Wähler aber sehr groß ist), kann die Condorcet-Methode im Allgemeinen nur in nichtzeitkritischen Szenarien eingesetzt werden.

[1] Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix - Essai sur la constitution et les fonctions des assemblées provinciales - Sur les élections

Lernen von Präferenzen

Die Condorcet-Methode setzt voraus, dass jeder Wähler die Kandidaten in einer Rangliste ordnet. In Systemen, die Benutzerfeedback analysieren, können Bewertungen und Ranglisten unvollständig sein oder es können gleichwertig bewertete Dokumente vorliegen. Um auch auf Basis partiell verfügbarer binärer Relevanzbewertungen ein Dokumentenranking ermitteln zu können, setzt man Verfahren des maschinellen Lernens ein.

Der Algorithmus RankSVM [HGO00] verwendet zum Lernen eine Support Vector Maschine, wobei als Fehlermaß die Kreuzentropie zum Einsatz kommt. Eine Optimierung des Ansatzes für das IR verwendet der Algorithmus ListNet [CQL+07], der als Optimierungskriterien ein Gütefunktional verwendet, das die Entropie von Listen beschreibt.

Der Algorithmus FRank [TLQ+07] kombiniert mehrere Modelle zum Lernen der Relevanzbeziehungen („Boosting“) und kann dabei vor allem bei verrauschten Daten (im Vergleich zu RankSVM [HGO00]) gute Ergebnisse erreichen.

Das Lernen von Präferenzrelationen hat den Vorteil, dass kein Skalieren von Bewertungen notwendig ist und dass nur partiell geordnete Dokumentenpaare benötigt werden. Allerdings müssen beim Zusammenfügen größerer geordneter Ergebnismengen sehr viele Dokumentenpaare berücksichtigt werden, was mit einem großen Ressourcenbedarf verbunden ist. Aufgrund der verwendeten Neuronalen Netze bzw. der jeweils verwendeten Strategien zur Auswahl der Trainingsdaten schwankt die erreichte Filterqualität zwischen verschiedenen Trainingsläufen. Die Verfahren zum Lernen von Präferenzrelationen sollten vor allem dann eingesetzt werden, wenn partiell bewertete Trainingsdaten zur Verfügung stehen und hinreichend Ressourcen zum Trainieren verfügbar sind.

6.5.4 Wahl der Kombinationsstrategie

Soll für eine Suchanfrage eine Ergebnismenge zusammengestellt werden, so muss jeweils ein geeignetes Verfahren zum Zusammenfassen von Bewertungen ausgewählt werden. Faktoren, anhand derer eine Strategie zum Kombinieren von Bewertungen ausgewählt werden kann, werden im Folgenden diskutiert.

Die zu kombinierenden Typen von Bewertungen: In offenen Filterframeworks tritt das Problem auf, dass unterschiedliche Bewertungsstrategien und somit auch unterschiedliche Arten von Bewertungen miteinander kombiniert werden sollen. Um die unterschiedlichen Typen von Bewertungen („mengenbasiert“, „rankingbasiert“, „ratingbasiert“) zu kombinieren, müssen die Bewertungen jeweils an den Kombinationsalgorithmus angepasst werden

(wenn sie nicht bereits im gewünschten Bewertungsformat vorliegen). Verfahren zum Konvertieren von Bewertungen werden im Abschnitt 6.6.1 diskutiert.

Aufwand der Kombination: Die Methoden zum Kombinieren von Bewertungen sind jeweils mit unterschiedlich hohem Aufwand verbunden. Es muss daher in Abhängigkeit vom jeweiligen Anwendungsszenario geprüft werden, welcher Algorithmus mit den verfügbaren Ressourcen innerhalb der gewünschten Bearbeitungszeit ausgeführt werden kann.

Kombination von Strategien versus komplexe Strategien: Analysiert man das Verknüpfen von Filterstrategien, so stellt sich die Frage, ob anstatt der Kombination von verschiedenen Filtern komplexere Filter verwendet werden sollten. Für die Verwendung einer komplexen Strategie spricht vor allem die dadurch erreichte Reduzierung des Koordinations- und Administrationsaufwandes. Die Kombination mehrerer Filterstrategien ermöglicht dagegen eine höhere Flexibilität („Austauschen von einzelnen Filterstrategien", „einfachere Anpassung an sich verändernde Anforderungen durch das Einbeziehen neuer Filterstrategien", „szenariobezogene Gewichtung von Filterstrategien"). Das Kombinieren mehrere Filterstrategien folgt dem Grundsatz des Heranziehens mehrerer Expertenmeinungen anstatt sich auf die Meinung eines universellen Experten zu verlassen: Auf eine begrenzten Bereich spezialisierte Filterstrategien sind im Allgemeinen weniger komplex und damit einfacher zu implementieren und zu aktualisieren.

6.6 Wie passt man die Kombinationsstrategien für das IR an?

Nachdem die Grundlagen der Filterkombinationsverfahren erläutert wurden, stellt der folgende Abschnitt Verfahren zur Anpassung von Relevanzbewertungen sowie zum Ermitteln von Gewichtungen vor.

6.6.1 Konvertieren von Bewertungen

Anhand der an moderne Informationssysteme gestellten Anforderungen (vgl. Kapitel 2) müssen in einem personalisierte Informationssystem verschiedene Filterstrategien verknüpft werden. Darüber hinaus sollte das System offen für neue Bewertungsstrategien sein. Um auch unterschiedliche Typen von Relevanzbewertungen (z. B. ratingbasierte und rangbasierte Bewertungen) miteinander kombinieren zu können und die Anwendung einer großen Anzahl von Kombinationsalgorithmen zu ermöglichen, müssen die Relevanzbewertungen jeweils geeignet an das ausgewählte Kombinationsverfahren angepasst werden [Lee95, LXQ+07, BK07a].

Konvertierung numerischer Bewertungen

Um numerische Bewertungen auch auf Basis von mengenbasierten Filterkombinationsverfahren verwenden zu können, wandelt man die numerischen (d. h. reellwertigen) Bewertungen in boolesche Bewertungen um. Man wählt dazu einen Schwellwert t, der angibt, ob eine Bewertung ein Dokument als relevant oder irrelevant einstuft. Den Schwellwert wählt man dabei anhand von Wissen über die jeweilige Filterstrategie oder anhand einer Analyse der Bewertungsverteilung.

$$\mathrm{Rel}_{F_{bool}}(d) = \begin{cases} 1, & r_{F_{rating}}(d) \geq t \\ 0, & r_{F_{rating}}(d) < t \end{cases}$$

Um eine Dokumentenmenge, bei der die Relevanz der Dokumente jeweils durch eine numerische Bewertung („Rating“) beschrieben wird, in eine sortierte Dokumentenliste („rangbasierte Relevanzbewertung“) zu überführen (z. B. um Sozialwahlverfahren zum Kombinieren von Bewertungen einsetzen zu können), sortiert man die Dokumente nach ihrer Bewertung und leitet auf Basis dieser Sortierung eine Rangfolge der Dokumente ab.

Beim Konvertieren von numerischen Relevanzbewertungen in boolesche oder rangbasierte Bewertungen gehen irreversibel Informationen verloren, was zu einer Reduzierung der Filterqualität führen kann.

Konvertierung rangbasierter Bewertungen

Um rangbasierte Bewertungen bei der Verwendung von numerischen Bewertungskombinationsverfahren (z. B. Comb$*$) berücksichtigen zu können, leitet man aus dem Rang eines Dokumentes ein Rating ab. Um eine Gleichverteilung der ermittelten Ratings zu erhalten, weist man einem Dokument mit dem Rang $\mathrm{rank}(d) \in \{1, \ldots, n\}$ die Bewertung $\mathrm{r}(d) = 1 - \frac{\mathrm{rank}(d)}{n}$ zu.

Eine Konvertierung von rangbasierten Bewertungen in boolesche Bewertungen ist im Allgemeinen nicht sinnvoll und sollte nur in Ausnahmefällen verwendet werden.

Konvertierung mengenbasierter Filtermengen

Die Umwandlung von rangbasierten oder ratingbasierten Bewertungen in mengenbasierte Bewertungen wird selten verwendet, da dabei viele Informationen verloren gehen. Ist in Ausnahmefällen eine Umwandlung von mengenbasierten Bewertungen in ratingbasierte Bewertungen notwendig, weist man meist den „relevant“ eingestuften Dokumente die Bewertung 1.0 und den „irrelevant“ eingestuften Dokumenten die Bewertung 0.0 zu.

Eine Konvertierung von mengenbasierten Bewertungen in rangbasierte Bewertungen ist im Allgemeinen nichtdeterministisch, da innerhalb der relevanten (bzw. irrelevanten) Dokumente ohne zusätzliches Wissen keine Sortierung existiert.

6.6.2 Skalierung und Gewichtung

Sollen die von unterschiedlichen Filteragenten stammenden Bewertungen kombiniert werden, so muss für jeden Agenten eine geeignete Gewichtung ermittelt werden. Bei der Kombination von ratingbasierten Filterstrategien muss außerdem eine passende Skalierung für die Bewertungen gewählt werden.

Der Begriff **Skalierung** bezeichnet das Vorgehen, mit dem die reellwertige Relevanzbewertungen untereinander vergleichbar gemacht werden. Bei booleschen und rangbasierten Strategien ist eine Skalierung nicht erforderlich. Die **Gewichtung** eines Filteragenten in einem Gremium beschreibt, wie groß der Einfluss eines Agenten innerhalb des Agentengremiums ist.

Skalierung bei der Kombination von Ratings

Um die Bewertungen von unterschiedlichen Filterstrategien untereinander vergleichbar zu machen, werden die Ratings skaliert:

$$r_{\text{skaliert}}(d) = \frac{\mathrm{r}(d) - a}{b}$$

Dabei seien a und b zwei Skalare, die in Abhängigkeit von der Verteilung der Ratings R geeignet wählt werden. Nimmt man eine Gleichverteilung der Bewertungen auf einem Intervall $[\min_{r\in R}(r), \max_{r\in R}(r)]$ an, so wählt man $a = \min_{r\in R}(r)$ und $b = \max_{r\in R}(r) - a$. Unterstellt man eine Normalverteilung der Bewertungen R so wählt man

$$a = E(R) = \frac{\sum_{r\in R} r}{|R|}, \quad b = \sigma(R) = \sqrt{E\left((R - E(R))^2\right)}$$

Unterscheiden sich die Verteilungen der Bewertungen der einzelnen Filter sehr stark, so ist die Annahme, dass die Bewertungen der Filterstrategien derselben Verteilung folgen, problematisch. In diesem Fall kann es sinnvoll sein, durch ein verteilungsspezifisches Skalieren eine einheitliche Verteilung (z. B. Normalverteilung) der Relevanzbewertungen zu erreichen.

Wahl der Gewichtungen

Bei der Wahl der Gewichtungen fordert man meist, dass die Summe der Gewichte eins ist und alle Gewichte nicht negativ sind. Kann durch die Verwendung negativer Gewichtungen eine Verbesserung der Filterqualität erreicht werden, deutet dies darauf hin, dass entweder ein Agent fehlerhaft arbeitet (d. h. im Durchschnitt relevante Dokumente schlechter bewertet als irrelevante Dokumente) oder dass ein Agent mit ähnlicher Strategie („abhängige Strategie“) eine zu hohe Gewichtung besitzt.

Darüber hinaus ist bei der Interpretation von Gewichtungen zu beachten, dass bei Verwendung einer alternativen Skalierungsmethode auch die Gewichtungen angepasst werden müssen. Bei Verzicht auf eine explizite Skalierung kann die Anpassung der Größenordnung der Bewertungen auch durch eine geeignete Wahl der Gewichtungen erreicht werden.

Mit welchen Gewichtungen in einem Gremium von Filteragenten optimale Ergebnisse erreicht werden können, hängt von verschiedenen Faktoren ab (z. B. Güte der einzelnen Filter, Ähnlichkeiten zwischen den Filterstrategien). Algorithmen zum Ermitteln geeigneter Gewichtungen werden im Abschnitt 7 ausführlich vorgestellt.

6.6.3 Auswählen von Filteragenten

Beim Zusammenstellen eines Gremiums von Filteragenten muss vor allem festgelegt werden, welche Typen von Bewertungsstrategien berücksichtigt und wie viele Agenten in ein Gremium aufgenommen werden sollten.

In der Literatur findet man beim Zusammenstellen von Gremien aus Filteragenten unterschiedliche Ansätze: Einige Autoren [KWSD00] konzentrieren sich darauf, vor allem solche Filterstrategien miteinander zu kombinieren, die auf demselben Algorithmus basieren, aber unterschiedliche Konfigurationen verwenden. Dieser Ansatz erscheint vor allem dann sinnvoll, wenn sich der gewählte Algorithmus bezüglich der Problemstellung als effektiv erwiesen hat, eine für alle relevanten Filteranfragen geeignete Parameterkonfiguration aber nicht ermittelt werden kann. Die Forschungsarbeiten von Liu et al.[WLP04] und Melville et al.[MM03] zeigen, dass zum Erreichen hochqualitativer Filterergebnisse verschiedenartige Filteralgorithmen miteinander kombiniert werden sollten. Zum Auswählen der zu kombinierenden Strategien benötigt man folglich ein Maß, das ein Berechnen der Ähnlichkeit zweier Filterstrategien ermöglicht.

Ähnlichkeitsmaße für Filterstrategien

Im folgenden Abschnitt wird gezeigt, wie auf Basis der Bewertungen der einzelnen Filteragenten Ähnlichkeitsmaße definiert werden können (wobei die Agenten selbst als „black box“ angesehen werden). Die durch die Analyse der Ähnlichkeit von Filteragenten gewonnenen Informationen helfen, effektiv Agentengremien zusammenzustellen.

Für die Berechnung der Korrelation zwischen zwei Filterstrategien werden zwei unterschiedliche Herangehensweisen untersucht:

- **Korrelation der Ergebnisqualität auf Basis von Benchmarkdatensätzen:** Stehen Benchmarkdaten für das Ermitteln der Güte der Filterstrategien zur Verfügung, so kann die Ähnlichkeit zwischen Filterstrategien anhand der für eine Anfrage erreichten Filterqualität ermittelt werden.

- **Korrelation der Bewertungen:** Stehen keine Benchmark-Daten zur Verfügung, so kann die Korrelation von Filteragenten mit Hilfe des Korrelationskoeffizienten bzw. des Rangkorrelationskoeffizienten auf den Filterergebnismengen berechnet werden.

Um die Korrelation von Filterstrategien zu ermitteln, die die Relevanz von Dokumenten mit numerischen Bewertungen beurteilen, verwendet man den **Pearson-Korrelationskoeffizienten** („Produkt-Moment-Korrelation“). Der Korrelationskoeffizient ist ein dimensionsloses Maß für den Grad des linearen Zusammenhangs zwischen zwei mindestens intervallskalierten Merkmalen, das Werte zwischen -1 und $+1$ annehmen kann. Ein Wert von $+1$ (bzw. -1) zeigt einen vollständig positiven (bzw. negativen) linearen Zusammenhang der betrachteten Merkmale an. Ein Korrelationskoeffizient mit dem Wert 0 besagt, dass die beiden Merkmale nicht *linear* voneinander abhängen.

Sind für zwei Zufallsvariablen X, Y die Messergebnisse (z. B. die für die Dokumente berechneten Bewertungen) $x_1, x_2, \ldots, x_n$ und $y_1, y_2, \ldots, y_n$ bekannt, so berechnet sich der empirische Korrelationskoeffizient (nach Pearson) folgendermaßen:

$$\rho_{\text{Pearson}}(X,Y) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})\,(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{n} (y_i - \bar{y})^2}}$$

Dabei sind $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ und $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ die empirischen ermittelten Erwartungswerte der vorliegenden Daten für X und Y.

Für die Berechnung der Ähnlichkeit sortierter Ergebnismengen setzt man den **Rangkorrelationskoeffizienten** ein. Der Rangkorrelationskoeffizient

gibt an, wie gut mit Hilfe einer monotonen Funktion der Zusammenhang zwischen zwei Variablen (hier Filterergebnismengen) beschrieben werden kann. Bei der Berechnung werden keine Annahmen über die Wahrscheinlichkeitsverteilung der Variablen oder über die Art des Zusammenhanges der Variablen benötigt. Der Rangkorrelationskoeffizient ist robust gegenüber „Ausreißern“ und kann auch für die Beschreibung von nichtlinearen Zusammenhängen verwendet werden. Die beiden am häufigsten verwendeten Verfahren zur Berechnung der Rangkorrelation sind der **Spearman-** und der **Kendalls-Tau-**Rangkorrelationskoeffizient.

Der **Spearman-Rangkorrelationskoeffizient** berechnet sich analog zum Pearson-Korrelationskoeffizienten. Es wird lediglich anstatt des Ratings x_i der Rang $\mathrm{rank}(x_i)$ verwendet.

$$\rho_{\mathrm{Spearman}}(X,Y) = \frac{\sum_{i=1}^{n} \left(\mathrm{rank}(x_i) - \overline{rank_x}\right)\left(\mathrm{rank}(y_i) - \overline{rank_y}\right)}{\sqrt{\sum_{i=1}^{n} \left(\mathrm{rank}(x_i) - \overline{rank_x}\right)^2}\sqrt{\sum_{i=1}^{n} \left(\mathrm{rank}(y_i) - \overline{rank_y}\right)^2}}$$

Eine Alternative zur Berechnung der Korrelation zweier sortierten Dokumentenmengen stellt der Rangkorrelationskoeffizient **Kendalls Tau** dar: Im Gegensatz zum Spearman-Korrelationskoeffizienten, der die Differenz zwischen den jeweils vergebenen Rängen berücksichtigt, unterscheidet Kendalls Tau nur die drei Relationen „höherer Rang“, „gleicher Rang“ und „niedrigerer Rang“. Kendalls Tau sollte dem Spearman-Korrelationskoeffizienten vorgezogen werden, wenn die Daten nicht normalverteilt sind, ungleiche Skalen aufweisen oder die Stichproben klein sind.

Zur Berechnung von Kendalls Tau auf einer jeweils durch die zwei Filteragenten F_1 und F_2 sortierten Dokumentenmenge D (mit n Elementen) betrachtet man die $\frac{n}{2}$ $(n-1)$ möglichen Dokumentenpaare (d_i, d_j) (wobei $i, j \in [1, \ldots, n], i \neq j$): Es bezeichne:

- P_{einig} die Menge der Dokumentenpaare für die gilt: $F_1(d_i) > F_1(d_j) \wedge F_2(d_i) > F_2(d_j)$ oder $F_1(d_i) < F_1(d_j) \wedge F_2(d_i) < F_2(d_j)$

- P_{uneinig} die Menge der Dokumentenpaare für die gilt: $F_1(d_i) > F_1(d_j) \wedge F_2(d_i) < F_2(d_j)$ oder $F_1(d_i) < F_1(d_j) \wedge F_2(d_i) > F_2(d_j)$

- $P_{1\mathrm{gleich}}$ die Menge der Dokumentenpaare für die gilt: $F_1(d_i) = F_1(d_j) \wedge F_2(d_i) \neq F_2(d_j)$

- $P_{2\mathrm{gleich}}$ die Menge der Dokumentenpaare für die gilt: $F_2(d_i) = F_2(d_j) \wedge F_1(d_i) \neq F_1(d_j)$

Anhand der Anzahl der Dokumentenpaare berechnet man Kendalls Tau:

$$\tau = \frac{|P_{\text{einig}}| - |P_{\text{uneinig}}|}{\sqrt{|P_{\text{einig}}| + |P_{\text{uneinig}}| + |P_{\text{1gleich}}|}\sqrt{|P_{\text{einig}}| + |P_{\text{uneinig}}| + |P_{\text{2gleich}}|}}$$

Die Maße zur Ähnlichkeit von Filterstrategien können genutzt werden, um die Filterstrategien zu gruppieren und die Diversität der in einem Gremium von Filteragenten vertretenen Strategien zu steuern.

Gruppieren von Filterstrategien

Um Filterstrategien auf Basis ihrer Ähnlichkeit zu gruppieren, können Verfahren des unüberwachten Lernens eingesetzt werden, wie z. B. **Clustering** [Bis06] oder **Self Organizing Maps** (SOMs) [RMS91]. Ein Algorithmus, der zum Clustern von Filterstrategien eingesetzt werden kann, ist das hierarchische Clustern auf Basis von Ähnlichkeitsmatrizen. Die erhaltenen Cluster können in einem baumförmigen Graphen („Dendrogramm“) repräsentiert werden, wobei jeweils verschiedene Cluster ineinander geschachtelt werden. Zum hierarchischen Clustern können zwei verschiedene Ansätze verfolgt werden [Hau04]:

- **Agglomerative Verfahren:** Man beginnt mit n Clustern (jedes Objekt bildet seinen eigenen Cluster) und fasst iterativ jeweils die sich untereinander ähnlichesten Cluster zusammen, bis alle Objekte zum selben Cluster gehören.

- **Partitionierende Verfahren:** Man beginnt mit einem Cluster, dem alle Objekte zugeordnet sind. Man teilt den Cluster so, dass die Heterogenität innerhalb des Clusters minimiert wird. Man setzt dieses Vorgehen iterativ fort, bis alle Cluster nur ein Objekt enthalten.

Agglomerative Verfahren liefern im Gegensatz zu partitionierenden Verfahren etwas weniger robuste Ergebnisse. Oft werden trotzdem agglomerative Clusteralgorithmen verwendet, da diese effizienter berechnet werden können [JMF99].

Algorithmus zum agglomerativen Clustern Zum agglomerativen hierarchischen Clustern verwendet man die folgenden Schritte:

1. Man berechnet die Distanz zwischen allen Objekten und erstellt eine Ähnlichkeitsmatrix. Das Definieren der Ähnlichkeit zwischen zwei Objekten kann auf Basis der Korrelation von Objekten erfolgen.

2. Man ermittelt die Cluster, die den geringsten Abstand von einander haben.

3. Man fasst die ausgewählten Cluster zusammen und berechnet die Distanzen zwischen den Clustern neu.

4. Die Schritte 2 und 3 werden iterativ solange ausgeführt, bis alle Objekte zum selben Cluster gehören.

Zum Berechnen der Distanz zwischen zwei Clustern können verschiedene Methoden zum Einsatz kommen.

- Nächste Nachbarn: Die Distanz zwischen zwei Clustern ergibt sich als der kürzeste Abstand von zwei Objekten unterschiedlicher Cluster. Beim Clusterbildungsprozess werden dabei bevorzugt die Objekte zu einem Cluster zusammengefasst, die aneinandergrenzen oder durch dazwischen liegende Objekte verknüpft sind.

- Entfernteste Nachbarn: Der Abstand zwischen zwei Clustern ergibt sich als der maximale Abstand von zwei Objekten der betrachteten Cluster. Das Verfahren wird verwendet, wenn Cluster mit geringem Durchmesser angestrebt werden.

- Durchschnittliche Distanz: Einen Mittelweg (zwischen den Ansätzen „Nächste Nachbarn“ und „Entfernteste Nachbarn“) bietet die Methode durchschnittliche Distanz. Der Abstand zwischen zwei Clustern wird dabei als der (ggf. gewichtete) Mittelwert der Distanzen aller Objektpaare der Cluster berechnet.

Der Vorteil von hierarchischen Clusteralgorithmen im Vergleich zu alternativen Algorithmen (wie **Expectation Maximation** [DLR77] oder ***K*-Means-Clustering** [Bis06, HBV01]) besteht darin, dass die zu gruppierenden Objekte nicht durch Merkmalsvektoren beschrieben werden müssen und kein Algorithmus zum Berechnen des Clustermittelpunktes benötigt wird. Es müssen beim hierarchischen Clustern keine zusätzlichen Parameter gewählt werden und es werden keine Informationen über die Struktur der Objekte benötigt.

Zusammenstellen von Gremien auf Basis von Clustern

Strategien zum Gruppieren von Filteragenten werden vor allem dann eingesetzt, wenn aus einer sehr großen Anzahl von Filterstrategien ein aus wenigen Agenten zusammengesetztes Gremium ermittelt werden soll (z. B. bei einer

begrenzten Verfügbarkeit von Ressourcen). Ein Leitgedanke dabei ist, dass ähnliche Strategien identifiziert und zusammengefasst werden, wodurch sich der Aufwand zum Zusammenstellen eines geeigneten Gremiums von Filteragenten reduziert.

Die beim Clustern von Filteragenten gewonnenen Informationen können genutzt werden, um effizient Gremien zusammenzustellen.

- **Berücksichtigen unterschiedlicher Filterstrategien:** Um die Diversität der in einem Gremium von Filteragenten berücksichtigen Strategien zu maximieren, wählt man aus jedem Cluster jeweils einen Filteragenten aus. Zu berücksichtigen ist, dass hierbei auch Agenten von Clustern berücksichtigt werden, die keine hochqualitativen Ergebnisse liefern, was zu einer Senkung der durch das Gremium von Filteragenten ermittelten Ergebnisse führen kann.

- **Kombinieren leistungsfähiger, ähnlicher Filteragenten:** Gruppiert man die Filterstrategien nach der Ergebnisqualität oder nach den (für gleiche Anfragen) ermittelten Dokumentensortierungen, so werden meist die besonders leistungsfähigen Filteragenten in einem Cluster und die „schlechten“ Filteragenten in anderen Clustern zusammengefasst. Um ein Gremium aus leistungsfähigen Agenten zusammenzustellen wählt man die Agenten bevorzugt aus dem Cluster, der im Durchschnitt die besten Agenten enthält [KWSD00].

- **Weitere Strategien** zum Auswählen von Filterstrategien auf Basis von Clustern sind: Bevorzugung von Clustern mit vielen Elementen („populäre Filterstrategien“), Vermeiden von Clustern mit sehr vielen Elementen („Verdacht des Manipulationsversuchs durch das Duplizieren von Agenten“) und Auswahl der Filterstrategien proportional zur Clustergröße.

Wie viele Strategien sollten untereinander kombiniert werden?

Neben der Fragestellung, welche Agenten verknüpft werden sollten, muss ermittelt werden, aus wie vielen Filteragenten ein leistungsfähiges Gremium bestehen sollte. Bei der Bestimmung sind verschiedene (sich teilweise entgegenstehende) Ziele zu berücksichtigen:

- Um Dokumente zuverlässig zu bewerten, sollten möglichst mehrere Meinungen (d. h. Filteralgorithmen) berücksichtigt werden. Mit dem Einbeziehen mehrerer Filterstrategien wird erreicht, dass alle für die Relevanzbewertung wichtigen Merkmale (Features, Bewertungsmethoden) in die Relevanzberechnung einfließen.

- Das Kombinieren einer großen Anzahl von Agenten erfordert Systemressourcen. Mit einer wachsenden Anzahl von Filteragenten steigt sowohl die von den Filteragenten benötigte Menge an Ressourcen als auch der Aufwand zum Zusammenführen der Bewertungen der einzelnen Filterstrategien. In einem Agentengremium sollten daher nicht zu viele Filterstrategien berücksichtigt werden.

- Um den Schaden zu minimieren, der bei Ausfall eines Filteragenten entsteht, sollte für jeden Filteragenten ein gleichwertiger alternativer Agent zur Verfügung stehen.

Bei der Bestimmung einer geeigneten Kombination von Filterstrategien muss jeweils ein Kompromiss zwischen den verschiedenen Zielen gefunden werden. Die optimale Größe für ein Gremium von Filteragenten hängt dabei vom betrachteten Szenario („Optimierungskriterium“), dem verwendeten Algorithmus zur Kombination der Filterergebnisse sowie von der Art der verfügbaren Filterstrategien ab. Detaillierte Untersuchungen anhand von Benchmarkdatensätzen werden im Kapitel 7 vorgestellt.

6.7 Zusammenfassung

In diesem Kapitel wurden Methoden für das Normalisieren und Kombinieren von Dokumentenbewertungen vorgestellt sowie das Ermitteln geeigneter Gewichtungen für die Filteragenten diskutiert. Es wurde gezeigt, wie boolesche, rangbasierte und ratingbasierte Bewertungen miteinander verknüpft werden können und welche Besonderheiten jeweils beachtet werden müssen. Um zu veranschaulichen, in welchen Szenarien die verschiedenen Strategien besonders leistungsfähig sind, werden die wichtigsten Merkmale der Kombinationsstrategien gegenübergestellt.

Analysiert man die die Stärken und Schwächen der verschiedenen Filterkombinationsverfahren, so lassen sich die folgenden Aussagen ableiten:

- **Die Comb∗-Strategien** sollten verwendet werden, wenn alle Filteragenten numerische Bewertungen zur Verfügung stellen. Gute Ergebnisse können dabei insbesondere unter Verwendung der Strategien CombSUM und CombMNZ erreicht werden [FS94]. Allerdings muss bei der Verwendung der Comb∗-Strategien sichergestellt werden, dass die Bewertungen normalisiert sind. Durch die Wahl geeigneter Gewichtungen für die in einem Gremium vertretenen Agenten kann die Leistungsfähigkeit gegenüber einem ungewichteten Gremium gesteigert werden. Die Comb∗-Strategien sind effizient berechenbar und können

daher auch bei einer großen Anzahl von Dokumenten und Filteragenten eingesetzt werden.

- **Rangbasierte Filterkombinationsstrategien** („Sozialwahlstrategien“) haben gegenüber den Comb∗-Strategien den Vorteil, dass ein Normalisieren von Bewertungen nicht notwendig ist und rangbasierte Bewertungen mit numerischen Bewertungen zusammengefasst werden können. Allerdings ist die Umwandlung von ratingbasierten Bewertungen in rangbasierte Relevanzbewertungen mit einem Informationsverlust verbunden. Bei der rangbasierte Kombination von ratingbasierten Bewertungen werden daher im Allgemeinen (im Vergleich zur Verwendung von CombSUM) schlechtere Ergebnisse erzielt. Soll eine Sozialwahlstrategie zum Zusammenführen von Bewertungen eingesetzt werden, muss beachtet werden, dass sich die Wahlverfahren in Bezug auf die Aufwandsklasse stark unterscheiden. So eignen sich CONDORCET-Verfahren aufgrund der Aufwandsklasse nur für Ergebnismengen mit einer geringen Anzahl von Dokumenten.

- **Mengenbasierte Filterkombinationsstrategien** werden im Allgemeinen nur für ausgewählte Szenarien eingesetzt. So werden boolesche Kombinationsstrategien eingesetzt, um Dokumente herauszufiltern, die vorgegebene Merkmale aufweisen, z. B. zum Herauszufiltern von Spam. Da mengenbasierte Filterkombinationen effizient implementierbar sind, werden diese Strategien häufig zur Vorverarbeitung eingesetzt, z. B. um die Anzahl der Dokumente zu verringern, für die dann auf Basis von Comb∗- oder Sozialwahlverfahren die jeweiligen Bewertungen zusammengefasst werden.

Eine Evaluation anhand von Benchmarkdaten ist im Kapitel 7 zu finden.

Kapitel 7

Algorithmen und Evaluation

7.1 Zusammenstellen von Agentengremien

Nachdem im Kapitel 6 die grundlegenden Algorithmen für die Kombination von Filterstrategien diskutiert wurden, werden im folgenden Kapitel Algorithmen entwickelt, mit denen leistungsfähige Gremien („Ensembles“) von Filteragenten zusammengestellt werden können. Es werden dabei unterschiedliche Funktionale zum Bewerten der Ergebnisqualität betrachtet (z. B. die Präzision und die Verlässlichkeit der Ergebnisse) und diskutiert, mit welchem Aufwand das Zusammenstellen eines Gremiums jeweils verbunden ist. Die Agentengremien werden anhand des LETOR-TD2004-Datensatzes evaluiert.

7.1.1 Analysiertes Szenario

Man betrachtet zunächst das Zusammenstellen eines Gremiums auf Basis einer statischen Menge von Bewertungsagenten. Es wird das Szenario analysiert, dass zu einer Anfrage jeweils eine Menge von potentiell relevanten Dokumenten gegeben ist. Zu jedem (potentiell relevanten) Dokument berechnet jeder Bewertungsagent eine numerische Beurteilung („rating“). Die Aufgabe besteht darin, die zum Bewerten der Dokumente besonders guten Agenten auszuwählen und die Bewertungen unter Verwendung eines geeigneten Algorithmus zu kombinieren.

Für die durchgeführten Untersuchungen wird der LETOR-TD2004-Datensatz (siehe Abschnitt 6.3) verwendet. Dieser Datensatz stellt für jede Anfrage jeweils 1000 Dokumente und die Bewertungen von 44 Agenten zur Verfügung. Die von den einzelnen Filterstrategien ermittelten reellwertigen Bewertungen werden (unter Annahme einer Gleichverteilung der Ratings) jeweils auf das Intervall $[0, 1]$ skaliert. Als Basisstrategie zur Kombination

von Dokumentenbewertungen wird die Strategie CombSUM verwendet, da diese auch bei großen Dokumentenmengen effizient berechnet werden kann und sehr gute Ergebnisse ermöglicht [FS94]. Im Gegensatz zur Kombinationsstrategie CombMNZ muss zudem keine Rating-Schwelle gewählt werden, ab der ein Dokument als relevant eingestuft wird. Als Maß zum Berechnen der Filterpräzision wird vorrangig die Mean Average Precision (MAP) verwendet. Das Maß MAP ist für Szenarien optimiert, bei denen zwischen zwei Relevanzlevel unterschieden wird. Im Gegensatz zum Maß Precision@n muss kein Evaluationspunkt n gewählt werden. Die Evaluation wird auf Basis der im LETOR-Datensatz definierten Kreuzvalidierung (mit fünf vorgegebenen Teilmengen) durchgeführt.

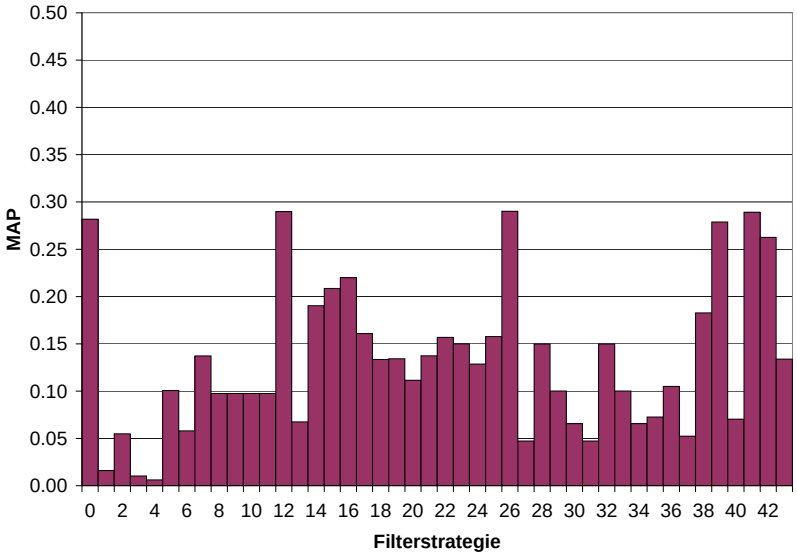


Abbildung 7.1: Das Diagramm visualisiert die Filterpräzision der Strategien des LETOR-TD2004-Datensatzes.

Einen Überblick über die mittlere Precision (MAP) der einzelnen Filterstrategien des LETOR-TD2004-Datensatz gibt die Abbildung 7.1. Die Präzision, die bei der Kombination der Einzelstrategien bei Verwendung unterschiedlicher Kombinationsverfahren erreicht wird, ist in der Abbildung 7.2 dargestellt. Die Graphik zeigt, dass die erreichte mittlere Precision (MAP) stark von der verwendeten Kombinationsstrategie abhängt. Wird die Bewertungen auf Basis der BORDA-Wahl kombiniert, so ist die erreichte mittlere Precision geringer als mittlere Precision des Agenten, der alle Dokumente als relevant bewertet. Unter Verwendung der SCHULZE-Methode wird dagegen

eine mittlere Precision (MAP) erreicht, die über der Precision jeder einzelnen Strategie des LETOR-TD2004-Datensatzes liegt.

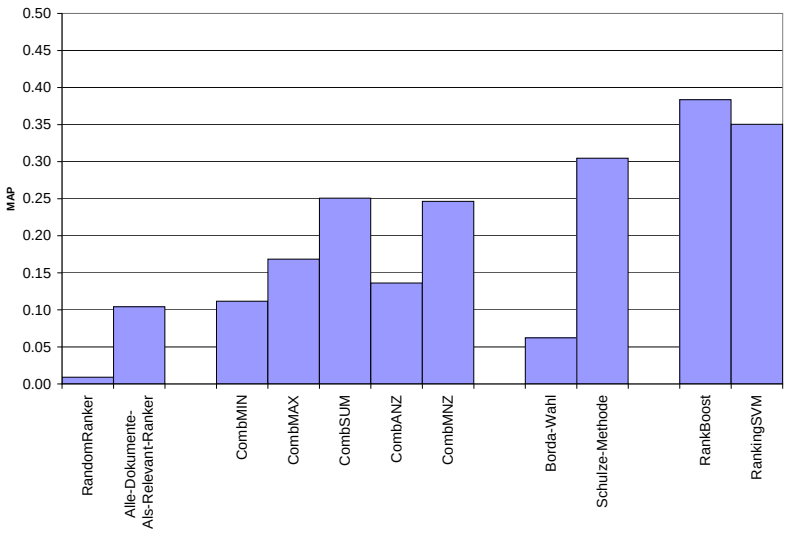


Abbildung 7.2: Die Graphik zeigt die mittlere Precision (MAP) für verschiedene Kombinationsstrategien. Es wurden jeweils alle 44 Strategien des LETOR-TD2004-Datensatzes (mit derselben Gewichtung) berücksichtigt. Zum Vergleich ist zusätzlich die mittlere Precision für den Filteragenten dargestellt, der die Relevanz von Dokumenten zufällig bewertet, sowie für den Filteragenten, der alle Dokumente als relevant bewertet. Außerdem ist die Filterpräzision der als Baseline gegebenen Strategien RankBoost und RankingSVM dargestellt.

7.1.2 Ein Algorithmus zum inkrementellen Zusammenstellen eines Gremiums

Zunächst wird das inkrementelle Zusammenstellen eines Gremiums von Filteragenten betrachtet. Das schrittweise Erstellen eines Gremiums ermöglicht es, sowohl das Zusammenwirken der Agenten als auch den Einfluss der Größe des Agentengremiums auf die Filterpräzision zu analysieren.

Beim inkrementellen Zusammenstellen von Agentengremien beginnt man mit einem „leeren“ (d. h. aus null Agenten bestehenden) Gremium und fügt schrittweise jeweils den Agenten hinzu, durch den die Leistungsfähigkeit des Agentenensembles am stärksten verbessert werden kann. Das Erweitern des

Agentengremiums wird solange fortgesetzt, bis alle Filteragenten zum Gremium gehören. Innerhalb des Gremiums erhalten alle Agenten dieselbe Gewichtung.

Algorithmus 1 Inkrementelles Erstellen eines Agentengremiums

$G \leftarrow \emptyset$
while $A \neq \emptyset$ **do**
  Ermittle $s \in A$, so dass $\mathrm{P}(\mathrm{Comb}(G \cup s)) \geq \mathrm{P}(\mathrm{Comb}(F \cup f_i))\ \forall f_i \in A$
  $G \leftarrow G \cup s$,
  $A \leftarrow A \backslash s$
end while

Eine formale Beschreibung des Vorgehens definiert Algorithmus 1. Dabei bezeichne A die Gesamtheit der verfügbaren Filteragenten und G die im Gremium vertretenen Agenten. Weiterhin sei $\mathrm{Comb}(F)$ eine Strategie zum Kombinieren der Bewertungen, die von den Agenten F ermittelt wurden. $\mathrm{P}(f, D)$ definiere die Güte eines Filters f bezüglich der Dokumentenmenge D. Bei der Implementierung des Algorithmus ist zu berücksichtigen, dass die Bestimmung des jeweils in einem Schritt aufzunehmenden Agenten nebenläufig erfolgen kann. Dies ermöglicht es, das Zusammenstellen des Agentengremiums stark zu beschleunigen.

Ergebnisse Die Filterpräzision der mit diesem Verfahren erstellen Gremien ist in der Graphik 7.3 dargestellt. Die Ergebnisse zeigen, dass durch die Kombination mehrerer Bewertungsstrategien die mittlere Precision (MAP) deutlich gesteigert werden kann (im Vergleich zur Verwendung eines einzelnen Agenten). Die mit dem beschriebenen Algorithmus erreichbare Filterqualität liegt dabei leicht oberhalb der im LETOR-TD2004-Datensatz vorgegebenen Baseline-Strategie RankingSVM [Joa02]. Um eine hohe Präzision der Filterung zu erreichen, müssen nur relativ wenige (d. h. ca. 10 der 44 verfügbaren) Agenten bei der Kombination berücksichtigt werden. Ein starkes Erweitern des Agentengremiums verbessert die Filterqualität nicht. Durch das Einbeziehen sehr vieler Agenten (mehr als 20) sinkt die Präzision der durch das Gremium von Filteragenten gebildeten Strategie. Das Berücksichtigen aller 44 Filteragenten lässt die mittlere Präzision sogar unter die Präzision des besten einzelnen Agenten sinken. Dies zeigt, dass nicht jede Erweiterung des Gremiums trotz des damit verbundenen erhöhten Aufwandes, zu einer Verbesserung der Präzision führt. Gute Filterergebnisse können bereits mit relativ kleinen Agentengremien erzielt werden.

Um die Filterstrategien zu ermitteln, die bevorzugt beim Erstellen von Agenten-Ensembles berücksichtigt werden sollten, bestimmt man, zu wie vie-

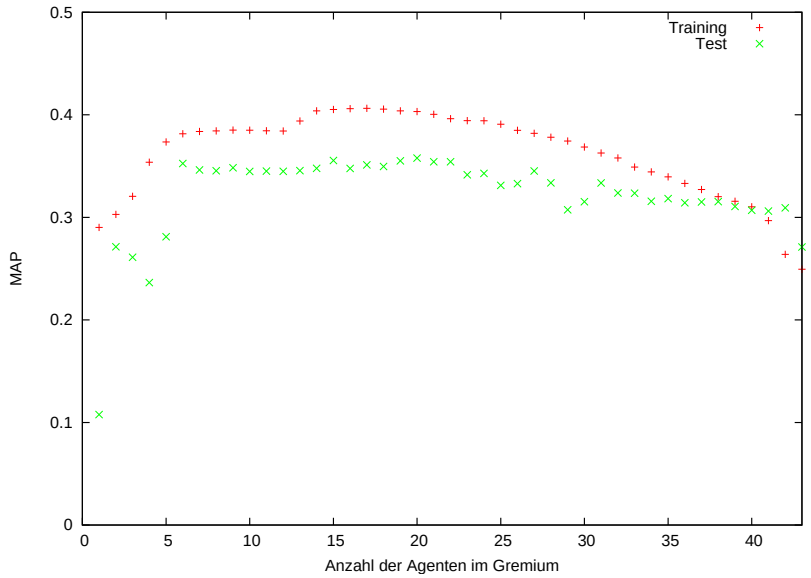


Abbildung 7.3: Die Graphik zeigt die mittlere Precision MAP in Abhängigkeit von der Anzahl der Agenten, deren Filterstrategien unter Verwendung von CombSUM kombiniert werden. Die Agentengremien wurden unter Verwendung von Algorithmus 1 zusammengestellt.

len der erstellten Gremien ein einzelner Filteragent gehört (unter Berücksichtigung einer Kreuzvalidierung mit 5 Teilmengen).

Agenten-ID	7	12	41	26	8	0	9
Verwendungshäufigkeit	90%	86%	86%	85%	84%	82%	81%

Die Tabelle zeigt, dass am häufigsten die Filterstrategie mit der ID '7' (HostRank) verwendet wird. Obwohl diese Strategie einzeln nur eine durchschnittliche Filterpräzision aufweist, können mit dieser Strategie in Kombination mit weiteren Filterstrategien gute Filterergebnisse erreicht werden. Weitere Bewertungskriterien, mit denen in einem Agentengremium eine hohe Filterpräzision erreicht werden kann, sind die Sidemap-basierte Relevanzberechnung (Filteragenten '12' und '26'), die Analyse der Verlinkung eines Dokumentes (Filteragent '41') sowie die Relevanzbewertung auf der Basis einer Termfrequenz-Analyse (Agenten '8' und '0').

Bei der Auswertung ist zu berücksichtigen, dass die Entscheidung, welcher Filteragent in ein Gremium aufgenommen werden sollte, nicht nur von der Leistungsfähigkeit des einzelnen Agenten abhängt, sondern auch davon beeinflusst wird, wie gut ein Agent zu dem bereits bestehenden Ensemble passt.

Zudem haben Agenten, zu denen es eine große Anzahl ähnlicher Filteragenten gibt, eine geringe Wahrscheinlichkeit, in einem Gremium berücksichtigt zu werden.

7.1.3 Agentengremien mit Gewichtungen

Um die Leistungsfähigkeit von Agentengremien zu verbessern, wird jedem Agenten des Ensembles eine Gewichtung zugeordnet. Für das Ermitteln geeigneter Gewichtungen wird (um den erforderlichen Aufwand zu begrenzen) ein an dem Prinzip des Stackings orientierter Algorithmus verwendet. Man bestimmt dazu jeweils die Gewichtung, die ein neu aufzunehmender Filteragent im Verhältnis zum bestehenden Gremium besitzen sollte. Die Relationen zwischen den Gewichtungen innerhalb des bestehenden Gremiums bleiben dabei unverändert.

Zum Zusammenstellen eines gewichteten Gremiums verwendet man den Algorithmus 2. Dabei beschreibe $g = [g_1, g_2, \ldots, g_n]$ die im Gremium berücksichtigten Agenten mit den Gewichten $w = [w_0, w_1, \ldots, w_n]$.

Algorithmus 2 Ermitteln eines gewichteten Expertengremiums

Es sei g ein Vektor der Länge null: $g \leftarrow [\,]$
Es sei w ein Gewichtsvektor der Länge null: $w \leftarrow [\,]$
while $A \neq \emptyset$ **do**
  Ermittle $s \in A$ und eine Gewichtung v, $0 \leq v < 1$,
    so dass P $\left(\text{Comb}_{[w'_0, w'_1, \ldots, w'_n, v]}([g_1, g_2, \ldots, g_n, s])\right)$ maximiert wird.
    Dabei sei $w'_i = \frac{1}{1-v} * w_i$.
  Füge s an den Vektor g an: $g \leftarrow [g_0, g_1, \ldots, g_n, s]$
  Füge v an den Gewichtsvektor w an: $w \leftarrow [w_0, w_1, \ldots, w_n, v]$
  Normiere w: $w \leftarrow \frac{1}{|w|} w$
end while

Ergebnisse Die Filterpräzision der auf Basis von Algorithmus 2 erstellten Gremien zeigt die Abbildung 7.4. Die Ergebnisse zeigen, dass durch die Verwendung von Gewichtungen die Filterpräzision gesteigert werden kann (im Vergleich zu Gremien, in dem alle Agenten die gleiche Gewichtung besitzen). Zudem wird die maximale Precision schon für Gremien mit einer geringeren Anzahl von Filteragenten erreicht. Bei der durchgeführten Untersuchung wurden maximal acht Agenten mit einem Gewicht größer null in das ermittelte Filteragentengremium aufgenommen. Der höhere Aufwand zum Zusammenstellen eines geeigneten Gremiums (beim Berücksichtigen von

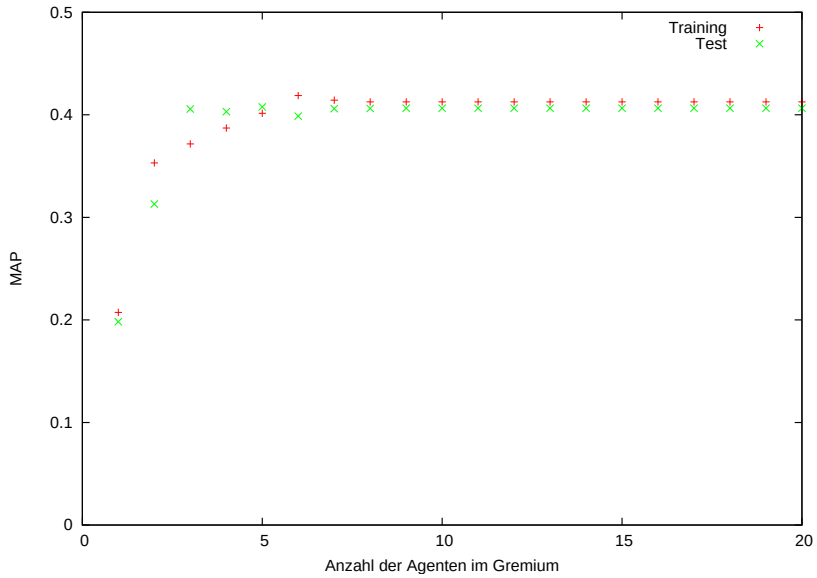


Abbildung 7.4: Die Graphik zeigt die Abhängigkeit der mittleren Precision (MAP) von der Anzahl der berücksichtigten Agenten. Die Kombination der Filterergebnisse erfolgte unter Verwendung der Strategie CombSUM mit Berücksichtigung von Gewichtungen.

Gewichtungen) steht einem reduzierten Aufwand für das Kombinieren von Dokumentenbewertungen gegenüber.

Beim Erstellen eines Gremiums von Filteragenten muss entschieden werden, ab wann keine weiteren Agenten mehr aufgenommen werden (d. h. wie viele Agenten maximal in einem Gremium vertreten sein sollten). Der Algorithmus zum inkrementellen Zusammenstellen eines Gremiums unter Berücksichtigung von Gewichtungen beinhaltet implizit ein Abbruchkriterium. Sobald kein Agent mehr mit einer Gewichtung $v > 0$ ermittelt werden kann, durch den die Leistungsfähigkeit des Gremiums verbessert wird, werden keine weiteren Agenten dem bestehenden Gremium hinzugefügt.

7.1.4 Optimieren von Gewichtungen

Die im vorangehenden Abschnitt vorgestellte Strategie zum Ermitteln der Gewichtungen berücksichtigt nicht, dass durch die Aufnahme eines Agenten in ein Gremium von Filteragenten auch die Gewichte innerhalb des Gremiums angepasst werden sollten. Anstatt jeweils immer nur die optimale Gewichtung für den neu aufzunehmenden Agenten zu ermitteln, können auch die Gewichtungen aller Agenten des Ensembles angepasst werden.

Für die Ermittlung der Gewichte muss ein geeignetes Optimierungsverfahren gewählt werden. Es wurden Verfahren der nichtlinearen Optimierung sowie genetische Algorithmen getestet [HO96]. Die Verwendung eines genetischen Algorithmus zur Optimierung erwies sich bei den durchgeführten Experimenten als etwas schneller und zuverlässiger als eine Optimierung auf Basis des NELDER-MEAD-Algorithmus [LRWW98] und besser als eine Optimierung auf Basis einer stochastischen Suche. Das eingesetzte Optimierungsverfahren (auf Basis genetischer Algorithmen) garantiert keine Verbesserung der initial gewählten Gewichte, führt aber im Allgemeinen nach wenigen Iterationsschritten zu einer geeigneten Anpassung und kann zudem jederzeit abgebrochen werden („Anytime-Fähigkeit").

Algorithmus 3 Ermitteln eines Agentengremiums mit optimierten Gewichtungen

Es sei g ein Vektor der Länge null: $g \leftarrow [\,]$
Es sei w ein Gewichtungsvektor der Länge null: $w \leftarrow [\,]$
while $A \neq \emptyset$ **do**
  Ermittle $s \in A$ und v $(0 \leq v < 1)$,
    so dass P $\left(\text{Comb}_{[w'_0, w'_1, \ldots, w'_n, v]}([g_1, g_2, \ldots, g_n, s])\right)$ maximiert wird.
    Dabei sei $w'_i = \frac{1}{1-v} * w_i$.
  Füge s an den Vektor g an: $g \leftarrow [g_0, g_1, \ldots, g_n, s]$
  Füge v an den Gewichtsvektor w an: $w \leftarrow [w_0, w_1, \ldots, w_n, v]$
  Normiere w: $w \leftarrow \frac{1}{|w|} w$
  Passe den Gewichtsvektor w so an, dass die Güte der Filterergebnisse,
    die man durch Kombination der Filter g erhält, maximiert wird.
end while

Ergebnisse Die Gremien, bei denen die Gewichtungen der Agenten optimiert wurden, weisen annähernd dieselbe Filterpräzision auf, wie die auf Basis des Algorithmus 2 ermittelten Filteragenten-Ensembles. Die Agentengremien, bei denen die Gewichtungen der Agenten optimiert wurden, tendieren dabei zu Overfitting, was an der höheren mittleren Precision (MAP) auf der Trainingsmenge zu erkennen ist, die mit keiner Verbesserung der Filterergebnisse auf der Testmenge verbunden ist. Bei Gremien aus wenigen Agenten kann durch die Optimierung der Gewichtungen eine geringfügige Verbesserung der Leistungsfähigkeit des Gremiums erreicht werden. Der für das Optimieren zusätzlich erforderliche Aufwand ist lohnenswert, wenn (z. B. bei beschränkter Verfügbarkeit von Ressourcen) „kleine" Filteragenten-Gremien zusammengestellt werden sollen.

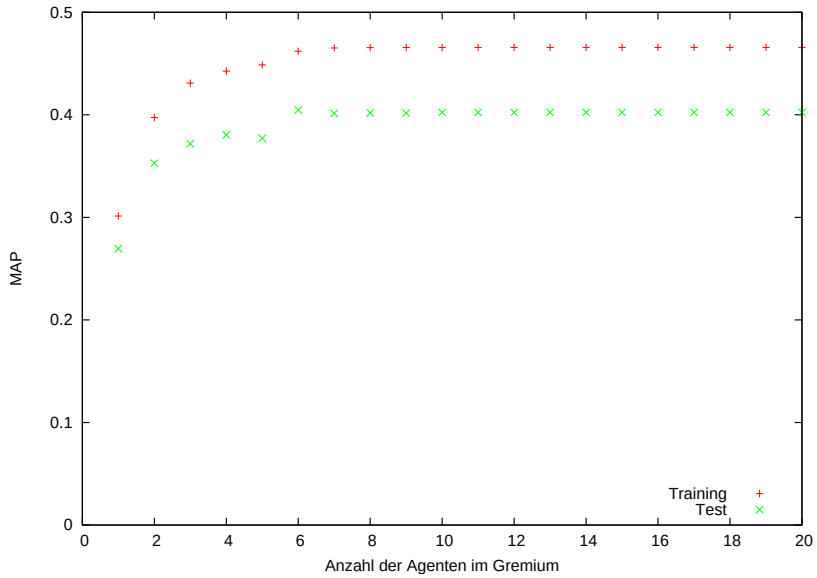


Abbildung 7.5: Die Graphik zeigt die mittlere Precision (MAP) in Abhängigkeit von der Anzahl der berücksichtigten Agenten. Die Kombination der Filterergebnisse erfolgte unter Verwendung von CombSUM, wobei die Gewichtungen mit Hilfe eines genetischen Algorithmus optimiert wurden.

Analyse alternativer Filterkombinationsstrategien

In den vorangehenden Abschnitten wurde jeweils die Kombinationsstrategie CombSUM verwendet. Um die Anwendbarkeit der entwickelten Algorithmen zum Zusammenstellen eines gewichteten Filteragenten-Ensembles zu analysieren, wird untersucht, welche mittlere Precision (MAP) erreicht wird, wenn anstatt von CombSUM die rankingbasierten Strategien Borda-Wahl bzw. die Schulze-Methode eingesetzt werden.

Die Graphik 7.6 zeigt die ermittelte mittlere Precision (MAP) für die erstellten Agentengremien, wobei die Strategien CombSUM, Borda-Wahl und die Schulze-Methode zum Kombinieren von Bewertungen eingesetzt wurden. Die Bestimmung geeigneter Gewichtungen erfolgte auf Basis von Algorithmus 2. Die Ergebnisse zeigen, dass CombSUM deutlich bessere Ergebnisse liefert als die rankingbasierten Strategien. Dies deutet darauf hin, dass der höhere Informationsgehalt von numerischen Bewertungen gegenüber einem Ranking eine höhere Filterpräzision ermöglicht.

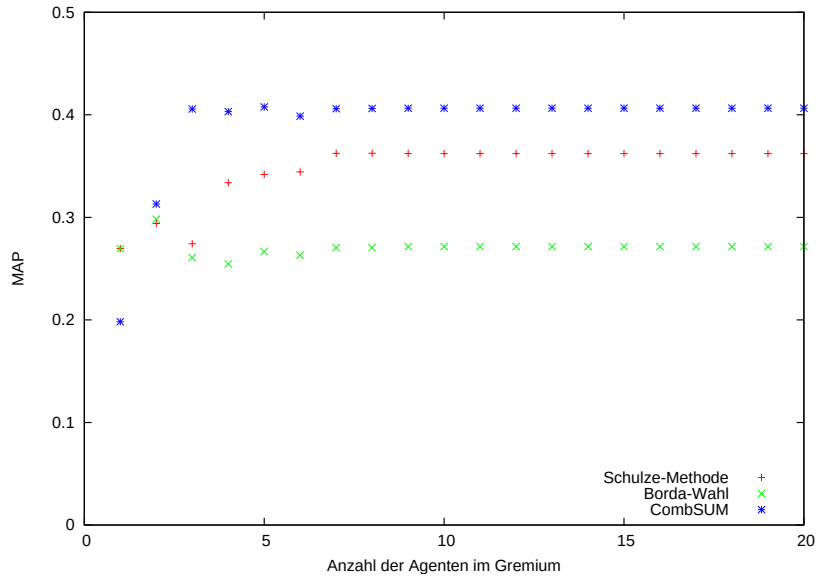


Abbildung 7.6: Die Graphik zeigt die mittlere Precision (MAP) in Abhängigkeit von der Anzahl der berücksichtigten Agenten für verschiedene Filterkombinationsalgorithmen.

7.1.5 Zusammenstellen eines Gremiums durch inkrementelles Reduzieren eines Agentengremiums

Um die Stärken und Schwächen des inkrementellen Zusammenstellens von Agentengremien mit anderen Ansätzen vergleichen zu können, werden im Folgenden verschiedene alternative Herangehensweisen zum Erstellen von Gremien skizziert und evaluiert.

Algorithmus 4 Inkrementelles Verkleinern eines Agenten-Gremiums.

Erstelle ein Gremium aus der Gesamtheit der Filteragenten: $G \leftarrow A$
while $G \neq \emptyset$ **do**
  Ermittle ein $s \in G$, so dass $\mathrm{P}(\mathrm{Comb}(G \backslash s)) \geq \mathrm{P}(\mathrm{Comb}(G \backslash s_i))\ \forall s_i \in G$
  $G \leftarrow G \backslash s$,
end while

Analog zum inkrementellen Zusammenstellen eines Gremiums von Filteragenten wird analysiert, wie sich die Filterpräzision beim schrittweisen Verkleinern eines Gremiums verändert. Man beginnt mit einem aus allen Agenten bestehenden Gremium und entfernt jeweils den Agenten, durch den das verbleibende Gremium die höchste Filterpräzision erreicht. Da bei Gremien mit vielen Agenten das Ermitteln von optimalen Gewichten mit hohem

Aufwand verbunden ist, wird allen Agenten dieselbe Gewichtung zugeordnet (Algorithmus 4).

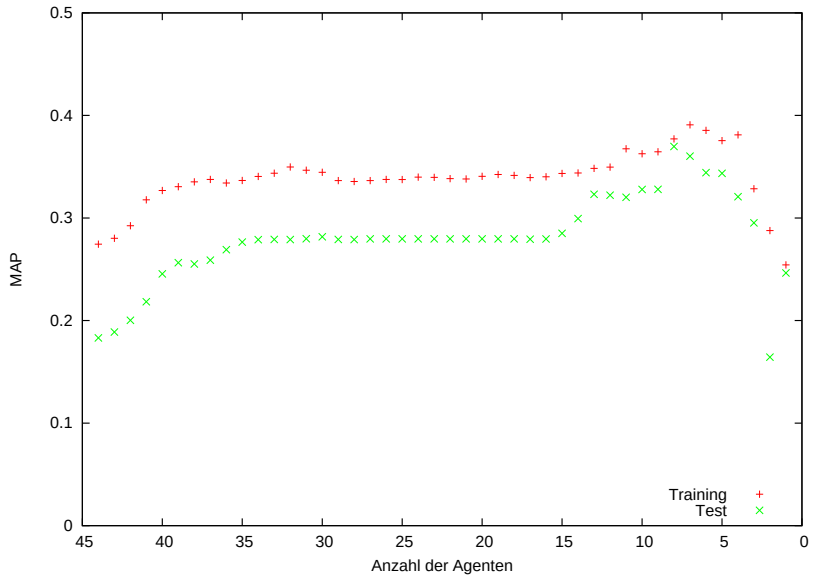


Abbildung 7.7: Die Graphik zeigt die mittlere Precision (MAP) in Abhängigkeit von der Größe eines Gremiums von Filteragenten. Die Kombination der einzelnen Filterstrategien erfolgt unter Verwendung von CombSUM (ohne die Berücksichtigung von Gewichtungen).

Ergebnisse Die beim schrittweisen Reduzieren der Größe des Agentengremiums beobachtete Abhängigkeit der Filterqualität von der Anzahl der Agenten bestätigt die beim inkrementellen Erweitern des Gremiums gewonnenen Aussagen (Abschnitt 7.1.2): Eine hohe Filterpräzision lässt sich mit einer relativ kleinen Anzahl von Filteragenten erreichen. Bei einer sehr starken Verkleinerung (z. B. auf zwei Agenten) fällt die durch das Gremium erreichte Filtergüte stark ab. Aus sehr vielen Agenten zusammengesetzte Gremien weisen eine geringe Filterpräzision auf. Durch das Verkleinern des Gremiums verbessert sich die Gesamtfilterqualität. Dies deutet darauf hin, dass in der Menge der verfügbaren Filteragenten mehrere weniger leistungsfähige Agenten enthalten sind, die die Bewertungen der guten Agenten „überstimmen“. Agenten, deren Entfernen aus dem Expertengremiums keine Veränderung der Filterqualität bewirkt, benötigen Ressourcen, ohne das Ergebnis zu verbessern.

7.1.6 Erstellen von Gremien auf der Basis von Clustern

Beim inkrementellen Zusammenstellen eines Gremiums von Bewertungsagenten wächst der zum Auswählen des Agenten notwendige Aufwand linear mit der Anzahl der verfügbaren Agenten. Um bei einer großen Anzahl von Agenten den erforderlichen Aufwand zum Zusammenstellen eines Gremiums zu reduzieren, identifiziert man Agenten mit ähnlichen Eigenschaften. Man wählt dazu ein geeignetes Ähnlichkeitsmaß und fasst untereinander ähnliche Agenten in Clustern zusammen. Die beim Ermitteln von Agenten gewonnenen Informationen können genutzt werden, um Duplikate von Filterstrategien („geklonte Filteragenten“) zu entdecken, um Ähnlichkeiten von Filtern mit bereits bekannten fehlerhaften Agenten zu erkennen und um die Heterogenität innerhalb von Agentengremien zu steuern [SNL+07, WLP04, MM03].

Ermitteln von Clustern

Um Gruppen von untereinander ähnlichen Bewertungsagenten zu bestimmen, wählt man ein Maß zum Ermitteln der Ähnlichkeit von Filteragenten sowie einen Cluster-Algorithmus. Es wird zunächst der Fall betrachtet, dass keine Benchmarkdaten zur Verfügung stehen und die Filterpräzision der Filteragenten nicht bekannt ist. Man definiert die Ähnlichkeit zwischen Filteragenten auf Basis der KENDALLS TAU-Rangkorrelation auf den pro Anfrage ermittelten Dokumentenmengen. Zum Gruppieren der Filteragenten wird ein Algorithmus des hierarchischen agglomerativen Clusterings verwendet, wobei der Abstand zwischen den Clustern als mittlere Distanz zwischen den Elementen der Cluster berechnet wird („Durchschnittliche Distanz“).

Die beim Gruppieren der Bewertungsstrategien des LETOR-TD2004-Datensatzes erhaltenen Cluster visualisiert die Abbildung 7.8. Die ermittelte Gruppierung zeigt, dass die ermittelten Cluster sehr unterschiedliche Größen besitzen. So bilden die Filterstrategien ‘8’ und ‘9’ sowie ‘10’ und ‘11’) zweielementige Cluster und heben sich dadurch von den anderen Strategien deutlich ab. Dies deutet darauf hin, dass sich die von diesen Filterstrategien eingesetzten Algorithmen von den übrigen Filterstrategien unterscheiden.

Zum Vergleich berechnet man die Gruppierung, die man erhält, wenn man als Ähnlichkeitsmaß die PEARSON-Korrelation der pro Anfrage erreichten Filterpräzision verwendet (Abbildung 7.9). Vergleicht man die unter Verwendung der PEARSON-Korrelation auf der mittleren Precision erhaltene Gruppierung mit der unter Verwendung des KENDALLS TAU-Rangkorrelation erhaltenen Gruppierung, so stellt man fest, dass sich die erhaltenen Cluster deutlich unterscheiden. Allerdings stimmen die ermittelten zweielementigen Cluster größtenteils überein. D. h. dass zwei Filterstrategien, die bezüglich

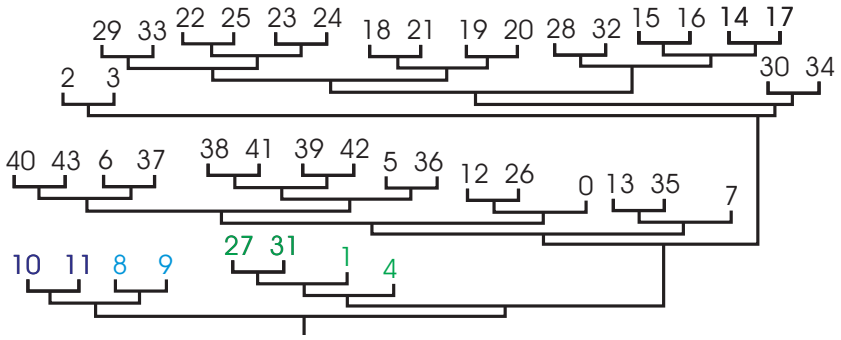


Abbildung 7.8: Die Abbildung zeigt eine Gruppierung der Filterstrategien des LETOR-TD2004-Datensatzes. Die Berechnung der Ähnlichkeit zwischen Filterstrategien erfolgte auf Basis der KENDALLS TAU-Rangkorrelation auf den sortierten Dokumentenlisten. Die Gruppierung der Agenten erfolge unter Verwendung eines hierarchischen agglomerativen Clusteralgorithmus, wobei die Distanz zwischen den Clustern als mittlere Distanz zwischen den Elementen der Cluster berechnet wurde.

der Rangkorrelation KENDALLS TAU stark miteinander korrelieren, auch bezüglich der PEARSON-Korrelation ähnlich sind.

Definieren von Gremien auf der Basis von Clustern

Die durch das Gruppieren von Bewertungsstrategien gewonnenen Informationen können genutzt werden, um beim Zusammenstellen von Agentengremien die Diversität der berücksichtigten Strategien zu steuern.

Es wird zunächst der Fall betrachtet, dass keine Benchmarkdaten zur Verfügung stehen. Das Clustern der Bewertungsstrategien erfolgt auf Basis der KENDALLS TAU-Rangkorrelation der pro Anfrage ermittelten Bewertungen. Da beim Fehlen von Benchmarkdaten die Güte der einzelnen Bewertungsstrategien nicht bekannt ist, kann die Auswahl der in einem Gremium berücksichtigten Strategien nicht auf Basis der Präzision erfolgen. Als repräsentative Filterstrategien für einen Cluster wählt man jeweils die Strategie aus, die die höchste mittlere Korrelation mit den anderen Strategien des Clusters aufweist. Zum Erstellen eines Agentengremiums wählt man die gewünschte Anzahl von Clustern und verwendet von jedem Cluster eine repräsentative Filterstrategie.

Die Abbildung 7.10 zeigt die mittlere Precision (MAP) für die auf Basis verschiedener Cluster-Strategien erstellten Agentengremien. Es wird die Präzision von zufällig zusammengestellten Gremien (unterschiedlicher Größe) der Präzision von Gremien gegenübergestellt, die anhand von Clustern

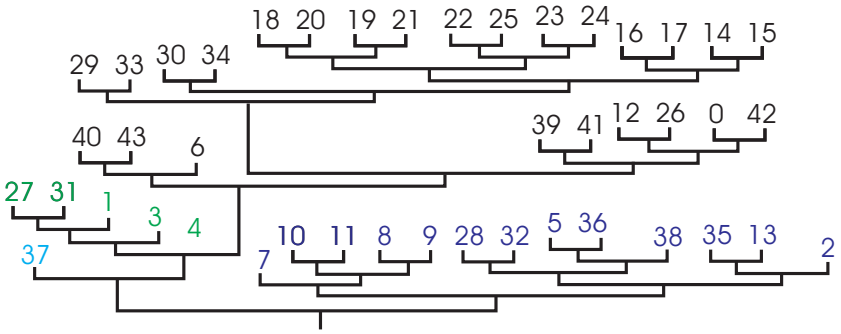


Abbildung 7.9: Die Abbildung zeigt eine Gruppierung der Filterstrategien des LETOR-TD2004-Datensatzes. Die Berechnung der Ähnlichkeit zwischen Filterstrategien erfolgte auf Basis der Pearson-Korrelation der pro Anfrage erreichten Average Precision (AP).

erstellt wurden. Die ermittelten Ergebnisse zeigen, dass die Präzision bei zufällig gebildeten Clustern im Durchschnitt mit der Anzahl der erstellten Clustern wächst. Die mittlere Precision variiert stark mit den erstellten Clustern.

Die Filterpräzision der Agentengremien, die auf der Basis einer Korrelationsanalyse erstellten Cluster gebildet wurden, ist deutlich geringer als die Filterpräzision zufällig zusammengestellter Agentengremien. Durch das Berücksichtigen möglichst heterogener Filterstrategien erhalten „außergewöhnliche“ Filter (mit einer geringen Filterpräzision) einen großen Einfluss. Dadurch weisen die abgeleiteten Agentengremien eine niedrige mittlere Precision auf. Dies gilt sowohl für die auf Basis der KENDALLS TAU-Randkorrelation als auch für die auf Basis der PEARSON-Korrelation zusammengestellten Agentengremien.

Erstellen von Gremien unter Verwendung von Benchmarkdatensätzen Stehen zu den Bewertungsstrategien Benchmarkdaten zur Verfügung, so kann die Filterpräzision zum Erstellen von Gremien genutzt werden. Für das Zusammenstellen von Gremien auf der Basis von geclusterten Bewertungsstrategien wurden die folgenden Verfahren analysiert:

- **Ausschließen von Clustern:** Ist die Leistungsfähigkeit der Filteragenten eines Clusters bekannt, so können beim Zusammenstellen eines Gremiums von Filteragenten die Cluster unberücksichtigt bleiben, deren Agenten eine schlechte Filterqualität aufweisen. Dies ermöglicht es, fehlerhafte Filteragenten auszuschließen.

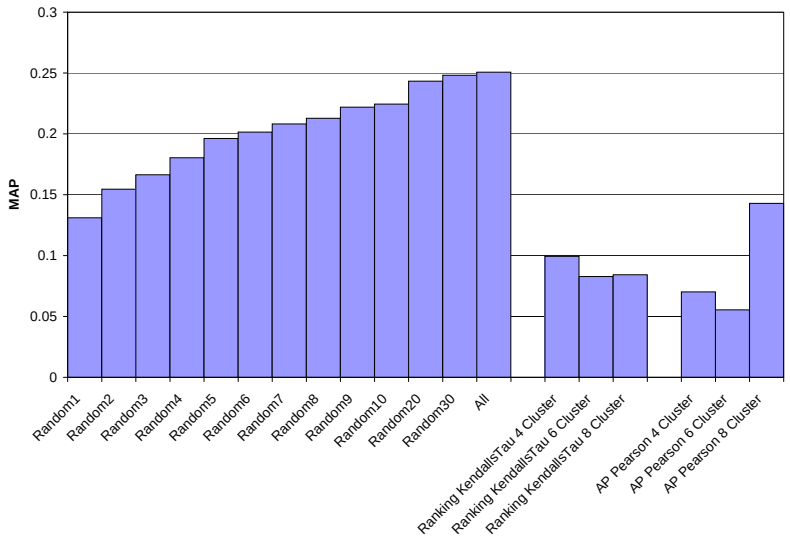


Abbildung 7.10: Das Diagramm zeigt die Filterpräzision für zufällig zusammengestellte Agentengremien unterschiedlicher Größe. Zusätzlich ist die Filterpräzision für die auf Basis von Clustern abgeleiteten Agentengremien dargestellt. Die Messwerte zeigen, dass die auf Basis von Clustern erstellten Agentengremien eine deutlich geringere Filterpräzision aufweisen als zufällig zusammengestellte Agentengremien. Dies zeigt, dass in den auf Basis von Clustern erstellten Gremien Agenten mit geringer Filterpräzision einen zu starken Einfluss haben.

- **Auswählen des jeweils besten Filteragenten pro Cluster:** Um ein leistungsfähiges Gremium von Agenten zusammenzustellen, sollten beim Bilden von Agentengremien die Cluster bevorzugt werden, die Agenten mit einer hohen Filterpräzision enthalten.

- **Mehrstufige Auswahl von Agenten:** Beim inkrementellen Zusammenstellen von Filterergebnissen wird für jeden Agenten analysiert, welcher Agent in ein Gremium aufgenommen werden sollte, um die Güte möglichst stark zu verbessern (Aufwandsklasse $\mathcal{O}(n)$, wobei n die Anzahl der Agenten). Um den Aufwand zu reduzieren, wählt man zunächst den am besten geeigneten Cluster und von diesem Cluster den am besten geeigneten Filteragenten aus (Aufwandsklasse $\mathcal{O}(\sqrt{n})$).

Die Graphik 7.11 visualisiert die Leistungsfähigkeit verschiedener Agentengremien, die auf der Basis von Clustern zusammengestellt wurden. Die Ergebnisse zeigen, dass die höchste Filterpräzision erreicht wird, wenn die

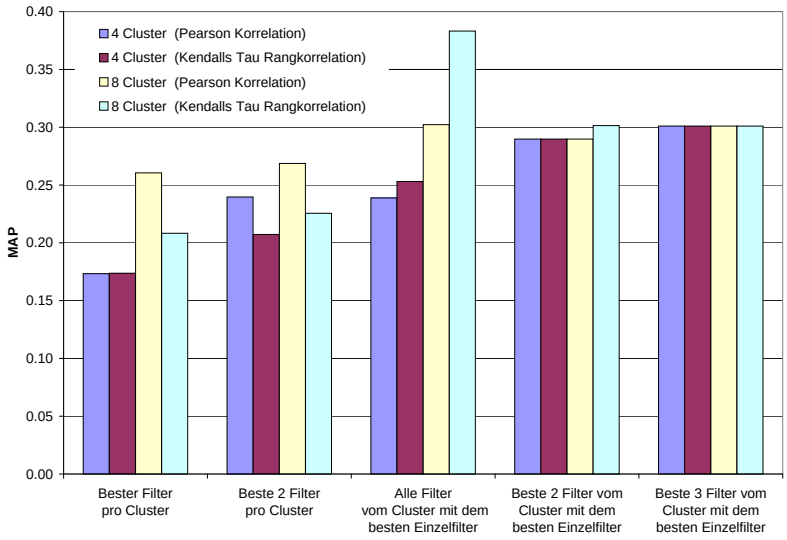


Abbildung 7.11: Das Diagramm zeigt die Filterpräzision der auf Basis von Clustern zusammengestellten Agentengremien. Eine hohe Filterpräzision wird erreicht, wenn man aus dem Cluster mit dem besten Einzelagenten die besten Agenten zu einem Gremium kombiniert.

Filterstrategien auf der Basis der Filterpräzision gruppiert werden und das Agentengremium aus den Agenten des besten Clusters zusammengestellt wird.

Das Berücksichtigen möglichst vieler Cluster ermöglicht im Allgemeinen keine guten Ergebnisse. Filterstrategien mit einer geringen Filterpräzision erhalten dabei einen zu starken Einfluss auf das zusammengestellte Agentengremium.

Gute Filterergebnisse erhält man, indem Gremien aus den besten Agenten des Clusters mit dem besten einzelnen Agenten erstellt werden. Dies deutet darauf hin, dass (sowohl bei der Verwendung der KENDALLS TAU-Rangkorrelation als auch bei der Verwendung der PEARSON-Korrelation) Agenten mit hoher Filterpräzision in einem Cluster zusammengefasst werden. Gremien, die aus Agenten mit hoher Filterpräzision zusammengestellt werden, weisen im Allgemeinen auch eine hohe Filterpräzision auf.

Diskussion Das Berücksichtigen von einer großen Anzahl von Clustern (um möglichst unterschiedliche Strategien im Agentengremium zu berücksichtigen) hat auf die Filterpräzision der erstellten Gremien keinen positiven Einfluss. Durch das Berücksichtigen von untereinander verschiedenen Strate-

gien steigt die Wahrscheinlichkeit, dass in ein Agentengremium auch weniger leistungsfähige Bewertungsagenten aufgenommen werden. Die Ergebnisse zeigen, dass ein Maximieren der Diversität nicht dazu geeignet ist, um besonders leistungsfähige Agentenensembles zu erstellen.

Clustern ist eine effiziente Strategie zum Verkleinern des Suchraumes. So können Gruppen von weniger leistungsfähigen Filterstrategien identifiziert und beim Erstellen von Gremien ausgeschlossen werden. Gute Agentengremien erhält man, indem man leistungsfähige Agenten aus dem Clusters mit den besten Filteragenten erstellt. Allerdings berücksichtigt dieses Vorgehen nicht, wie diese Filteragenten zusammenpassen. Die Leistungsfähigkeit der auf der Basis von Clustern erstellten Gremien liegt unter der Filterpräzision, die beim inkrementellen Zusammenstellen von Agentengremien erreicht wird.

7.1.7 Ermitteln von Gewichtungen auf Basis von Regression

Die bisher vorgestellten Verfahren verfolgen das Ziel, inkrementell die mittlere Precision (MAP) zu optimieren. Stehen Benchmarkdaten zur Verfügung kann so kann das Ermitteln geeigneter Gewichte für CombSUM auch als ein Regressionsproblem behandelt werden.

Man wählt den folgenden Ansatz: Es sei M_q eine Matrix mit Bewertungen zu einer Anfrage q, die beschreibt, wie eine Menge von Agenten $A = \{a_0, a_1, \ldots, a_n\}$ eine Menge von Dokumenten $D = \{d_0, d_1, \ldots, d_m\}$ bewertet. Dabei bezeichne m_{ij} die Bewertung des i-ten Agenten für das Dokument d_j. Darüber hinaus sei w ein n-elementiger Gewichtungsvektor und r_q ein ausschließlich aus den Zahlen 0 und 1 bestehenden Vektor, der definiert, ob ein Dokument für die Anfrage q relevant ist. Zum Ermitteln der Gewichtungen für eine Anfrage q, betrachtet man das Gleichungssystem $M_q w = r_q$. Fasst man die Gleichungssysteme für alle Anfragen $Q = q_0, q_1, \ldots, q_k$ zusammen, erhält man:

$$Mw = r \quad \Longleftrightarrow \quad \begin{bmatrix} M_{q0} \\ M_{q1} \\ \vdots \\ M_{qk} \end{bmatrix} w = \begin{bmatrix} r_{q0} \\ r_{q1} \\ \vdots \\ r_{qk} \end{bmatrix}$$

Da im Allgemeinen die Menge der Dokumente groß gegen die Menge der Bewertungsagenten ist, ist das Gleichungssystem überbestimmt. Man ermittelt unter Verwendung der Methode der kleinsten Quadrate den Gewichtungsvektor w, so dass $\left\|Mw - r\right\|^2$ minimal wird. Dazu berechnet man

$w = M^+ r$, wobei M^+ die Moore-Penrose-Inverse zu M ist. Beim Lösen des Ausgleichsproblems kann es zu Overfitting kommen. Dieses Overfitting kann durch das Durchführen einer Regularisierung reduziert werden. Ziel der Regularisierung ist es, den Einfluss kleiner Singulärwerte der Matrix M zu reduzieren. An Stelle des Gleichungssystems $Mw = r$ löst man $\left(M^T M + \alpha I\right) w = M^T r$, d. h. man minimiert $\left\|Mw - r\right\|^2 + \alpha \left\|w\right\|^2$. Die Regularisierung vergrößert zwar $\left\|Mw - r\right\|^2$, führt aber bei geeignetem Parameter α zu einem Gewichtsvektor, der besser zum gestellten Problem passt (Reduzierung des Rauschens) [Tik63, Bis06].

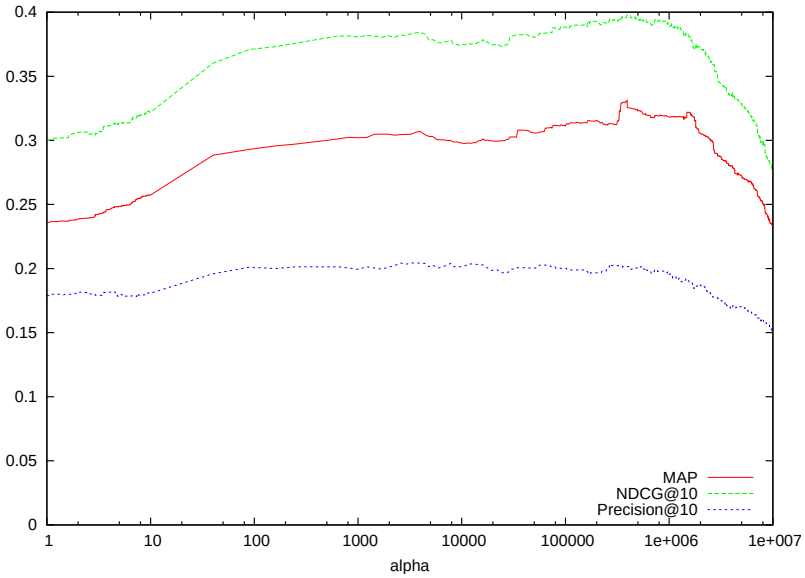


Abbildung 7.12: Die Graphik zeigt Exaktheit (MAP, Precision@10, NDCG@10) für ein Agentengremium bestehend aus allen 44 Agenten des LETOR-TD2004-Datensatzes in Abhängigkeit vom Regularisierungsparameter α. Die Gewichtungen der Agenten wurden durch Minimierung des Gleichungssystems $||Mw - r\ ||^2 + \alpha||r||^2$ ermittelt.

Die Graphik 7.12 zeigt die erreichte Filterpräzision in Abhängigkeit vom Regularisierungsparameter α. Die ermittelten Ergebnisse zeigen, dass durch eine geeignete Regularisierung die Filterpräzision deutlich verbessert wird. Die erreichte mittlere Precision (MAP) liegt dabei über der mittleren Precision (MAP) des besten Einzelagenten, aber deutlich unter der, die durch inkrementelles Zusammenstellen eines Filteragenten-Ensemble erreicht wird (vgl. Abschnitt 7.1.3). Im Gegensatz zu den inkrementellen Verfahren hat das Berechnen des Gewichtungsvektors als lineares Ausgleichsproblem den Vorteil, dass der Vektor w effizient berechnet werden kann. Der ermittel-

te Gewichtsvektor kann sowohl positive als auch negative Zahlen enthalten. Untersucht man die ermittelten Gewichtsvektoren w in Abhängigkeit vom Regularisierungsparameter α, so stellt man fest, dass bei größerem α der Anteil der positiven Elemente des Gewichtsvektors steigt und der Betrag der Elemente sinkt (die Norm des Vektors w wird minimiert).

Um die Güte des linearen Ausgleichsproblems weiter zu verbessern, projiziert man die Daten in einen neuen Raum. Man verwendet die folgende Umformungen: Aus $Mw = r$ folgt mit $\tilde{w} = M^T w$ die Gleichung $MM^T\tilde{w} = r$. Unter Verwendung der (pseudo-)inversen Matrix zu MM^T, so erhält man $w = M^T \left(MM^T\right)^+ r$. Um MM^T effizient berechnen zu können, führt man eine Dimensionsreduktion durch. Auf Basis eine Hauptkomponentenanalyse von M leitet eine Abbildung U_n ab, die die Spaltenvektoren von M in einen n dimensionalen Unterraum projiziert. Der Gewichtungsvektor w lässt sich nun berechnen als

$$w = \left(U_n\left(M\right)\right)^T \left(U_n\left(M\right) U_n\left(M\right)^T\right)^{-1} U_n\left(r\right)$$

Um die Bewertung $\hat{r}$ für einen neuen Eingabevektor $\hat{m}$ zu ermitteln, berechnet man:

$$\hat{r} = U_n\left(\hat{m}\right)\left(U_n\left(M\right)\right)^T \left(U_n\left(M\right) U_n\left(M\right)^T\right)^{-1} U_n\left(r\right)$$

Die erreichte Filter-Exaktheit in Abhängigkeit der Anzahl der Dimensionen n (des bei der Hauptkomponentenanalyse verwendeten Unterraumes) zeigt die Graphik 7.13. Anhand der ermittelten Ergebnisse ist zu erkennen, dass mit einer Dimensionsreduktion auf 10 - 20 Dimensionen die Güte der Ergebnisse kaum sinkt.

Als eine weitere Strategie, um die Berechnung der Gewichte zu verbessern, wird die Verwendung des „Kernel-Tricks“ untersucht. Man ersetzt dazu die Matrix MM^T durch eine von einer Kernelfunktion abgeleiteten Matrix $k(M, M)$. Es wurden unterschiedliche (polynomielle und RBF) Kernel getestet. Allerdings konnte durch keinen der getesteten Kernel eine signifikante Verbesserung der Filterqualität erreicht werden.

Für das Zusammenstellen eines optimalen Gremiums von Filteragenten sind die durch Lösung des linearen Ausgleichsproblems erhaltenen Gewichtsvektoren nicht geeignet. Allerdings können die erhaltenen Gewichtsvektoren Informationen über Abhängigkeiten zwischen den verfügbaren Filteragenten liefern und als Ausgangspunkt für weitergehende Optimierungen dienen.

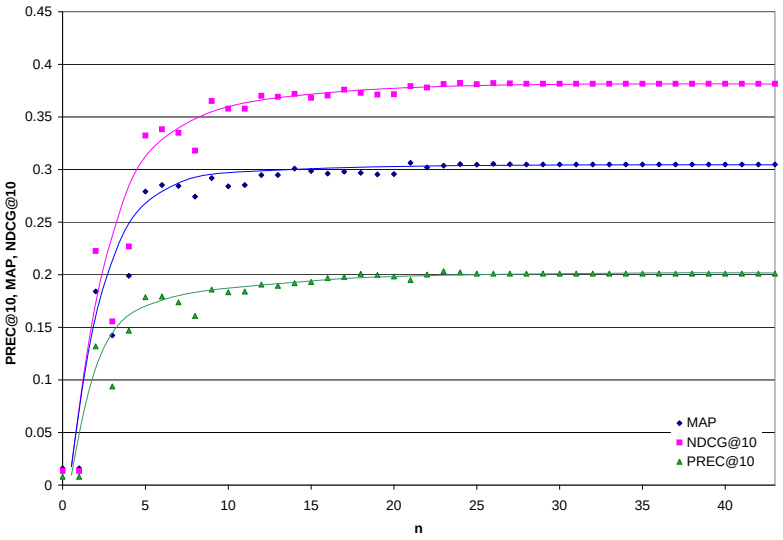


Abbildung 7.13: Die Graphik zeigt die erreichte Filterexaktheit in Abhängigkeit von der bei der Hauptkomponentenanalyse berücksichtigten Dimensionen. Die Gewichtungen werden dabei nach dem Prinzip des linearen Ausgleichsproblems berechnet, wobei eine geeignete Regularisierung zur Anwendung kommt.

7.1.8 Delegieren von Anfragen

Eine Alternative zum Kombinieren der Bewertungen mehrerer Bewertungsagenten besteht darin, Suchanfragen jeweils an den für eine Anfrage am besten geeigneten Agenten zu delegieren. Man zerlegt dazu das Bewerten von Dokumenten in zwei Teilaufgaben: Das Ermitteln, welcher Agent am besten für eine Anfrage geeignet ist, und das Delegieren der Anfrage an den jeweils besten Agenten.

Zunächst wird das Delegieren von Anfragen analysiert. Um zu ermitteln, welche mittlere Precision (MAP) maximal beim Delegieren von Anfragen an einen einzelnen Agenten erreicht werden kann und wie viele Agenten beim Delegieren von Anfragen berücksichtigt werden müssen, wird das folgende Szenario genauer betrachtet:

- Es wird angenommen, dass es anhand der Anfrage möglich ist, den für eine Anfrage am besten geeigneten Agenten zu ermitteln. Auch wenn diese Annahme für praktische Anwendungen wenig realistisch ist, ermöglicht die Annahme (unter Verwendung von Trainingsdatensätzen),

die bei einer optimalen Delegationsstrategie maximal erreichbare Präzision zu ermitteln.

- Um die erreichte mittlere Precision (MAP) in Abhängigkeit von der Anzahl der beim Delegieren berücksichtigten Agenten zu ermitteln, wird inkrementell eine Menge von Agenten F bestimmt, aus der jeweils der am besten geeignete Filteragent $s \in F \subseteq A$ zum Bewerten der Dokumente bezüglich der Anfrage ausgewählt wird. Die dabei erreichte mittlere Precision ergibt sich dabei wie folgt: $\mathrm{MAP}(F,Q) = \frac{1}{|Q|}\sum_{q\in Q}\max_{f\in F} AP(f,q)$.

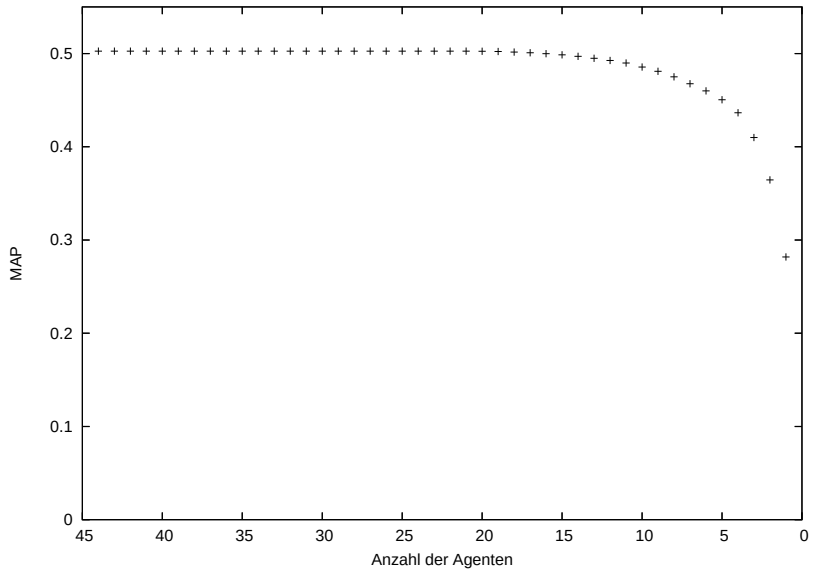


Abbildung 7.14: Die mit dem LETOR-TD2004-Datensatz erreichte mittlere Precision (MAP) in Abhängigkeit der Anzahl der Agenten, die beim Delegieren von der Anfrage berücksichtigt werden. Die Anfrage wird jeweils an den am besten geeigneten Agenten weitergeleitet.

Für das Zusammenstellen der Menge F der berücksichtigten Agenten wird sowohl das inkrementelle Vergrößern als auch das inkrementelle Verkleinern der Menge analysiert:

- Man beginnt mit einer Menge $F = A$ aus allen verfügbaren Agenten A und entfernt aus dieser Menge F jeweils den Agenten s, so dass die mittlere Precision $\mathrm{MAP}(F,Q)$ möglichst wenig sinkt, d. h. $\mathrm{MAP}(F\backslash s,Q) \geq \mathrm{MAP}(F\backslash f,Q) \;\; \forall f \in F$.

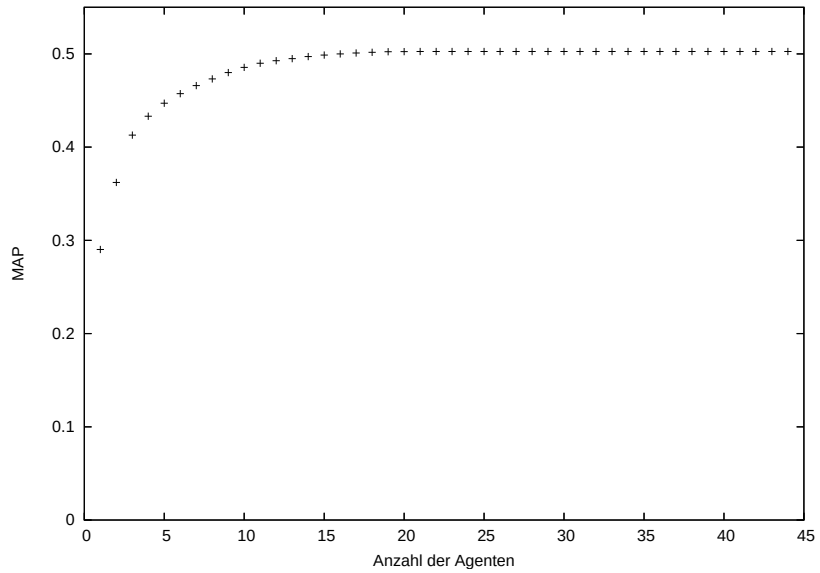


Abbildung 7.15: Die mit dem LETOR-TD2004-Datensatz erreichte mittlere Precision (MAP) in Abhängigkeit der Anzahl der Agenten, die beim Delegieren von der Anfrage berücksichtigt werden. Die Anfrage wird jeweils an den am besten geeigneten Agenten weitergeleitet. Die Menge der berücksichtigten Agenten wird schrittweise erweitert.

- Man beginnt mit einer leeren Menge $F = \emptyset$ und erweitert die Menge F jeweils um den Agenten s durch den $\text{MAP}(F, Q)$ am stärksten steigt, d. h. $\text{MAP}(F \cup s, Q) \geq \text{MAP}(F \cup f, Q) \quad \forall f \in F$.

Die ermittelten Ergebnisse sind in den Graphiken 7.14 und 7.15 dargestellt. Die Analyse zeigt, dass von den 44 Filteragenten des LETOR-TD2004-Datensatzes nur die Hälfte der Agenten beim Delegieren berücksichtigt werden müssen, um die maximale mittlere Precision (MAP) zu erreichen. D. h. etwa die Hälfe der Agenten sind bei allen Anfragen schlechter als ein anderer Agent. Um 95% der maximal beim Delegieren erreichbaren Filterqualität zu erreichen, werden lediglich acht Agenten benötigt; um 80% von $\text{MAP}(A, Q)$ zu erreichen werden nur drei Agenten benötigt[1].

Vergleicht man die in den Abbildungen 7.14 und 7.15 dargestellten Graphen, so stellt man fest, dass diese große Übereinstimmungen aufweisen. Die erreichte Filterpräzision ist bei gleicher Anzahl der berücksichtigten Agenten annähernd gleich.

[1] Werden alle 44 Agenten berücksichtigt, wird eine $\text{MAP}(A, Q) = 0.50$ erreicht; unter Berücksichtigung von 8 Agenten eine $\text{MAP}(F8, Q) = 0.47$ und mit drei Agenten eine $\text{MAP}(F3, Q) = 0.4$.

Die beim Delegieren von Anfragen ermittelte Zuordnung von Anfragen zu Agenten kann zum Gruppieren der Anfragen genutzt werden. Man fasst jeweils die Anfragen in einem Cluster zusammen, die an denselben Agenten delegiert werden. Die Abbildungen 7.16 und 7.17 zeigen die Zuordnung der Anfragen des LETOR-TD2004-Datensatzes zu Agenten für den Fall, dass 4 bzw. 8 Agenten beim Delegieren berücksichtigt werden. Die Menge der zum Delegieren von Anfragen herangezogenen Agenten wurde durch das inkrementelle Erweitern der Agentenmenge ermittelt.

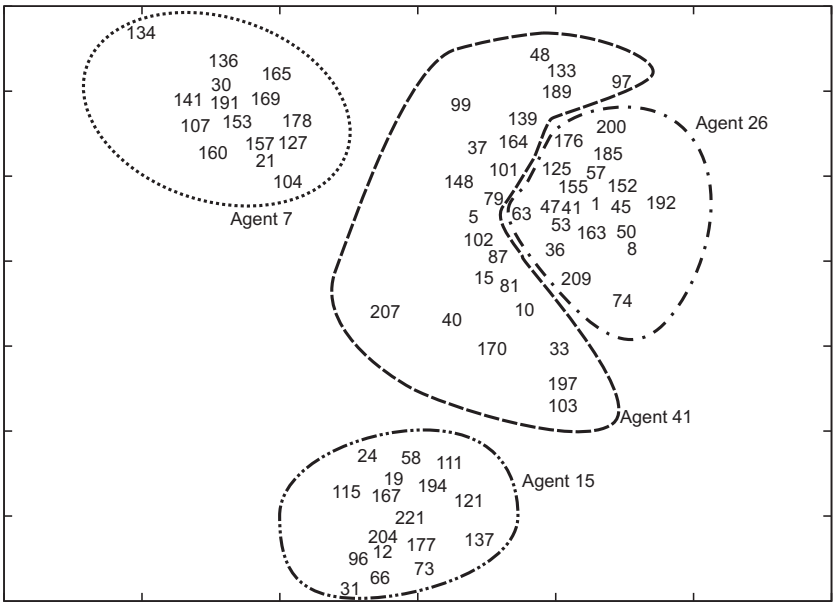


Abbildung 7.16: Die Abbildung zeigt die Zuordnung von Anfragen zu Agenten. Diese wurde ermittelt, indem für jede Anfrage der Agent bestimmt wurde, der für die Anfrage die höchste mittlere Precision (MAP) erreicht. Die Menge der berücksichtigen Agenten wurde mit $G = \{7, 15, 26, 41\}$ so gewählt, dass die mittlere Precision des System beim Delegieren optimal für alle $|G| = 4$ ist.

Die durch das Gruppieren von Anfragen gewonnenen Informationen können zum Zusammenstellen von Agentengremien genutzt werden. Die den Clustern zugeordneten Agenten können jeweils für eine Gruppe von Anfragen präzise Bewertungen ermitteln. Die Agenten sind untereinander verschieden, da sie jeweils auf eine Gruppe von Anfragen „spezialisiert“ sind.

Die Abbildung 7.18 zeigt die ermittelte mittlere Precision (MAP) für die auf Basis der Cluster erstellten Gremien. Es wird die mittlere Precision (MAP) der Einzelagenten der Präzision der Gremien gegenübergestellt,

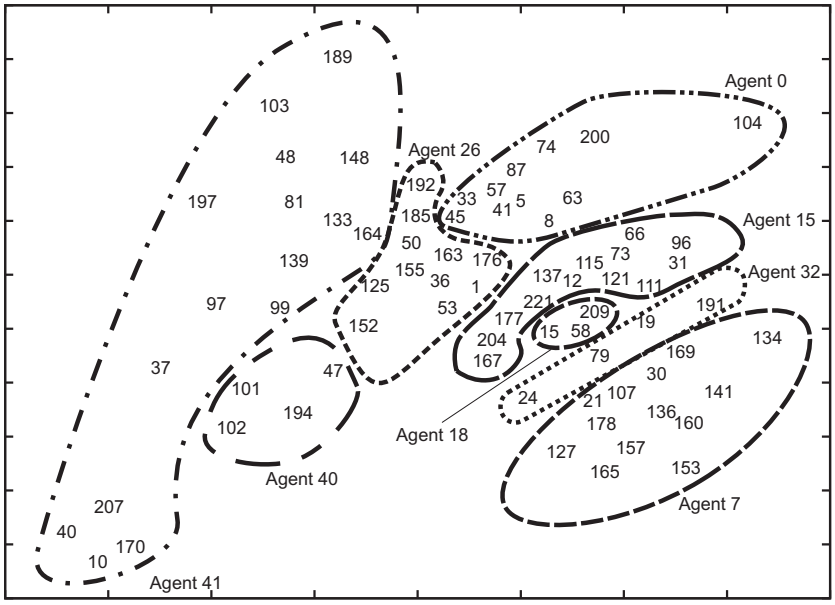


Abbildung 7.17: Die Abbildung zeigt die Zuordnung von Anfragen zu Agenten, falls die Anfragen jeweils an den am besten für eine Anfrage geeigneten Agenten delegiert werden. Die Menge der berücksichtigen Agenten wurde mit $G = \{0, 7, 15, 18, 26, 32, 40, 41\}$ so gewählt, dass die mittlere Precision des System beim Delegieren optimal für alle $|G| = 8$ ist.

welche die Kombinationsstrategien CombSUM, CombMAX und CombMIN verwenden.

Die Ergebnisse zeigen, dass die Strategie CombSUM die beste Filterpräzision ermöglicht. Eine besonders hohe mittlere Precision wird bei einem aus nur vier Agenten bestehenden Gremium erreicht. Das aus acht Agenten bestehende Gremium weist eine etwas geringere Filterpräzision auf. Durch die höhere Anzahl von berücksichtigen Agenten sind im Gremium auch Agenten vertreten, die auf nur wenige Anfragen spezialisiert sind, im Mittel über alle Anfragen aber nur eine niedrige Präzision erreichen (wie z. B. Agent 40). Die Filterpräzision des aus acht Agenten bestehenden Gremiums liegt aber über der Filterpräzision des besten Einzelagenten.

Diskussion Durch das Delegieren von Anfragen an den jeweils am besten geeigneten Agenten können hochqualitative Filterergebnisse erreicht werden. Allerdings kann im Allgemeinen weder anhand einer Anfrage noch an den von einem Agenten ermittelten Bewertungen bestimmt werden, welcher Filteragent am besten für eine Anfrage geeignet ist. Um eine Delegationsstrategie zum Filtern anwenden zu können, wird zusätzliches Wissen benötigt. Werden

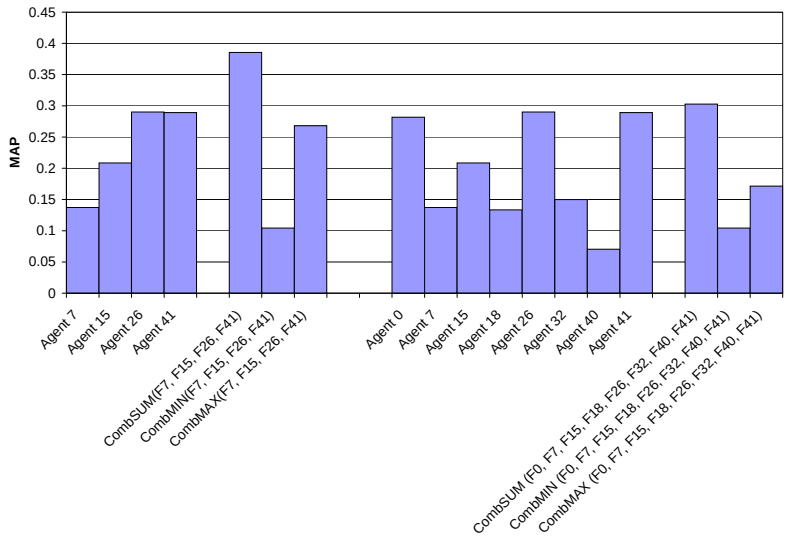


Abbildung 7.18: Die Abbildung zeigt die erreichte mittlere Precision für die Kombination von Filterstrategien unter Verwendung von Comb∗-Strategien (ohne die Verwendung von Gewichtungen).

Anfragen durch eine fehlerhafte Delegationsstrategie an ungeeignete Agenten delegiert, führt dies zu einer zu einer stark verringerten Filterpräzision.

Um das Problem zu umgehen, jeweils den für eine Anfrage am besten geeigneten Agenten zu bestimmen, fasst man Anfragen, die von einem Agenten besonders gut bewertet werden können in Gruppen zusammen. Auf Basis der den Clustern zugeordneten Agenten (d. h der Agenten, die die Anfragen besonders gut beurteilen) erstellt man unter Verwendung der Strategie CombSUM ein Agentengremium. Die in dem Gremium berücksichtigten Strategien zeichnen sich durch eine hohe Filterpräzision für eine Gruppe von Anfragen und durch die „Spezialisierung“ auf unterschiedliche Typen von Anfragen aus. Das beschriebene Vorgehen ermöglicht es, (mit geringem Aufwand) Gremien zu erstellen, die sich durch eine hohe Filterpräzision auszeichnen.

7.2 Verlässliche Agentengremien

Ein wichtiges Kriterium zum Bewerten der Güte eines Filteragenten ist neben der Filterpräzision die Wahrscheinlichkeit für eine neue Anfrage gute Ergebnisse zu erhalten. Man analysiert dabei, wie die Güte der Filterergebnisse über die Menge der Anfragen variiert. Im Folgenden werden zunächst verschiedene Gütefunktionale erläutert, mit denen die Verlässlichkeit von Fil-

teragenten bzw. Agentengremien evaluiert werden kann. In Anschluss daran wird gezeigt, wie bezüglich der Verlässlichkeit optimierte Agentengremien erstellt werden können. Es wird analysiert, inwieweit sich die Verlässlichkeit dieser Agentengremien von der Verlässlichkeit von Gremien unterscheidet, die auf eine maximale Präzision optimiert sind.

7.2.1 Berechnen der Verlässlichkeit

Um beim Bestimmen der Verlässlichkeit der Relevanzbewertungen die Menge der verfügbaren Bewertungsagenten und den Schwierigkeitsgrad der Anfragen zu berücksichtigen, wird die jeweils für eine Anfrage erreichte Filterpräzision im Vergleich zu einer Referenzfilterstrategie betrachtet. Als Referenzstrategien werden die folgenden Filterstrategien betrachtet:

- **Referenzstrategie R_1:** Für jede Anfrage wird der am besten für diese Anfrage geeignete Filteragent ausgewählt. Die für eine Anfrage q erreichte durchschnittliche Precision (AP) ergibt sich wie folgt: $\mathrm{AP}(R_1, q) = \max_{f \in A} \mathrm{AP}(f, q)$. Dabei sei A die Gesamtheit der verfügbaren Filteragenten.

- **Referenzstrategie R_2:** Es wird eine Filterstrategie verwendet, deren durchschnittliche Precision für eine Anfrage q mit dem über alle Filteragenten A gemittelten durchschnittlichen Precision (AP) entspricht: $\mathrm{AP}(R_2, q) = \frac{1}{|A|} \sum_{f \in A} \mathrm{AP}(f, q)$.

Auf Basis dieser Referenzstrategien werden die folgenden Maße berechnet:

- **Die durchschnittliche Abweichung von der optimalen Delegationsstrategie (V_1):** Es wird die über alle Anfragen gemittelte Differenz zwischen der durchschnittliche Precision (AP) eines Filteragenten f und der AP der Referenzstrategie R_1 berechnet.

$$V_1(f) = \frac{1}{|Q|} \sum_{q \in Q} (\mathrm{AP}(f, q) - \mathrm{AP}(R_1, q))$$

- **Die relative Häufigkeit, dass ein Filteragent für eine Anfrage schlechter als der durchschnittliche Agent ist (V_2):** Es wird das Verhältnis zwischen der Anzahl der Anfragen, für die ein Filteragent f eine höhere durchschnittliche Precision (AP) als der Referenzfilter R_2 erreicht, und der Anzahl der Anfragen berechnet.

7.2.2 Analyse der bezüglich MAP optimierten Gremien

Man analysiert, welche Verlässlichkeit die Gremien aufweisen, die bezüglich der mittleren Precision (MAP) optimiert wurden. Die Graphik 7.19 zeigt die ermittelte Präzision und die Verlässlichkeit der im Abschnitt 7.1.3 zusammengestellten Agentengremien.

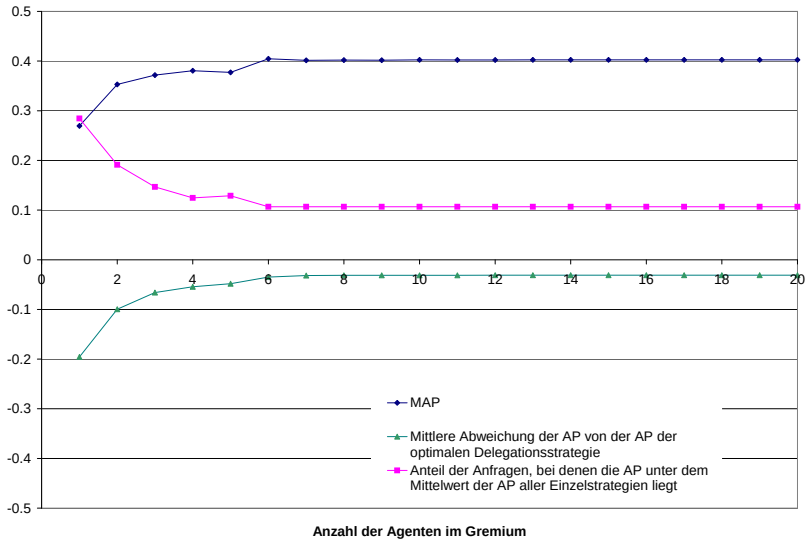


Abbildung 7.19: Die Graphik zeigt die Verlässlichkeit der im Abschnitt 7.1.3 zusammengestellten Gremien von Filteragenten. Die Bewertungen der Agenten werden unter Verwendung von CombSUM kombiniert.

Die ermittelten Graphen zeigen, dass die Verlässlichkeit stark mit der Filterpräzision korreliert. D. h. je höher die Filterpräzision eines Agentengremiums, desto geringer ist die Wahrscheinlichkeit, für eine Anfrage unterdurchschnittliche Ergebnisse zu erhalten. Eine hohe Filterpräzision impliziert auf der analysierten Anfragemenge, dass die mittlere Abweichung von der optimalen Delegationsstrategie gering ist.

Um die Abhängigkeit zwischen Präzision und Verlässlichkeit genauer zu analysieren, untersucht man, welche Filterpräzision Gremien aufweisen, die auf Verlässlichkeit optimiert werden.

7.2.3 Optimieren der Verlässlichkeit

Beim Zusammenstellen von Agentengremien kann anstatt der mittleren Precision (MAP) ein Gütefunktional zur Bewertung der Verlässlichkeit verwen-

det werden. Dazu passt man den Algorithmus 3 (Abschnitt 7.1.4) an, indem man als Gütefunktional P die Verlässlichkeit (z. B. V_1, V_2) verwendet.

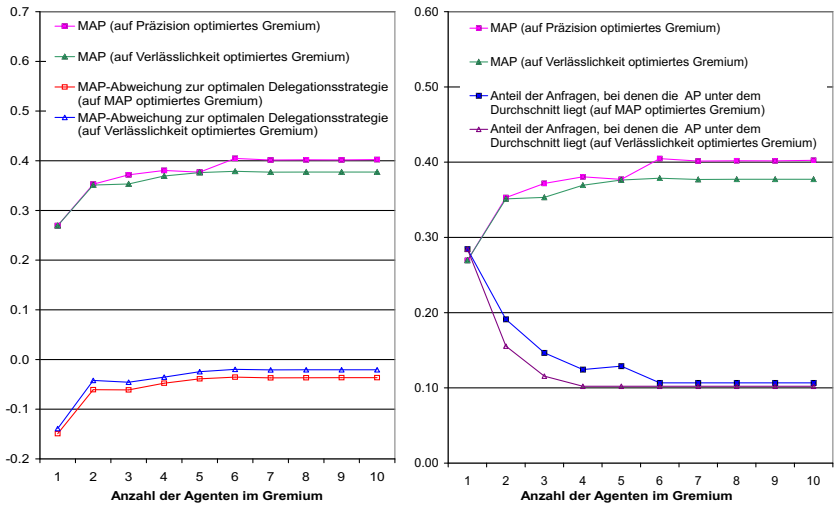


Abbildung 7.20: Die Graphik zeigt die mittlere Precision (MAP) und die Verlässlichkeit von auf die Verlässlichkeit optimierten Agentengremien in Abhängigkeit von der Größe des Agenten-Ensembles.

Graphik 7.20 stellt die erreichte Filterpräzision und die Verlässlichkeit für verschiedene Optimierungsfunktionale gegenüber. Die Ergebnisse zeigen, dass auf Verlässlichkeit optimierte Gremien im Vergleich zu auf MAP-optimierten Gremien eine etwas höhere Verlässlichkeit, bei etwas reduzierter Filterpräzision aufweisen.

Die ermittelten Messwerte bestätigen die starke Korrelation zwischen der Verlässlichkeit und der Filterpräzision. Lediglich bei Verwendung des Maßes V_2 ist bei kleinen Agentengremien festzustellen, dass im Vergleich mit den bezüglich MAP optimierten Gremien eine höhere Verlässlichkeit erreicht werden kann.

Aufgrund der vergleichbaren Verlässlichkeit und Filterpräzision der mit Hilfe von verschiedenen Gütefunktionalen erstellten Agentengremien erscheint bei dem analysierten Datensatz (LETOR-TD2004) eine spezielle Optimierung bezüglich der Verlässlichkeit nicht sinnvoll. Bei Verwendung der mittleren Precision (MAP) zum Erstellen von Agentengremien weisen diese bereits eine gute Verlässlichkeit auf.

7.3 Adaptive Agentengremien

Die in den vorangehenden Abschnitten verwendeten Methoden zum Erstellen von Agentengremien sind auf Szenarien optimiert, bei denen die Menge der verfügbaren Bewertungsagenten unveränderlich ist und eine Menge von bewerteten Anfragen zur Verfügung steht. In der Praxis muss aber der Fall berücksichtigt werden, dass ein neuer Anwender das Informationssystem benutzen möchte. Für neue Benutzer stehen keine bewerteten Anfragen zur Verfügung, so dass das Informationssystem aus den inkrementell vom Benutzer abgegebenen Bewertungen lernen muss. Darüber hinaus muss ein adaptives Informationssystem Verfahren anbieten, um auf das Hinzufügen und den Ausfall von Agenten zu reagieren.

7.3.1 Inkrementelle Benutzerbewertungen

Für Benutzer, die das Informationssystem erstmalig benutzen, stehen zunächst keine bewerteten Trainingsdokumente zur Verfügung, die für das Zusammenstellen eines Agentengremiums genutzt werden können. Das Informationssystem sollte in der Lage sein, anhand der inkrementell vom Benutzer abgegebenen Bewertungen ein auf die individuellen Präferenzen des Benutzers abgestimmtes Gremium zu erstellen. Im Folgenden werden verschiedene Methoden betrachtet, mit denen Agentengremien an neue Bewertungen angepasst werden können.

Ein Gremium mit fester Größe

Es wird zunächst an einem Beispiel untersucht, welche Filterpräzision erreicht wird, wenn bei inkrementellen Bewertungen ein statisches Agentengremium eingesetzt wird. Die Abbildung 7.21 zeigt die erreichte Filterpräzision für den Fall, dass ein unveränderliches Agentengremium verwendet wird. Die ermittelten Ergebnisse zeigen, dass die für eine Anfrage erreichte durchschnittliche Precision stark variiert. Die über die bereits bewerteten Anfragen gemittelte Präzision variiert zunächst stark, stabilisiert sich aber nach ca. 15-20 Anfragen. Da das verwendete Agentengremium unverändert bleibt (d. h. keine Adaption an die Bewertungen vorgenommen wird), ist die über alle Anfragen gemittelte durchschnittliche Precision konstant.

Für die Erstellung eines auf die inkrementell abgegebenen Bewertungen optimierten Agentengremiums wird der Fall betrachtet, dass ein Gremium mit einer festen Anzahl von Agenten zum Einsatz kommt. Die im Gremium vertretenen Agenten und die den Agenten zugeordneten Gewichtungen werden auf der Basis der Benutzerbewertungen angepasst. Ein initial gewähltes

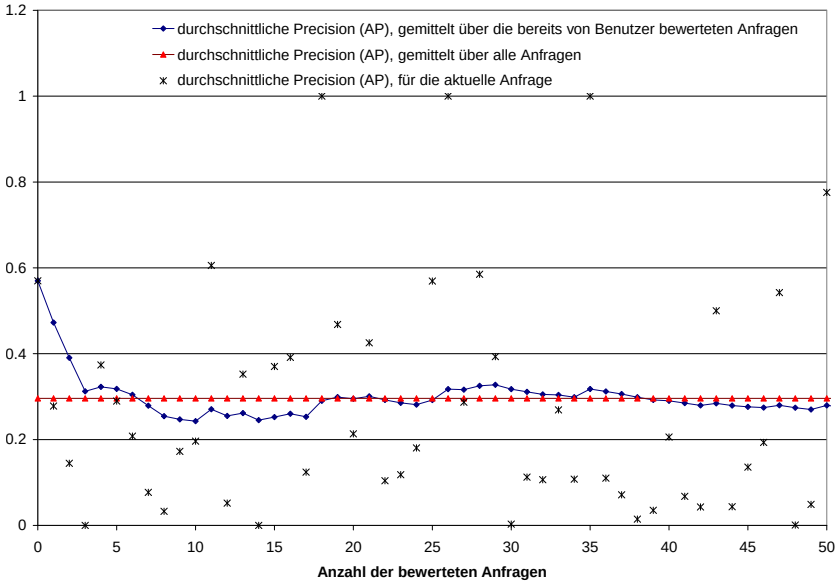


Abbildung 7.21: Die Graphik zeigt die mittlere Precision (MAP), die erreicht wird, wenn in einem Szenario mit inkrementellen Benutzerbewertungen ein statisches Gremium eingesetzt wird. Das Gremium verwendet die Kombinationsstrategien (CombSUM unter Berücksichtigung von Gewichtungen. Das Gremium besteht aus den Filteragenten '0' (58%), '14' (7%), '28' (10%), '34' (7%) und '35' (18%). Die ermittelten Ergebnisse zeigen, dass die pro Anfrage erreichte Filterpräzision stark variiert. Die über alle Anfragen gemittelte durchschnittliche Precision ist konstant, da das Agentengremium unverändert bleibt.

Gremium wird jeweils bei Vorliegen von neuen Bewertungen adaptiert. Zum Anpassen des Agentengremiums wird der folgende Algorithmus verwendet:

1. Das Agentengremium wird um einen Agenten erweitert. Dazu wird aus der Menge der bisher im Gremium nicht vertretenden Agenten der Agent ausgewählt, durch den die Güte des Gremiums am stärksten verbessert werden kann.

2. Unter Verwendung eines Optimierungsalgorithmus werden die Gewichtungen für die Agenten des Gremiums angepasst.

3. Um die ursprüngliche Größe des Agentengremiums wiederherzustellen, wird ein Agent aus dem Gremium entfernt. Es wird der Agent ausgewählt, bei dessen Entfernen die Güte des Gremiums am geringsten sinkt.

4. Unter Verwendung eines Optimierungsalgorithmus werden die Gewichtungen für die Agenten des Gremiums adaptiert.

Algorithmus 5 Anpassen eines Gremiums von Filteragenten an die Bewertungen des Benutzers

---

Füge q_1 in die Menge der bewerten Anfragen Q ein: $Q \leftarrow \{q_1\}$
Ermittle für die Anfrage q_1 ein geeignetes Filteragentengremium $[F]$ (auf der Basis des Algorithmus 2).
while Der Benutzer hat zu einer Anfrage q_{neu} Dokumente bewertet. **do**
  Füge q_{neu} in die Menge der bewerteten Anfragen Q ein: $Q \leftarrow Q \cup q_{neu}$
  Ermittle den Filteragenten s ($s \notin [F]$) und eine Gewichtung v, so dass
    $Q\left([F,s],[\frac{w_0}{1-v},\ldots,\frac{w_n}{1-v},v]\right) \geq Q\left([F,f],[\frac{w_0}{1-\tilde{v}},\ldots,\frac{w_n}{1-\tilde{v}},\tilde{v}]\right) \forall f \in A$
    $0 \leq \tilde{v} < 1$
  Füge den Agenten s mit der Gewichtung v in das Gremium $[F]$ ein:
    $[F] \leftarrow [F,s], [w] \leftarrow [\frac{w_0}{1-v},\ldots,\frac{w_n}{1-v},v]$
  Bestimme ein $[\tilde{w}]$, so dass $Q([F],[\tilde{w}]) \geq Q([F],[w])\ \ \forall w.\ [w] \leftarrow \tilde{w}$
  Ermittle den Agenten f_i aus $[F]$, so dass
    $Q([f_0,\ldots,f_{i-1},f_{i+1},\ldots,f_n],[w_0,\ldots,w_{i-1},w_{i+1},\ldots,w_n]) \geq$
    $Q([f_0,\ldots,f_{j-1},f_{j+1},\ldots,f_n],[w_0,\ldots,w_{j-1},w_{j+1},\ldots,w_n]) \forall j \in [0,n]$
  Entferne den Agenten f_i aus dem Gremium:
    $[F] \leftarrow [f_0,\ldots,f_{i-1},f_{i+1},\ldots,f_n], [w] \leftarrow [w_0,\ldots,w_{i-1},w_{i+1},\ldots,w_n]$
  Bestimme ein $[\tilde{w}]$, so dass $Q([F],[\tilde{w}]) \geq Q([F],[w])\ \ \forall w.\ [w] \leftarrow \tilde{w}$
end while

Evaluation Zur Evaluation des Algorithmus werden die Anfragen des LETOR-TD2004-Datensatzes verwendet. Es wird das Szenario betrachtet, dass der Benutzer zunächst Bewertungen zu drei Anfragen abgibt und auf Basis dieser Bewertungen ein optimiertes Agentengremium erstellt wird. Die Größe des Agentenensembles wird auf vier Agenten festgelegt, da mit vier Agenten eine gute Filterpräzision erreicht werden kann (wie die Untersuchungen im Abschnitt 7.1 zeigen) und eine zu starke Anpassung an die Trainingsdaten verhindert wird. Um das Bewerten von Dokumenten zu neuen Anfragen zu simulieren, wird aus den bisher nicht verwendeten Anfragen des LETOR-TD2004-Datensatzes eine Anfrage ausgewählt und zur Adaption des Agentengremiums genutzt. Um die Vergleichbarkeit mit den in vorhergehenden Abschnitten ermittelten Ergebnissen sicherzustellen, wird das Prinzip der Kreuzvalidierung angewendet (entsprechend der vom Datensatz vorgegebenen Ordner). Dies stellt sicher, dass die Präzision eines erstellten

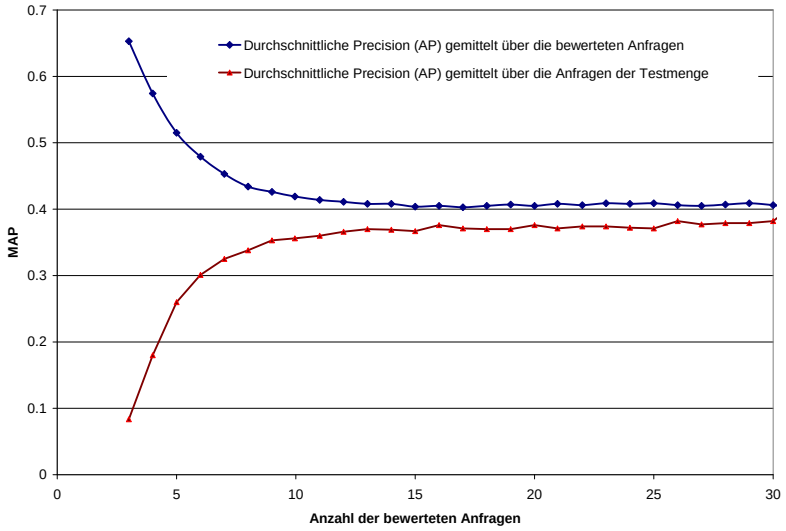


Abbildung 7.22: Die Graphik zeigt die erreichte mittlere Precision, die bei der Adaption eines Agentengremiums an die vom Benutzer abgegebenen Bewertungen erreicht wird.

Filteragentengremiums nicht anhand der Anfragen bewertet wird, die bereits zum Trainieren verwendet wurden.

Die Abbildung 7.22 zeigt die erhaltene Filterpräzision für den Fall, dass ein aus vier Agenten bestehendes Gremium verwendet wird. Die durch das Agentengremium erreichte Filterpräzision zeigt, dass mit dem verwendeten Algorithmus eine gute Anpassung an die Benutzerpräferenzen erreicht wird. So reichen ca. 10 bewertete Anfragen aus, um ein leistungsfähiges Gremium von Filteragenten zusammenzustellen. Die beim inkrementellen Anpassen von Agentengremien erreichte Filterpräzision liegt dabei nur geringfügig unter der Präzision, die beim Trainieren mit einer umfangreichen Menge von Anfragen erreicht wird.

7.3.2 Agentengremien unterschiedlicher Größe

Um den Einfluss der Größe des Agentengremiums auf die erreichte Filterpräzision zu bestimmen, werden anstatt eines Agentengremiums jeweils Gremien unterschiedlicher Größe parallel betrachtet ($[F_1], [F_2], \ldots, [F_n]$), wobei das Agentenensemble $[F_i]$ aus i Agenten zusammengestellt wird). Zur Adaption der Gremien wird der Algorithmus 5 modifiziert: In jedem Adaptionsschritt wird für jedes Gremium geprüft, welcher Agent in das Gremium aufgenom-

men werden sollte, um eine hohe Filterpräzision zu erreichen. Die Filterpräzision des durch die Aufnahme eines Agenten erhaltenen Gremiums wird mit der Filterpräzision der bereits vorhandenen Gremien derselben Größe verglichen. Das Gremium mit der höchsten Filterpräzision wird als Agentengremium (für die betrachtete Gremiengröße) verwendet. Analog zum Erweitern von Agentengremien wird beim Verkleinern von Gremien verfahren. Das Vorgehen im Detail beschreibt der Algorithmus 6.

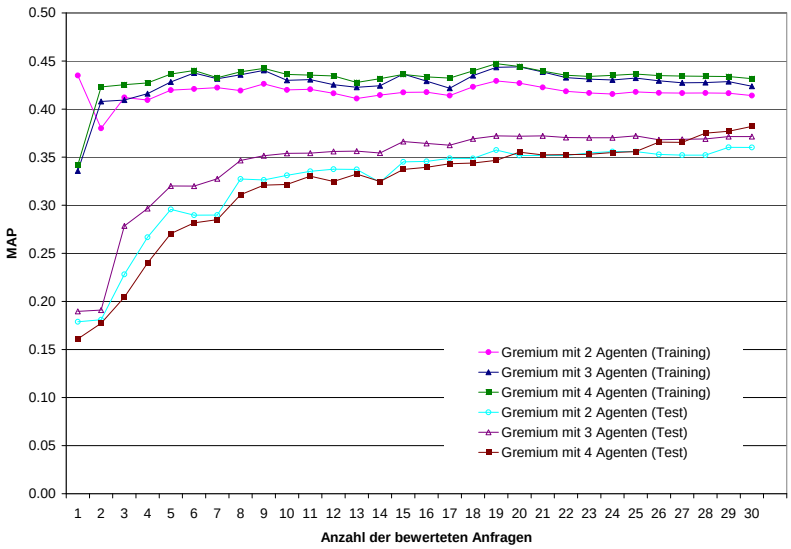


Abbildung 7.23: Die Graphik zeigt die erreichte mittlere Precision (MAP) für Agentengremien unterschiedlicher Größe in Abhängigkeit von der Anzahl der bewerteten Anfragen. Es ist zu erkennen, dass sich größere Agentengremien stärker auf die Trainingsmenge adaptieren als kleine Agentengremien. Eine zu starke Anpassung an eine kleine Trainingsmenge führt zu einer reduzierten Filterpräzision auf der Testmenge. Beim Zusammenstellen von Agentengremien für inkrementell abgegebene Bewertungen muss die Gremiengröße hinreichend groß gewählt werden, um eine gute Filterpräzision zu erreichen („Berücksichtigen verschiedener Kriterien“). Sie darf aber nicht zu groß definiert werden, um eine zu starke Anpassung an die Trainingsmenge zu verhindern („Overfitting“).

Evaluation Das Adaptieren unterschiedlich großer Agentengremien an die Bewertungen des Benutzers verfolgt das Ziel, die gute Anpassungsfähigkeit kleiner Agentengremien mit der guten Filterpräzision größerer (d. h. aus 4-6

Algorithmus 6 Anpassen einer Menge von Gremien aus Filteragenten an die Bewertungen des Benutzers

Man ermittelt ein Gremium $[F_1]$, bestehend aus einem Filteragenten, so dass dieses Gremium auf die Benutzerbewertungen für die Anfrage $q1$ optimiert ist.
for $i = 2$ bis n **do**
Man erstellt ein Gremium $[F_i]$, indem man das Agentengremium $[F_{i-1}]$ um einen Agenten $s \notin [F_{i-1}]$ erweitert. Im neu erstellten Gremium $[F_i]$ habe der Agent s die Gewichtung w. Der Agent s und die Gewichtung w werden so gewählt, dass $P([[F_{i-1}], s]) \geq P([[F_{i-1}], f])$ für alle Filteragenten f und jedes $w \in [0, 1]$.
Man passt die Gewichtungen der Agenten im erstellten Gremium an: $[F_i]$: $[W_i] \leftarrow [W_{i-1}, w] / \, |[W_{i-1}, w]|$
end for
while Der Benutzer hat zu einer neuen Anfrage q_{neu} Dokumente bewertet **do**
for $i = 1$ bis $n - 1$ **do**
Man erstellt ein neues Agentengremium $[F_{neu}]$, indem man das Agentenengremium $[F_i]$ um einen Agenten $s \notin [F_i]$ erweitert. Der Agent s erhält im neu erstellten Gremium die Gewichtung w. Man wählt s und w so, dass $P([[F_i], s]) \geq P([[F_i], f])$ für alle Filteragenten f und jedes $w \in [0, 1]$.
Man passt die Gewichtungen der Agenten in erstellten Gremium an: $[F_neu]$: $[W_{neu}] \leftarrow [W_i, w] / \, |[W_i, w]|$
Man optimiert die Gewichtungen im Gremium $[F_{neu}]$, so dass $P([F_{neu}])$ maximiert wird.
if $P(F_{neu}) \geq P(F_{i+1})$ **then**
Ersetze das bestehende Gremium durch das neue Gremium: $[F_{i+1}] \leftarrow [F_{neu}]$.
end if
end for
for $i = 2$ bis n **do**
Erstelle ein neues Agentengremium $[F_{neu}]$ in dem aus dem Agentengremium $[F_i]$ ein Agent s entfernt wird. Dabei wählt man s so, dass $P(F_i \backslash s) \geq P(F_i \backslash f) \forall f \in F_i$.
Entferne den Agenten s aus dem Gremium: $[F_{neu}] \leftarrow [F_i] \backslash s$.
Optimiere die Gewichte $[W_{neu}]$, so dass $P([F_{neu}])$ maximiert wird.
if $P(F_{neu}) \geq P(F_{i-1})$ **then**
Ersetze das bestehende Gremium durch das neue Gremium: $[F_{i-1}] \leftarrow [F_{neu}]$.
end if
end for
end while

Agenten bestehenden) Gremien zu kombinieren. Die experimentell ermittelten Ergebnisse (Abbildung 7.23) zeigen, dass durch das Betrachten einer Menge von Agentengremien (im Vergleich zu einem einzelnen Gremium) eine geringfügig höhere Filterpräzision erreicht wird. Es ist zu erkennen, dass größere Agentengremien sich besser an die bewerteten Anfragen (Trainingsdaten) anpassen. Allerdings führt dies (insbesondere bei einer sehr kleinen Anzahl von bewerteten Anfragen) zu einer Verschlechterung der Filterpräzision auf der Menge der Testanfragen. Dies deutet darauf hin, dass beim Anpassen eines Agentengremiums an die inkrementellen Bewertungen eines Benutzers, die Gremiengröße langsam erhöht werden sollte.

Hat der Benutzer nur sehr wenige Anfragen (weniger als 10), so lassen die bewerteten Anfragen noch keinen aussagekräftigen Rückschluss auf die Interessen des Benutzers zu. Dies zeigt sich daran, dass die anhand der Bewertungen abgeleiteten Agentengremien und die auf der Testmenge erreichte Filterpräzision stark variieren. Um verlässliche Filterergebnisse zu erhalten, sollte anstatt eines auf die kleine Menge von Bewertungen abgestimmten Agentengremiums, ein unpersonalisiertes, auf die Gesamtheit der Benutzer abgestimmtes Agentengremium verwendet werden. Dies kann beispielsweise ein Gremium sein, dass drei unterschiedliche Agenten mit hoher Filterpräzision kombiniert, z. B. eine termfrequenzbasierte Strategie (BM25), eine reputationsbasierte Strategie (HostRank) und eine Sidemap-basierte Strategie (Agent '26').

Anpassen eines Gremiums an sich ändernde Interessen Beim Erstellen von personalisierten Agentengremien ist zu berücksichtigen, dass sich die Interessen des Benutzers mit der Zeit verändern können. Das verwendete Gütefunktional sollte daher den zeitlichen Verlauf der Bewertungen mit einbeziehen. Man modifiziert die verwendeten Gütefunktionale (MAP, NDCG), indem aktuelle Anfragen eine höhere Gewichtung bei der Berechnung (als länger zurückliegende Bewertungen) erhalten.

$$\text{tMAP}(R_{[q_1,q_2,\ldots,q_n]}) = \frac{\sum_{i=1}^{n} t_i \cdot AP(R_{q_i})}{\sum_{i=1}^{n} t_i} \quad , \; t_i = \frac{i}{i+c}$$

Analog definiert man:

$$\text{tNDCG}(R_{[q_1,q_2,\ldots,q_n]}) = \frac{\sum_{i=1}^{n} t_i \cdot NDCG(R_{q_i})}{\sum_{i=1}^{n} t_i} \quad , \; t_i = \frac{i}{i+c}$$

Dabei bezeichne jeweils q_1 die zuletzt bewertete Anfrage und q_n die letzte noch berücksichtigte Anfrage. In Abhängigkeit von der jeweiligen Domäne und der erwarteten Änderung von Interessen können unterschiedliche Gewichtungsmodelle ausgewählt werden.

Eine wichtige Fragestellung im Bereich der personalisierten Informationssysteme ist, ob für sehr spezielle Benutzerinteressen auf Basis der verfügbaren Filteragenten ein adäquates Gremium zusammengestellt werden kann. Wird trotz einer Adaption der verwendeten Filteragentengremien nur eine unzureichende Filterpräzision auf den Trainingsdaten (d. h. den vom Benutzer bewerteten Anfragen) erreicht, so sollte analysiert werden, welche Kriterien der Benutzer zum Bewerten der Relevanz von Dokumenten verwendet. Werden diese nicht hinreichend durch die bisher verfügbaren Agenten berücksichtigt, so sollten zusätzliche Filteragenten definiert werden, die auf der Basis der für den Benutzer gewünschten Kriterien die Dokumente bewerten. Die neu hinzukommenden Agenten können beim Zusammenstellen neuer Agentenensembles bzw. bei der Adaption bestehender Agentenensembles herangezogen werden. Dies ermöglicht es, auch individuelle Kriterien bzw. Strategien bei der Bewertung von Dokumenten einzubeziehen. Der Fall, dass die vom Informationssystem verwendeten Kriterien nicht zu den Präferenzen der Benutzer passen, kann insbesondere dann auftreten, wenn das System von neuen Benutzergruppen eingesetzt wird. Durch die Offenheit des Informationssystems (z. B. durch die Verwendung einer Multi-Agentenplattform) können neue Bewertungsstrategien in das Informationssystem integriert werden, die die individuellen Präferenzen von Benutzern berücksichtigen.

Eine Voraussetzung für das Personalisieren eines Agentengremiums ist, dass die Präferenzen des Benutzers bekannt sind. Für neue Benutzer, über die nur wenige Informationen verfügbar sind, wird ein auf die Interessen des durchschnittlichen Benutzers abgestimmtes Agentengremium gewählt. Dieses wird auf Basis der bei der Nutzung des Informationssystems ermittelten impliziten und expliziten Benutzerbewertungen an die individuellen Benutzerpräferenzen angepasst.

7.3.3 Integration eines neuen Agenten

Stehen in einem Informationssystem neue Bewertungsagenten zur Verfügung, so sollte das System Mechanismen anbieten, diese neuen Agenten zu integrieren. Im Kontext von auf Agentengremien basierenden Informationssystemen muss geprüft werden, ob ein neuer Agent ein bestehendes Gremium verbessern kann. Dafür muss nicht nur die Filterpräzision des neu hinzukommenden Agenten ermittelt werden, sondern auch analysiert werden, wie gut die neue Filterstrategie die bestehenden Agenten ergänzt.

Das im Algorithmus 3 vorgestellte Vorgehen ermöglicht es, inkrementell anhand der Bewertungen des Benutzers das verwendete Agentengremium anzupassen. Der Algorithmus erlaubt dabei insbesondere neu hinzukommende Agenten bei der Adaption des Agentengremiums zu berücksichtigen.

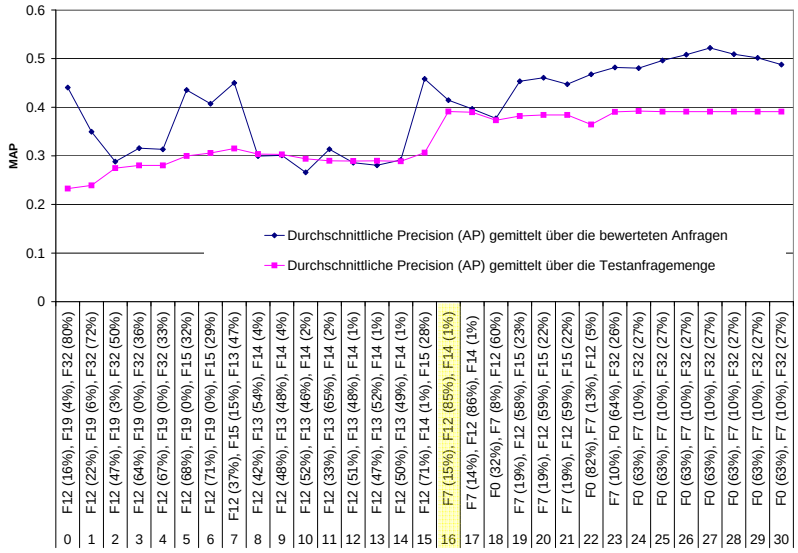


Abbildung 7.24: Die Graphik zeigt, wie sich die mittlere Precision der Gremien verändert, wenn neue Agenten hinzugefügt werden. Es wird dafür der Fall betrachtet, dass für die Anfragen 1 – 15 der Filteragent '0' (BM25) nicht zur Verfügung steht und erst nach der 15. Anfrage in ein Gremium aufgenommen werden kann.

Abbildung 7.24 zeigt beispielhaft die Integration eines neuen Agenten. Es wird das Szenario betrachtet, dass auf Basis inkrementell abgegebener Bewertungen ein Agentengremium zusammengestellt wird. Aus der Menge der verfügbaren Filteragenten wird die Strategie '0' entfernt. Nachdem der Benutzer 15 Anfragen bewertet hat, steht die Strategie '0' als neuer Agent zur Verfügung. Da unter Berücksichtigung dieses (neuen) Filteragenten die Filterpräzision auf der Menge der bewerteten Anfragen verbessert werden kann, wird die Filterstrategie '0' in das Agentengremium aufgenommen. Es wird damit die Filterpräzision sowohl für die Menge der bewerteten Anfragen (Trainingsmenge) als auch für die Menge der Testanfragen gesteigert. Das Bespiel zeigt, dass durch den verwendeten Algorithmus zur Adaption des Agentengremiums neue Filterstrategien integriert werden können.

7.3.4 Ausfall eines Agenten

Werden in Informationssystemen Gremien von Agenten zum Filtern eingesetzt, so kann es zum Ausfall von Agenten kommen. Ein Informationssystem sollte daher Methoden unterstützen, um ausgefallene Agenten zu ersetzen.

Bei der Verwendung von Agentengremien können bei Ausfall eines einzelnen Agenten im Allgemeinen weiterhin Informationen gefiltert werden. Der Ausfall eines Agenten führt aber zu einer Verringerung der Filterpräzision. Um eine möglichst hohe Filterpräzision zu erreichen, sollten ausgefallene Agenten durch zur Verfügung stehende alternative Agenten ersetzt werden.

Um die in einem Gremium ausgefallenen Agenten durch geeignete Agenten zu ersetzen, kann der leicht modifizierte Algorithmus 2 verwendet werden. Umfasst das auf die Präferenzen des Benutzers (aufgrund des Ausfalls eines Agenten) weniger als die angestrebte Anzahl von Agenten, so wird auf das Entfernen von Agenten verzichtet. Es wird aus der Menge der verfügbaren Agenten der Agent in das Gremium aufgenommen, durch den die Filterpräzision am stärksten verbessert werden kann.

Ein Beispiel für das Behandeln des Ausfalls eines Agenten zeigt die Abbildung 7.25. Es wird das Szenario betrachtet, dass nach dem Bewerten der 15. Anfrage der Agent '20' ausfällt. Das Beispiel zeigt, dass durch die Integration der Bewertungsstrategie '0' die Filterpräzision auf hohem Niveau gehalten werden kann. Der Ausfall bleibt für den Benutzer transparent.

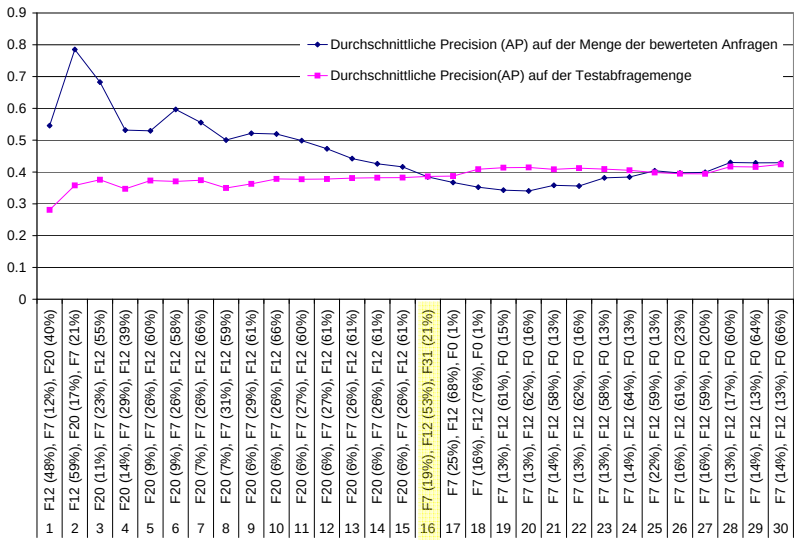


Abbildung 7.25: Die Graphik zeigt die Auswirkungen auf die mittlere Precision (MAP) für den Fall, dass der Filteragent '20' nach der 15. bewerteten Anfrage ausfällt.

Diskussion

Die durchgeführten Untersuchungen zeigen, das anhand von inkrementell vom Benutzer abgegebener Bewertungen effektiv Agentengremien zum Bewerten der Relevanz von Dokumenten zusammengestellt werden können. Allerdings besteht bei einer geringen Anzahl von bewerteten Anfragen die Gefahr, dass eine zu starke Anpassung an die bewerteten Anfragen erfolgt, die nicht immer repräsentativ für die Interessen des Benutzers sind. Für eine Anpassung an die Präferenzen des Benutzers werden beim verwendeten LETOR-TD2004-Datensatz 10-15 bewertete Anfragen benötigt.

Als Strategien zum Anpassen des Informationssystems an die Präferenzen des Benutzers wurden neben Gremien von Filteragenten auch Gruppen von unterschiedlich großen Agentengremien betrachtet. Die Ergebnisse zeigen, dass durch die Verwendung einer Gruppe unterschiedlich großer Agentengremien eine hohe Filterpräzision auf der Menge der bewerteten Anfragen (Trainingsmenge) erreicht werden kann. Allerdings ist dies bei einer kleinen Anzahl von bewerteten Anfragen nicht mit einer verbesserten Filterpräzision auf der Testmenge verbunden.

Durch das inkrementelle Anpassen des Agentengremiums kann auf Veränderungen in der Menge der verfügbaren Agenten reagiert werden. Es wurde gezeigt, dass neue Filterstrategien in das Gremium aufgenommen werden, wenn dadurch die Filterpräzision verbessert werden kann. Darüber hinaus wurde gezeigt, wie der Ausfall eines Filteragenten ausgeglichen werden kann, ohne dass der Benutzer den Ausfall eines Filteragenten bemerkt. Da der verwendete Optimierungsalgorithmus einen geringen Ressourcenbedarf aufweist, kann die Adaption des Agentengremiums jeweils online (bei Änderungen der Menge der verfügbaren Agenten) ausgeführt werden.

7.4 Personalisierte Bewertungsmaße

In den bisher durchgeführten Untersuchungen wurden zum Messen der Filterpräzision jeweils die zusammen mit dem LETOR-TD2004-Datensatz verfügbaren Referenzbewertungen verwendet. Da Benutzer aber unterschiedliche Präferenzen und Erfahrungen haben, sind für die Benutzer auch unterschiedliche Informationen relevant. Beim Zusammenstellen von Filteragentengremien sollten daher auch auf die individuellen Präferenzen des Benutzers abgestimmte Bewertungen verwendet werden können.

Eine Möglichkeit, individuelle Präferenzen auf Basis des LETOR-Datensatzes zu simulieren, besteht darin, einen Bewertungsagenten als Referenzagenten auszuwählen. Es wird ein Agentengremium zusammengestellt,

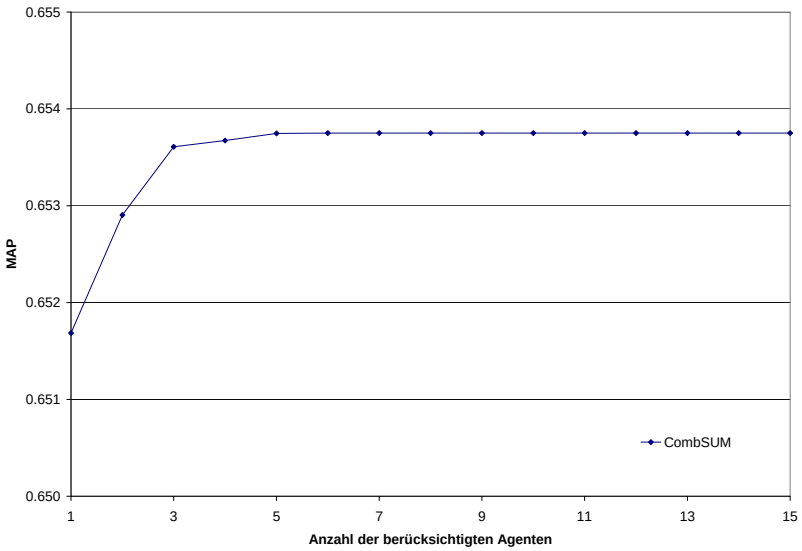


Abbildung 7.26: Die Graphik zeigt das schrittweise Anpassen eines Agentengremiums an die „Interessen“ des Agenten ‘0’ (BM25). Es wird dabei Algorithmus 3 verwendet.

das die Bewertungsstrategie des ausgewählten Referenzagenten möglichst gut nachbildet. D. h. man stellt ein Agentengremium zusammen, das auf die Bewertungsstrategie das Referenzagenten optimiert ist („Lernen einer Filterstrategie“).

Beispielhaft wird analysiert, wie gut für die Filterstrategie ‘0’ (BM25) ein Agentengremium zusammengestellt werden kann. Das Auswählen der Agenten und das Ermitteln geeigneter Gewichtungen erfolgt unter Verwendung des Algorithmus 3. Die beim Lernen der Filterstrategie BM25 erreichten Filterpräzision zeigt die Abbildung 7.26.

Das Beispiel zeigt, dass auch für alternative Referenzstrategien geeignete Gremien von Filteragenten erstellt werden können. Im Rahmen von Informationssystemen können damit die verfügbaren Filterstrategien so kombiniert werden, dass sie auf die individuellen Präferenzen des Benutzers optimiert sind. Zu beachten ist, dass die bei unterschiedlichen Referenzstrategien erreichte mittlere Precision (MAP) nicht direkt miteinander vergleichbar sind, da die A-Priori-Werteverteilung von MAP vom Anteil der relevanten Dokumente in der Gesamtdokumentenmenge abhängt.

7.5 Zusammenfassung

In diesem Kapitel wurden verschiedene Algorithmen entwickelt und evaluiert, mit denen auf das Szenario und die Präferenzen des Benutzers abgestimmte Agentengremien zusammengestellt werden können. Dabei wurden sowohl unterschiedliche Anwendungsfälle als auch unterschiedliche Gütekriterien für die Agentengremien analysiert. Es wurde gezeigt, dass sowohl auf Basis einer Menge von bewerteten Anfragen („Batch-Learning“) als auch auf Basis inkrementell abgegebener Benutzerbewertungen Gremien zusammengestellt werden können.

Darüber hinaus wurden verschiedene Ansätze zum Gruppieren von Agenten und von Anfragen vorgestellt. Die durch das Gruppieren gewonnenen Informationen ermöglichen das effiziente Zusammenstellen leistungsfähiger Agentengremien. Die Methode, mit der die besten Ergebnisse erreicht wurden, besteht im Gruppieren von Anfragen bezüglich des am besten für jede Anfrage geeigneten Agenten. Die bei diesem Ansatz gebildeten Agentengremien berücksichtigen die besonders leistungsfähigen Agenten und stellen außerdem sicher, dass für jede Anfrage der Trainingsmenge ein geeigneter Agent im Agentenensemble vorhanden ist. Dies stellt die Vielfältigkeit der im Agenten-Ensemble vertretenden Filterstrategien sicher.

In den durchgeführten Untersuchungen wurde gezeigt, dass durch das Kombinieren mehrerer Filteragenten sehr leistungsfähige Filterstrategien realisiert werden können. Ohne die von den einzelnen Filteragenten verwendeten Algorithmen zu kennen, können auf die individuellen Präferenzen des Benutzers abgestimmte Filter erstellt werden („personalisierte Filterung“). Im Gegensatz zur statischen Verknüpfung von Filterstrategien ermöglicht die Verwendung von Agentengremien, die dynamische Integration neuer Filterstrategien und das Ersetzen fehlerhafter Agenten. Es wurde gezeigt, dass durch die geeignete Kombination von Filterstrategien verlässliche, hochqualitative Filterergebnisse erreicht werden, die eine deutlich bessere Filterqualität erreichen, als jeder einzelne Filteragent.

Im folgenden Kapitel wird anhand der Fallstudie eines agentenbasierten Informationssystems gezeigt, wie die Verfahren in der Praxis eingesetzt werden können.

Kapitel 8

Fallstudie: Personalisierte Informationsagenten

Auf der Basis der entwickelten Architektur wurden für unterschiedliche Anwendungsgebiete Informationssysteme entwickelt. Dies ermöglicht es, die Stärken und Schwächen der Architektur sowie die entwickelten Algorithmen in der Praxis zu untersuchen. Das Kapitel ist wie folgt gegliedert: Zunächst wird ein Überblick über die implementierten Informationssysteme gegeben und auf die Besonderheiten der jeweiligen Anwendungsbereiche eingegangen. Im Anschluss daran werden die in den Informationssystemen eingesetzten Konfigurationen und Algorithmen erläutert. Die Systeme werden anhand der durch Analyse des Benutzerverhaltens und der im Rahmen von Benutzerbefragungen erhobenen Daten evaluiert.

8.1 Überblick über die entwickelten Systeme

Die implementierten Informationssysteme sind jeweils auf ein spezifisches Anwendungsszenario optimiert. Neben Systemen für die umfassende Informationsversorgung wurden auf eine beschränkte Domäne spezialisierte Systeme realisiert.

- Das Informationssystem PIA (**P**ersonalisierte **I**nformations-**A**genten) stellt themenbereichsübergreifende Informationen zur Verfügung und integriert neben Informationsquellen für aktuelle Nachrichten (Politik, Wirtschaft, Sport) auch Suchmaschinen (z. B. YAHOO[1]) und fachspezifische Informationsquellen aus den Bereichen Informatik und Elektrotechnik.

[1] `http://www.yahoo.com/`

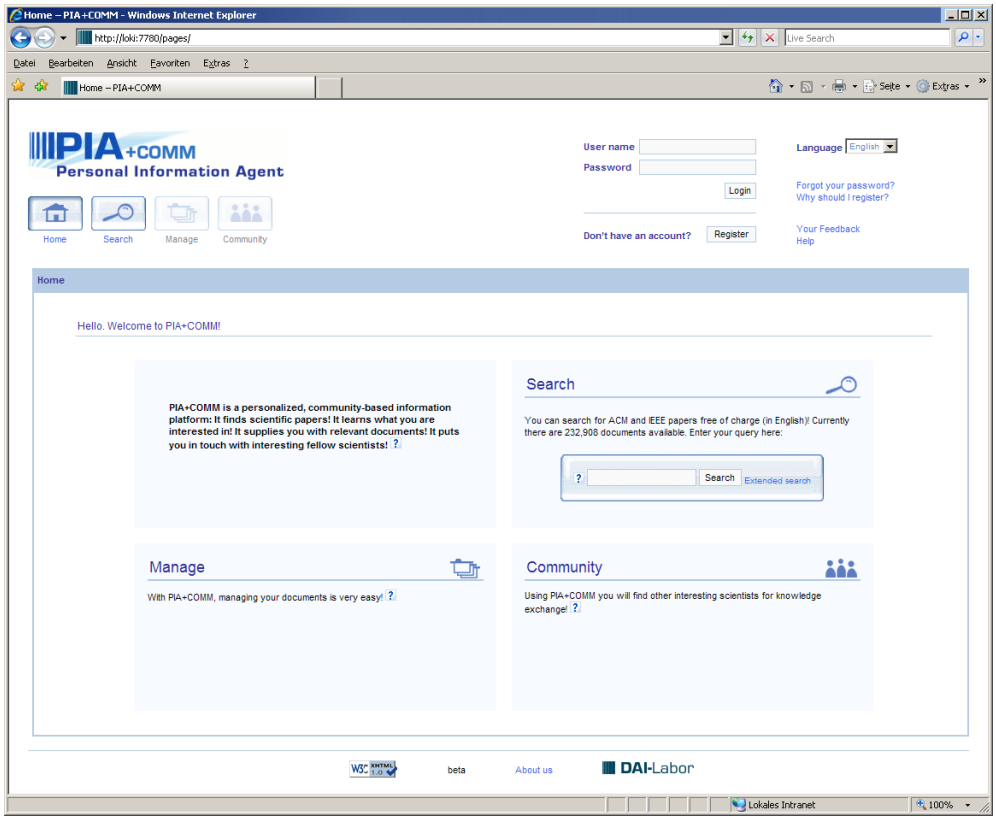


Abbildung 8.1: Die Abbildung zeigt die Startseite des Informationssystems für Wissenschaftler PIA+COMM (oben) und die Startseite des Persönlichen Zeitungsdienstes PZD (unten).

- Aktuelle Informationen aus dem Bereich Gesundheit und Fitness stellt das System SMART HEALTH ASSISTANT (SHA)[2] bereit. Dabei werden Dokumente aus verschiedenen Quellen gefiltert und zu einem personalisierten Newsletter aggregiert.

- Der **P**ERSÖNLICHE **Z**EITUNGS**D**IENST (**PZD**) ist Teil des Projektes **S**EAMLESS **H**OME **S**ERVICES (**SHS**) und stellt neben aktuellen Nachrichten aus Politik, Wirtschaft und Sport auch Empfehlungen für das aktuelle Fernsehprogramm zusammen. Dazu werden die durch den elektronischen Programmführer (EPG) verfügbaren Daten mit Informationen aus Datenbanken und Online-TV-Zeitschriften angereichert und auf Basis des Interessenprofils des Benutzers personalisierte Empfehlungen ermittelt. Das System bietet die Möglichkeit, die als besonders empfehlenswert bewerteten Sendungen automatisch aufzuzeichnen, so dass der Benutzer diese Sendungen zeitversetzt ansehen kann.

- Eine Unterstützung des wissenschaftlichen Arbeitens von Forschern bietet das System PIA+COMM. PIA+COMM integriert mehrere Datenbanken für wissenschaftliche Dokumente und bietet darüber hinaus vielfältige Funktionen zum Bilden von Gemeinschaften („Communities“) sowie zur Erleichterung des kollaborativen Arbeitens (z. B. durch das Bereitstellen von gemeinsam nutzbaren Netzwerklaufwerken).

Die implementierten Systeme basieren alle auf der im Kapitel 4 vorgestellten Architektur. Entsprechend der jeweiligen Zielgruppe und den Anforderungen an die Filterergebnisse werden auf das Anwendungsszenario optimierte Datenquellen, Bewertungsagenten und Strategien zum Zusammenführen von Bewertungen eingesetzt. Die Offenheit und Flexibilität der Architektur stellt sicher, dass eine Anpassung an die projektspezifischen Anforderungen mit geringem Aufwand möglich ist.

8.2 Implementierung und Besonderheiten der Systeme

Die Implementierung der Informationssysteme erfolgte auf Basis der Programmiersprache JAVA unter Verwendung von Komponenten zur Realisierung webbasierter Benutzerschnittstellen (wie APACHE TOMCAT[3], JBOSS[4],

[2] http://www.smart-health-assistant.de
[3] http://tomcat.apache.org/
[4] http://www.jboss.com/

JETTY[5]). Zum Filtern und Bewerten der Dokumente kommen sowohl Standardkomponenten (wie z. B. LUCENE[6] als auch speziell für das jeweilige Anwendungsszenario abgestimmte Filteragenten zum Einsatz. Die Verwendung einer einheitlichen Systemarchitektur, der Einsatz standardisierter Kommunikationsprotokolle und das gemeinsame Nutzen von Systemkomponenten („Filteragenten", „Extraktionsagenten") von mehreren Informationssystemen ermöglicht vielfältige Synergien und die systemübergreifende Nutzung von Ressourcen.

Da sich die realisierten Informationssysteme bezüglich der jeweiligen Benutzerzielgruppen und den Nutzungsszenarien unterscheiden, wurde für jedes der Systeme eine spezielle Benutzeroberfläche entwickelt. Darüber hinaus verwendet jedes Informationssystem spezifische, auf die jeweilige Domäne abgestimmte Informationsquellen. Die dabei jeweils unterstützten Datenformate sind vielfältig: Sie reichen von semantisch annotierten Daten (z. B. von der Filmdatenbank IMDB[7]) über semi-strukturierte Internetseiten bis zu unformatierten Nachrichten (z. B. Plain Text-Emails). Daraus ergibt sich, dass in Abhängigkeit von den Informationsquellen auch unterschiedliche Algorithmen zur Aufbereitung und Filterung der Dokumente eingesetzt werden. Es müssen teilweise fehlende Metadaten (z. B. Zugehörigkeit von Dokumenten zu einem Themenbereich) generiert sowie Fehler in Dokumenten behoben werden (z. B. bei von der Quelle falsch angegebener Zeichen-Codierung). Im Gegensatz zu Dokumenten aus Benchmarkdatensätzen ist bei der Filterung von Dokumenten aus heterogenen Quellen sicherzustellen, dass fehlerhafte und unvollständige Dokumente erkannt und behandelt werden. Insbesondere bei der Integration von Suchmaschinen und Blogs (z. B. beim PIA-System) kommt einer sorgfältigen Prüfung von Dokumenten, bezüglich des Vorhandenseins von Spam, vorgetäuschten Inhalten („Cloaking") und fehlerhaft kodierten Dokumenten eine große Bedeutung zu, da die meisten Benutzern diesen Dokumententyp als sehr störend empfinden. Ein frühzeitiges Erkennen und Herausfiltern dieser Dokumente hilft, Ressourcen zu sparen und die Filterqualität zu verbessern. Werden in einem Informationssystem ausschließlich vertrauenswürdige Quellen (z. B. redaktionell gepflegte Datenbanken) verwendet (wie z. B. bei PIA+COMM), so kann im Allgemeinen auf einen Spam-Filter verzichtet werden.

Um die Filterung an die Besonderheiten des jeweiligen Informationssystems anzupassen, wird die Menge der jeweils eingesetzten Filter- bzw. Bewertungsagenten sowie die Menge der Agenten zum Zusammenstellen der

[5] `http://www.mortbay.org/`
[6] `http://lucene.apache.org/`
[7] `http://www.imdb.com/`

Filterergebnismenge an die speziellen Anforderungen des Anwendungsbereiches angepasst. So werden beispielsweise in den Systemen SHA und PZD zum Zusammenstellen von personalisierten Newslettern, komplexe Algorithmen zum Prüfen der einzelnen Dokumente eingesetzt, die zuverlässig das Erkennen von Spam sowie von nicht mehr verfügbaren Dokumenten ermöglichen. In Informationssystemen, bei denen der Benutzer schnell Filterergebnisse zu einer Anfrage erwartet (wie z. B. PIA+COMM), müssen dagegen schnelle Filteralgorithmen sowie Strategien zum Überbrücken der Wartezeit eingesetzt werden (z. B. Informationen zum Status des Filterprozesses, Vorschau auf Filterergebnisse). Sehr aufwändige Filterstrategien werden bei der zeitkritischen Filterung ausgeschlossen.

8.3 Strategien zum Analysieren des Informationssystems

Zur Evaluation der realisierten Informationssysteme wurden die von den Systemen bereitgestellten Dokumente bezüglich der Filterqualität analysiert sowie das Laufzeitverhalten und der Ressourcenbedarf untersucht. Dazu wurden die Protokolldaten („log data“) der realisierten Informationssysteme ausgewertet und in Benutzerbefragungen die Meinung von Anwendern erhoben (in Kooperation mit dem Zentrum Technik und Gesellschaft der TU Berlin[8])[HJ07]. Da die Zielgruppe des PIA+COMM-Systems Studenten und Wissenschaftler sind, wurden 32 Studenten und Mitarbeiter der Fakultät IV der TU Berlin (Elektrotechnik und Informatik) befragt. Es wurden dabei jeweils 60- bis 90minütige Interviews durchgeführt, wobei halbstandardisierte Interview-Leitfäden verwendet wurden (vgl. [Fli00]). Im Rahmen der Benutzerbefragung wurde untersucht, welche Funktionen des Informationssystems als besonders hilfreich bewertet werden und inwieweit der Benutzer in komplexe Filterprozesse einbezogen werden sollte. Im Gegensatz zur Evaluation der eingesetzten Algorithmen auf Basis von Benchmarkdatensätzen, bei der vorrangig die Präzision und die Aufwandsklasse analysiert wird, finden bei der Evaluation der Informationssysteme auf Basis von Benutzerbefragungen die Benutzerschnittstelle, die Nachvollziehbarkeit und die subjektive Dokumentenqualität besondere Berücksichtigung.

In diesem Kapitel werden ausgewählte Aspekte der implementierten Informationssysteme detailliert vorgestellt. Den Schwerpunkt bilden dabei die Algorithmen zur Optimierung der Anfragen sowie die Strategien zum Bewerten von Dokumenten.

[8] `http://www.ztg.tu-berlin.de/`

8.4 Erstellen und Optimieren von Anfragen

Um den Benutzer beim Erstellen und Verfeinern von Suchanfragen zu unterstützen, bieten die Informationssysteme PIA und PIA+COMM eine Komponente, die den Benutzer bei der Optimierung von Anfragen unterstützt. Im Folgenden werden die implementierten Verfahren zur Optimierung von Anfragen und zum Erstellen von Anfragen auf Basis von Benutzerbewertungen anhand von Beispielen erläutert und evaluiert.

8.4.1 Optimieren von Anfragen

Die von einem Informationssystem ermittelten Filterergebnisse hängen stark von der verwendeten Anfrage ab. Die Verwendung von fehlerhaften oder unvollständigen Anfragen kann zu Filterergebnissen führen, die nicht die Erwartungen des Benutzers erfüllen. Im Rahmen des entwickelten Informationssystems wurden daher verschiedene Methoden entwickelt, mit denen Fehler erkannt und Anfragen optimiert werden können. Die Verfahren werden am Beispiel der Systeme PIA und PIA+COMM veranschaulicht.

Erkennen von Fehlern

Ein häufig auftretendes Problem bei der Verwendung schlüsselwortbasierter Suchanfragen stellen Schreibfehler dar. Um Schreibfehler zu vermeiden, ohne den Benutzer in der Wahl der Begriffe einzuschränken, kann eine Vorschlagsfunktion für Begriffe integriert werden, die zu den Benutzereingaben passende Begriffe vorschlägt. Das PIA+COMM-System bietet eine (unter Verwendung von AJAX implementierte) Funktion, die inkrementell zu den Eingaben des Benutzers passende Wörter und Phrasen vorschlägt. Der Benutzer hat die Möglichkeit, einen der vorgeschlagenen Begriffe zu übernehmen, Vorschläge zu modifizieren oder auch die Suchbegriffe selbst zu definieren. Zum Generieren von Begriffsvorschlägen werden Wörterbücher (z. B. BEOLINGUS[9]) und domänenspezifische Begriffssammlungen (z. B. DMOZ) herangezogen.

Die Funktion zum Vervollständigen von Suchbegriffen wird von der Mehrheit der Benutzer positiv bewertet, da diese effektiv hilft, Fehler zu vermeiden. Zudem ist eine ähnliche Funktionalität vielen Benutzern bereits aus populären Anwendungsprogrammen (z. B. Web-Browsern) vertraut. Das Vorschlagen von Phrasen auf Basis der Benutzereingaben hilft dem Benutzer, längere Suchanfragen zu verwenden, mit denen die Interessen des Benutzers präziser beschrieben werden. Die zum Vervollständigen von Anfragen ermittelten Vorschläge sollten jeweils das Suchszenario und die Anwendungsdomä-

[9] http://dict.tu-chemnitz.de/

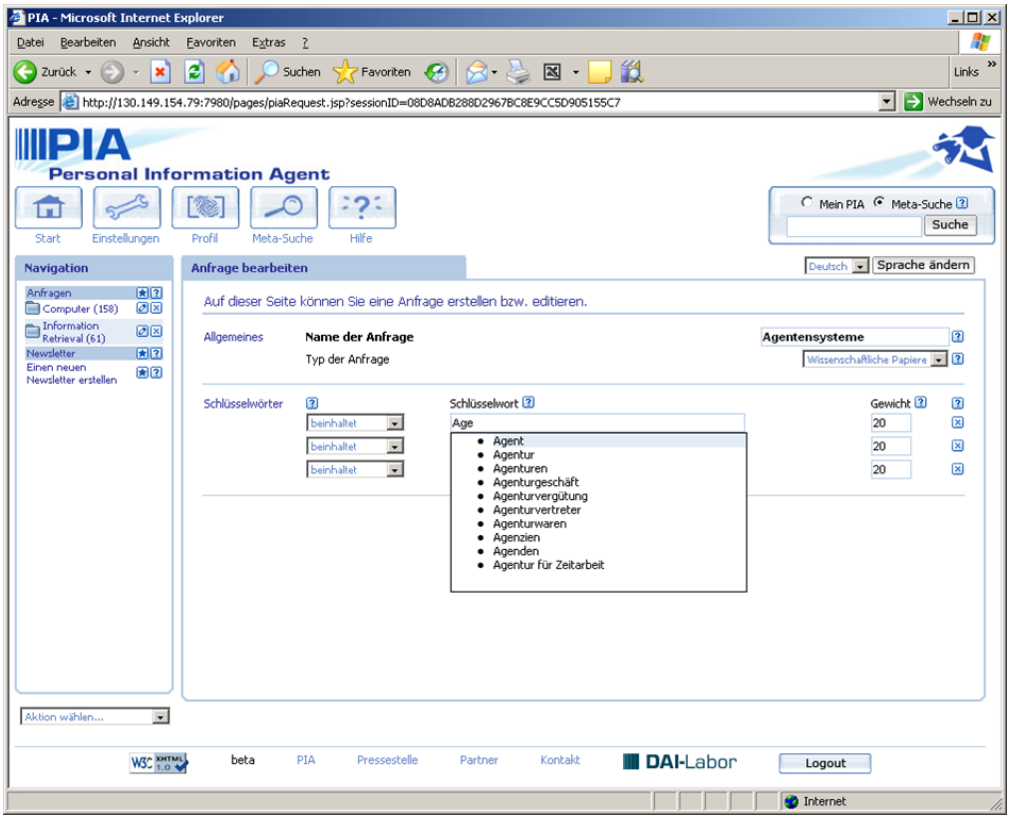


Abbildung 8.2: Die Abbildung zeigt das Vervollständigen von Benutzereingaben im Informationssystem PIA.

ne berücksichtigen. Um aktuelle Begriffe beim Ermitteln von Vorschlägen zu berücksichtigen, können kollaborative Algorithmen und Verfahren zur Extraktion von Phrasen aus aktuellen Dokumenten (z. B. Nachrichten) zum Einsatz kommen.

Die Funktion zum Vervollständigen von Benutzereingaben, wie sie im PIA-System eingesetzt wird, zeigt die Abbildung 8.2.

Extraktion von Begriffen aus Lexika

Um aus einem Begriff bestehende Suchanfragen durch zusätzliche Wörter und Phrasen zu verfeinern, schlägt das Informationssystem dem Benutzer Begriffe vor, die mit dem initial definierten Wort in enger Beziehung stehen. Eine Möglichkeit, dieses Verfahren zu implementieren, besteht in der Extraktion von charakteristischen Wörtern aus Lexika-Artikeln, die zu dem vom Benutzer definierten Suchbegriff passen.

Das PIA-System verwendet als Nachschlagewerk die freie Enzyklopädie WIKIPEDIA[10]. Um zu einem Suchbegriff verwandte Wörter und Phrasen zu ermitteln, wird zunächst der zum Begriff passende WIKIPEDIA-Artikel extrahiert (falls dieser existiert). Aus dem Lexikon-Artikel werden unter Verwendung einer Heuristik Begriffe und Phrasen ermittelt. Es werden dabei solche Wortgruppen als besonders relevant bewertet, zu denen ein WIKIPEDIA-Artikel existiert und die am Beginn des Artikels auftreten.

Eine Voraussetzung für die Anwendung des Verfahrens ist, dass zum benutzerdefinierten Suchbegriff ein passender WIKIPEDIA-Artikel verfügbar ist. Für den Fall, dass zu dem Begriff kein Artikel existiert, wird das Verfahren nicht angewendet.

Ein Beispiel für das Ermitteln von relevanten Begriffen auf Basis von WIKIPEDIA-Artikeln zeigt die Abbildung 8.3. Die Analyse der Ergebnisse dieses Verfahrens zeigt, dass bei der Extraktion von Begriffen aus WIKIPEDIA überwiegend für den Benutzer hilfreiche Begriffe ermittelt werden. Aufgrund der unterschiedlichen Struktur von WIKIPEDIA-Artikeln (z. B. Begriffsklärung, detaillierte Erklärung eines Begriffes) können sich die Ergebnisse jedoch von zu Begriff zu Begriff unterscheiden. Ein Vorteil der Verwendung von WIKIPEDIA besteht darin, dass Artikel in mehreren Sprachen verfügbar sind und somit dieses Verfahren jeweils an die vom Benutzer gewählte Sprache angepasst werden kann.

Das Extrahieren von Begriffen und Phrasen aus WIKIPEDIA-Artikeln ist vor allem dann sinnvoll, wenn der Benutzer eine aus nur einem Begriff bestehende Anfrage definiert hat und seine Anfrage verfeinern möchte. Die unter Verwendung von WIKIPEDIA-Artikeln extrahierten Begriffe helfen dem Benutzer, eine Anfrage genauer zu spezifizieren und Unklarheiten zu erkennen.

Um auch bei dem Vorliegen mehrerer Suchbegriffe, Vorschläge ermitteln zu können, wird das Verfahren modifiziert: Man extrahiert zu den vorliegenden Begriffen die WIKIPEDIA-Artikel (soweit vorhanden) und führt eine Schnittmengenanalyse der auftretenden Begriffe durch. Dabei werden die Wörter auf ihren Wortstamm reduziert [Por97] und Stopp-Wörter ignoriert. Die Begriffe, die in mehreren extrahierten Artikeln auftreten und für die WIKIPEDIA-Artikel existieren, werden dem Benutzer zum Optimieren der Anfrage vorgeschlagen. Das Verfahren ermöglicht es, zu den vom Benutzer definierten Suchbegriffen, gemeinsame Oberbegriffe und Anwendungsbereiche zu ermitteln. Ein Beispiel für die Ermittlung von Begriffen auf Basis einer Schnittmengenanalyse von WIKIPEDIA-Artikeln zeigt die Abbildung 8.4.

[10] `http://www.wikipedia.org/`

Abbildung 8.3: Die Abbildungen demonstrieren das Ermitteln von Begriffen zum benutzerdefinierten Begriff `FIPA`. Die obere Abbildung zeigt den WIKIPEDIA-Artikel zum Begriff `FIPA` (08.05.2006), die untere Abbildung die durch den PIA Anfrageoptimierungsagenten ermittelten Begriffe.

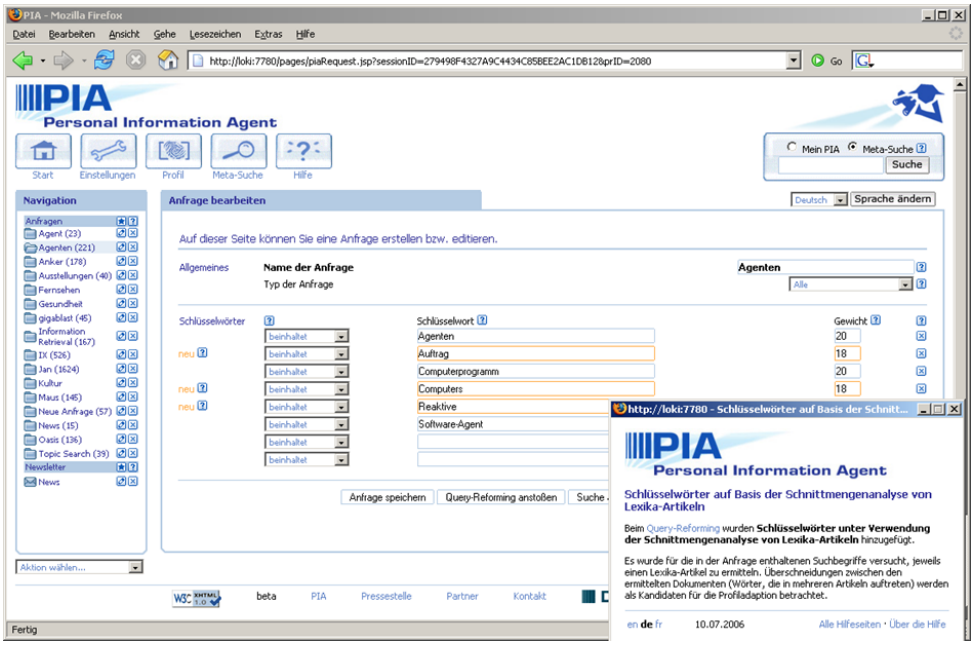


Abbildung 8.4: Die Abbildung zeigt die unter Verwendung einer Schnittmengenanalyse zwischen WIKIPEDIA-Artikeln ermittelten Suchbegriffen für eine aus den Begriffen „Agenten“, „Computerprogramm“ und „Software-Agent“ bestehende Anfrage.

Anfrageoptimierung auf Basis von Social Tagging-Diensten

Social Tagging-Dienste bieten die Möglichkeit, Bookmarks online zu verwalten, mit Schlagworten („Tags“) zu versehen und mit anderen Benutzern gemeinsam zu verwenden. Die von Benutzern vergebenen Tags können bei der Dokumentensuche genutzt werden, um für den Benutzer hilfreiche Suchbegriffe zu ermitteln [HKGM08].

Das System PIA verwendet zum Ermitteln von Begriffen die Funktionalität „related Tags“ des Social Tagging-Dienstes `del.icio.us`[11]. Dabei werden die Tags ermittelt, die von `del.icio.us`-Benutzern häufig zusammen mit einem gegebenen Begriff verwendet wurden.

Die Abbildung 8.5 veranschaulicht das Generieren von Vorschlägen für den Begriff `SOA`. Die Verwendung von `del.icio.us` ermöglicht es, die umfangreichen, von Benutzern generierten Inhalte für die Optimierung von Suchanfragen heranzuziehen. Die große Anzahl von neuen Tags (> 20000 neue Tags pro Monat [WZB08]) ermöglicht es, aktuelle Trends zu erfassen und populäre Begriffe zu identifizieren. Die in `del.icio.us` verwen-

[11] http://del.icio.us

Abbildung 8.5: Die obere Abbildung zeigt `del.icio.us`-Webseite zum Begriff `SOA`. Die untere Abbildung zeigt, die Erweiterung einer aus dem Begriff `SOA` bestehenden Anfrage auf Basis der `del.icio.us`-Funktionalität „related Tags“.

deten Tags sind im Allgemeinen Schlagwörter, die von den Benutzern als hilfreich für die Erweiterung von Anfragen bewertet wurden. Allerdings besteht bei der Verwendung von benutzergenerierten Inhalten das Risiko von Rechtschreibfehlern und Spam. Bei der durchgeführten Evaluation traten unpassende Empfehlungen beim Ermitteln von Suchbegriffen auf Basis von `del.icio.us` sehr selten auf.

Anfrageoptimierung auf Basis von Klassifikatoren

Gibt ein Benutzer zu einer Anfrage Bewertungen ab, so können diese Bewertungen zum Optimieren der Anfrage genutzt werden. Vergibt der Benutzer für einige Dokumente positive Bewertungen und für einige Dokumente negative Bewertungen, kann ein Klassifikator gelernt werden, mit dem „gute" von „schlechten" Dokumenten getrennt werden können. Durch Analyse der Merkmale, anhand derer ein gelernter Klassifikator entscheidet, zu welcher Klasse ein Dokument zuzuordnen ist, werden Begriffe zur Verfeinerung der Suchanfrage abgeleitet.

Das PIA-System setzt als Klassifikatoren den Algorithmus C4.5 [Qui93] und den ROCCHIO-Algorithmus [Joa97] ein. Zum Erstellen eines ROCCHIO-Klassifikators berechnet man zunächst für die bewerteten Dokumente den Termhäufigkeitsvektor. Dabei werden Wörter auf ihren Stamm reduziert und Stopp-Wörter ignoriert (unter Verwendung der Biblioteken LUCENE[12] und SNOWBALL[13]). Für alle in den bewerteten Dokumenten auftretenden Wortstämme („Terme") wird der Tf − Idf-Wert berechnet. Unter Verwendung des ROCCHIO-Algorithmus werden für die positiv bewerteten Dokumente (D_+) und für die negativ bewerteten Dokumente (D_-) die Prototyp-Vektoren (c_+, c_-) berechnet.

$$
\begin{aligned}
c_+ &= \frac{1}{|D_+|} \sum_{d \in D_+} \frac{\mathrm{Tf} - \mathrm{Idf}(d)}{|\mathrm{Tf} - \mathrm{Idf}(d)|} - \frac{1}{|D_-|} \sum_{d \in D_-} \frac{\mathrm{Tf} - \mathrm{Idf}(d)}{|\mathrm{Tf} - \mathrm{Idf}(d)|} \\
c_- &= \frac{1}{|D_-|} \sum_{d \in D_-} \frac{d}{|d|} - \frac{1}{|D_+|} \sum_{d \in D_+} \frac{d}{|d|}
\end{aligned}
$$

Die Terme (bzw. die zugehörigen Wörter), denen im Tf − Idf-Vektor c_+ ein großer Wert zugeordnet ist, werden dem Benutzer zum Erweitern der Anfrage vorgeschlagen, da diese Begriffe besonders gut zum Erkennen von positiv bewerteten Dokumenten geeignet sind. Terme, denen eine Gewichtung im c_--Vektor zugeordnet ist, können dem Benutzer als Terme vorgeschlagen werden,

[12] `http://lucene.apache.org/`
[13] `http://snowball.tartarus.org/`

die mit einem `NOT`-Operator in die Anfrage aufgenommen werden sollten, um Dokumente auszuschließen, die zu negativ bewerteten Filterergebnisse ähnlich sind.

Ein Beispiel für die Extraktion von Begriffen unter Verwendung des ROCCHIO-Algorithmus zeigt die Abbildung 8.6. Die unter Verwendung des ROCCHIO-Algorithmus ermittelten Suchbegriffe wurden im Allgemeinen von den Benutzern als gut bewertet und der Ansatz, auf Basis der Benutzerbewertungen Suchbegriffe abzuleiten, als gut nachvollziehbar eingestuft. Eine Voraussetzung für die Bestimmung guter Begriffe ist, dass hinreichend viele Dokumente bewertet werden, um eine aussagekräftige Tf – Idf-Statistik berechnen zu können. Darüber hinaus ist zu berücksichtigen, dass durch die Reduktion von Wörtern auf Wortstämme der Algorithmus als Vorschläge für Suchbegriffe Wortstämme bereitstellt. Da die Mehrheit der Benutzer von einem Anfrageassistenten das Vorschlagen von Wörtern erwartet, müssen die Wortstämme in passende Wörter transformiert werden. Wortstammbasierte Vorschläge wurden als verwirrend und schwer nachvollziehbar bewertet.

Anfrageoptimierung durch Dokumentenanalyse

Eine weitere Strategie zum Generieren von Vorschlägen zur Anfrageoptimierung besteht im Extrahieren von charakteristischen Wörtern und Phrasen aus Dokumenten, die der Benutzer positiv bewertet hat. Die Extraktion von Features (d. h. Wörter, Phrasen oder Entitäten, wie z. B. Autoren) aus Texten hilft dem Benutzer wichtige Wörter und Phrasen in als relevant eingestuften Dokumenten zu finden, die zur Beschreibung der Interessensgebiete des Benutzers geeignet sind.

Im PIA-System werden verschiedene Heuristiken zum Extrahieren eingesetzt. Dazu werden die Wörter des analysierten Textes bezüglich mehrerer Faktoren beurteilt:

- **Die** Tf – Idf**-Gewichtung des Wortes:** Wörter, die im analysierten Dokument häufig, in der gesamten Dokumentensammlung aber selten vorkommen, sind charakteristisch für das Dokument.

- **Die Häufigkeit eines Wortes:** Wörter, die innerhalb eines Dokumentes häufig auftreten, sind potentiell gut für die Charakterisierung des Dokumentes geeignet. Bei der Analyse der Worthäufigkeiten (Tf Analyse) müssen Stoppwörter erkannt und ausgeschlossen werden. Im Gegensatz zur Berechnung der Tf – Idf-Gewichtung wird keine repräsentative Dokumentensammlung benötigt.

Abbildung 8.6: Optimieren von Anfragen unter Verwendung des ROCCHIO-Algorithmus. Die obere Abbildung zeigt, dass der Benutzer sowohl positive als auch negative Dokumentenbewertungen vergeben hat. Die untere Abbildung zeigt, die auf Basis des ROCCHIO-Algorithmus abgeleiteten Begriffe.

- **Position des Wortes im Dokument:** Wörter, die in der Überschrift und der der Zusammenfassung des Dokumentes auftreten, sind im Allgemeinen wichtige Wörter zum Beschreiben des Dokumentes. Wörter am Anfang („Einleitung“) und am Ende („Zusammenfassung') eines Textes sind im Durchschnitt besser geeignet, als die Wörter im Mittelteil eines Dokumentes [KPK01].

- **Wortart:** Substantive und Namen beschreiben oft charakteristische Konzepte eines Dokumentes.

- **Zusammenhängende Begriffe:** Wortpaare, die mehrfach in einem Text auftreten, die kein Stoppwort beinhalten (z. B. durch ein Adjektiv näher bestimmtes Substantiv) beschreiben oft wichtige Konzepte eines Textes.

Bei der Extraktion von Begriffen aus Texten werden anhand der genannten Kriterien Bewertungen für die Wörter und Phrasen eines Textes berechnet. Die verwendete Gewichtung der Faktoren wird im PIA-System in Abhängigkeit von der Dokumentengröße gewählt: Bei sehr kurzen Dokumenten werden die Faktoren Wortart und Position besonders berücksichtigt. Bei umfangreichen Dokumenten erhält die auf Basis des Tf − Idf Wertes ermittelte Bewertung eine höhere Gewichtung, da auf Basis von umfangreichen Dokumenten eine aussagekräftige Tf − Idf-Statistik ermittelt werden kann.

Die Evaluation der heuristischen Extraktion von Begriffen zeigte, dass die Benutzer die Empfehlungen als hilfreich und sinnvoll bewerten. Als positiv wurde dabei angemerkt, dass oft Wortgruppen vorgeschlagen werden, die eine gute Präzisierung von Suchanfragen ermöglichen.

Ein Beispiel für die Extraktion von Phrasen zum Optimieren von Anfragen anhand von positiv bewerteten Dokumenten zeigt die Abbildung 8.7.

Anfrageoptimierung unter Verwendung von Clustern

Die Verwendung von Cluster-Algorithmen ermöglicht es, untereinander ähnliche Dokumente in Gruppen zusammenzufassen. Das PIA-System bietet die Funktionalität, die zu einer Anfrage ermittelte Menge von Filterergebnissen in Form eines Baumes darzustellen, wobei ähnliche Dokumente in einem Knoten zusammengefasst werden. Ist anhand des Benutzerverhaltens (Bewertungen, Auswertung von impliziten Feedback) zu erkennen, dass sich der Benutzer besonders für die einem Knoten zugeordneten Dokumente interessiert, so können die für den Knoten charakteristischen Merkmale (z. B. Wörter und Phrasen) dem Benutzer zur Anfrageoptimierung vorgeschlagen werden. Das Gruppieren von Dokumenten erfolgt im PIA-System auf Basis von

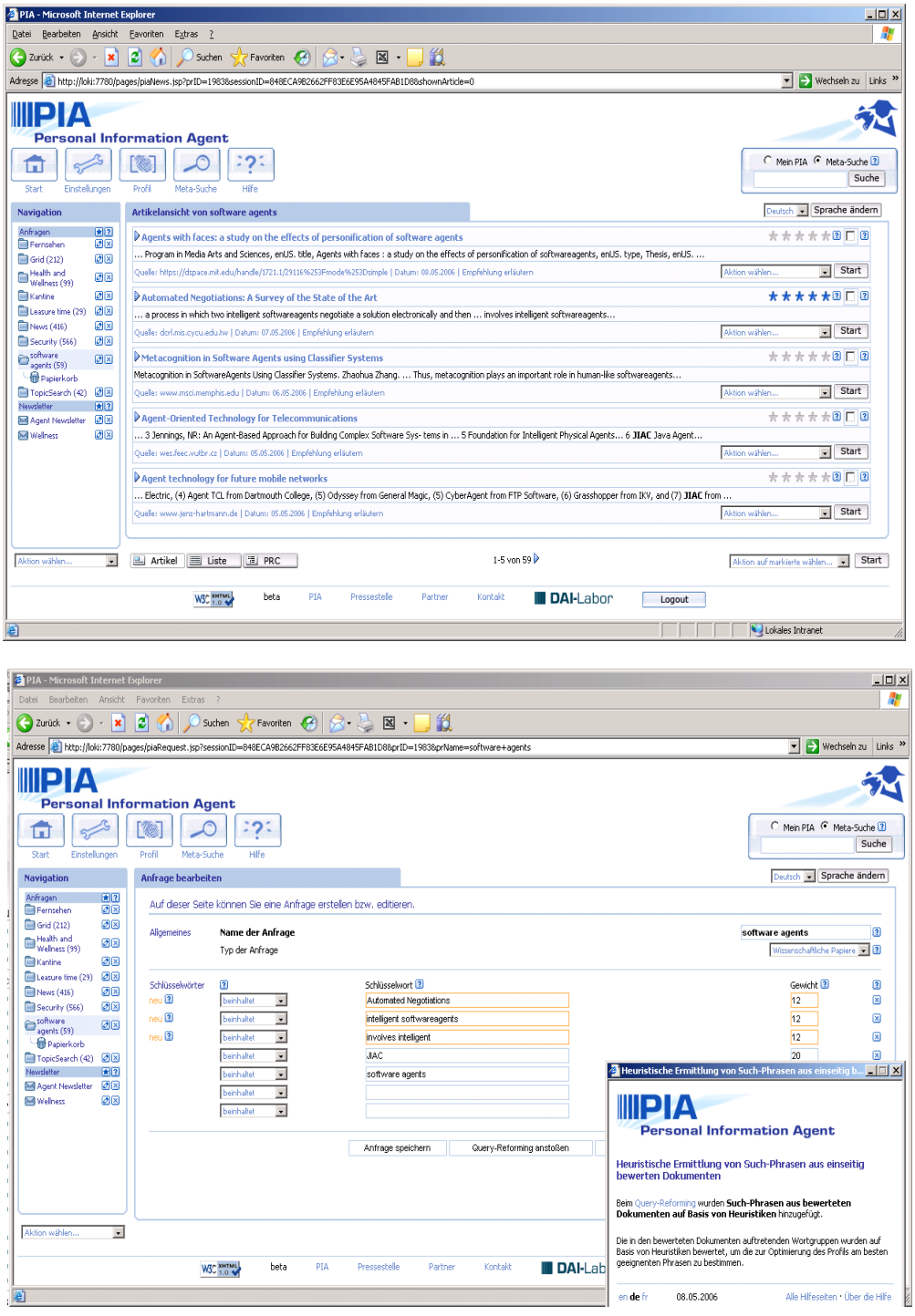


Abbildung 8.7: Die obere Abbildung zeigt eine Ergebnisliste, für die der Benutzer die Relevanz von Dokumenten bewertet hat. Die untere Abbildung zeigt die auf Basis der Bewertungen ermittelten Phrasen zur Optimierung der Suchanfrage.

Heuristiken und unter Verwendung des DisCover-Algorithmus [KLR+04]. Ein Beispiel für die Extraktion von Begriffen auf Basis einer hierarchisch geclusterten Dokumentenmenge zeigt die Abbildung 8.8.

Kollaborative Anfrageoptimierung

Die Verfahren zur kollaborativen Anfrageoptimierung analysieren die Ähnlichkeit zwischen Benutzern und generieren Anfrageoptimierungsvorschläge, indem sie häufig von ähnlichen Benutzern verwendete Begriffe ermitteln.

Das PIA-System setzt zur kollaborativen Anfrageoptimierung das folgende Vorgehen ein: Zunächst werden anhand der in der zu optimierenden Anfrage verwendeten Wörter ähnliche Anfragen bestimmt. Als Ähnlichkeitsmaß wird hierbei die Anzahl der übereinstimmenden Wortstämme in den Anfragen eingesetzt. Über die (in den ermittelten Anfragen) auftretenden Wortstämme wird eine Häufigkeitsstatistik erstellt, wobei die Wortstämme jeweils mit der berechneten Anfrageähnlichkeit gewichtet werden.

Um das Problem von geringen Übereinstimmungen zwischen Anfragen zu berücksichtigen („sparsity“), wurden alternativ auch graphbasierte Ähnlichkeitsmaße zwischen Anfragen getestet. Durch die Berücksichtigung transitiver Beziehungen zwischen Begriffen kann eine größere Anzahl von Begriffen bei der Ähnlichkeitsberechnung berücksichtigt werden, was zu zuverlässigeren Ergebnissen führt. Der Nachteil besteht vor allem in einer schlechten Nachvollziehbarkeit und damit geringer Akzeptanz durch den Benutzer. Zudem sind graphbasierte Ähnlichkeitsmaße nur mit erheblich höherem Aufwand berechenbar. Die Abbildung 8.9 veranschaulicht die kollaborative Generierung von Vorschlägen anhand des Übernehmen von Begriffen aus der Anfrage eines Benutzers mit ähnlichen Begriffen.

Die Güte der bei diesem Verfahren generierten Vorschläge hängt (wie bei den meisten kollaborativen Algorithmen) vor allem von der Anzahl der im System hinterlegten Anfragen sowie der Qualität der hinterlegten Schlüsselwörter ab. Gute Vorschläge können nur dann ermittelt werden, wenn bereits Benutzer mit ähnlichen Interessen existieren („cold start problem“).

Das Verfahren zur kollaborativen Ermittlung von Begriffen weist die folgenden Vorteile auf:

- Die ermittelten Vorschläge basieren auf von Nutzern definierten Begriffen, die oft eine gute Hilfestellungen zum Finden relevanter Dokumente geben. Die Nutzer profitieren gegenseitig von ihrem Wissen.

- Kollaborative Verfahren verfolgen einen zu inhaltsbasierten Methoden orthogonalen Ansatz. Dies ermöglicht es, Begriffe zu ermitteln, die neue Aspekte der Suchanfrage berücksichtigen.

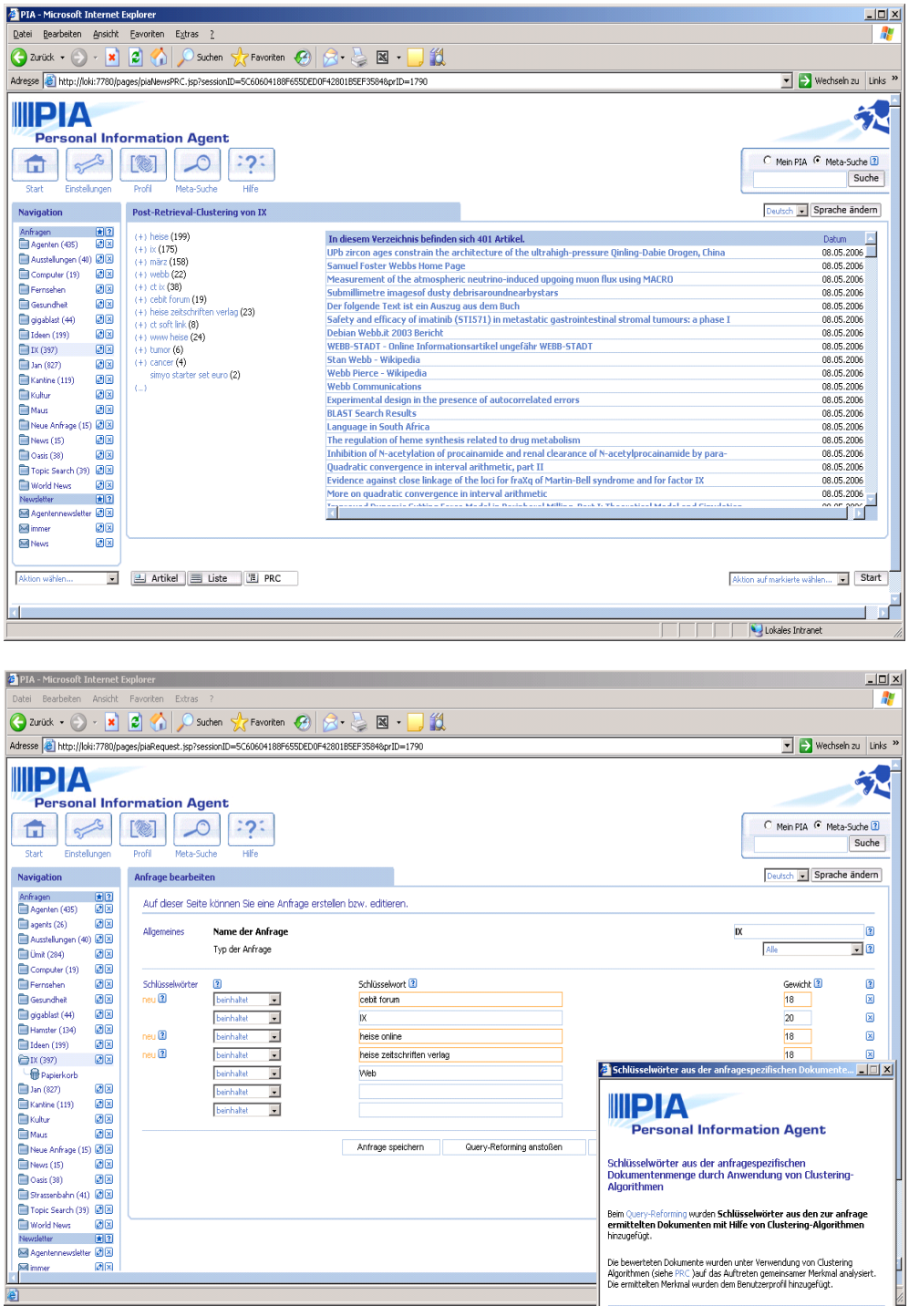


Abbildung 8.8: Die obere Abbildung zeigt die auf Basis einer Ergebnismenge abgeleiteten Cluster. Hat der Benutzer Dokumente positiv bewertet, so werden die Cluster ermittelt, denen die bewerteten Dokumente zugeordnet sind. Die zur Beschreibung dieser Cluster ermittelten Begriffe („Cluster-Labels“) werden dem Benutzer zum Optimieren der Anfragen vorgeschlagen.

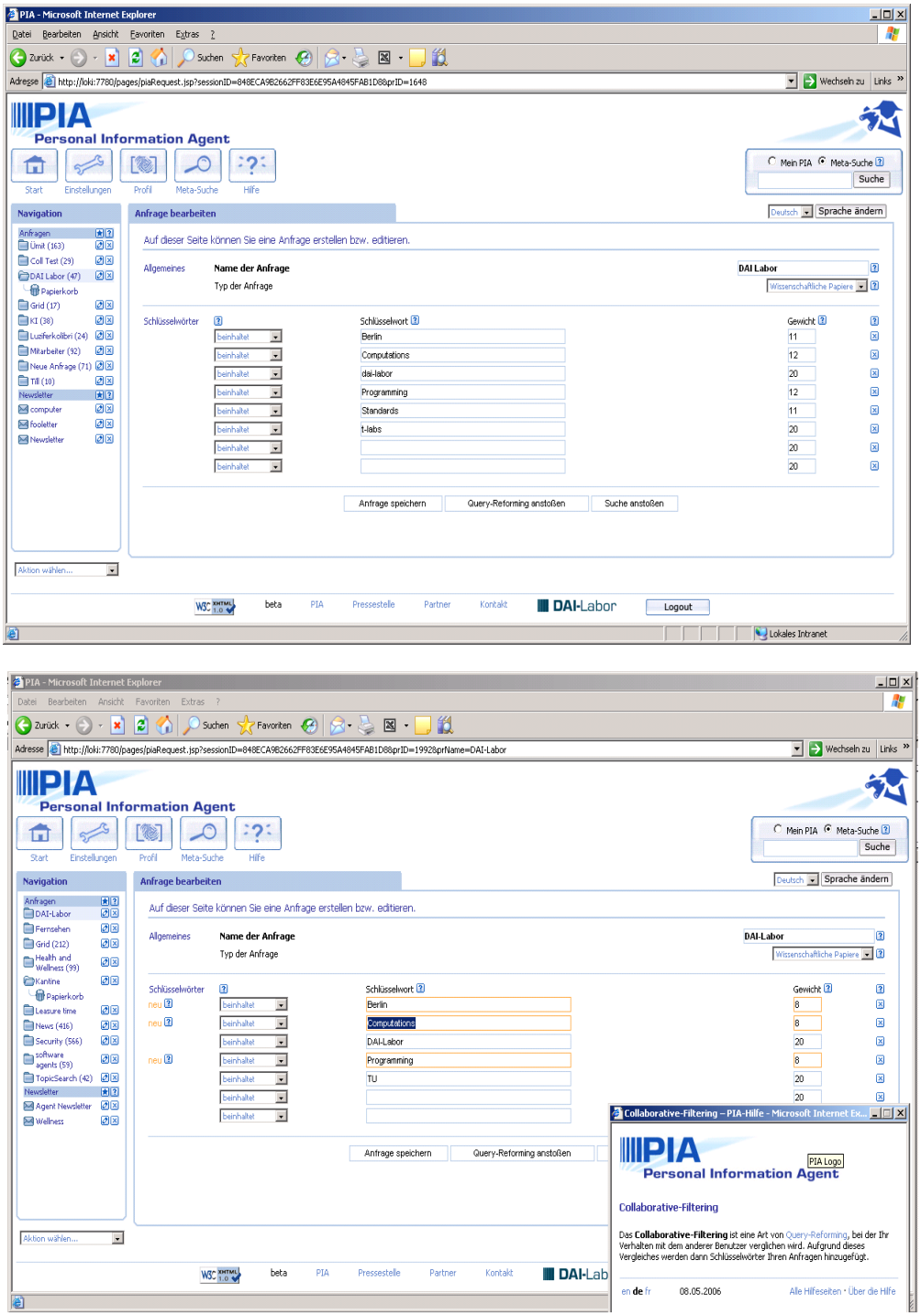


Abbildung 8.9: Die Abbildung veranschaulicht das Erweitern einer Anfrage um Begriffe, die zum Profil eines ähnlichen Benutzers gehören.

- Das Verfahren kann zum Ermitteln von relevanten Wörter, Phrasen und Entitäten eingesetzt werden.

Die Nachteile des Verfahrens, die insbesondere bei geringer Anzahl von Benutzern auftreten, sind:

- Weisen die zur Suche verwendeten Begriffe Rechtschreibfehler auf, so besteht die Gefahr, dass Benutzern fehlerhaft geschriebene Begriffe vorgeschlagen werden.
- Böswillige Benutzer können dieses Verfahren ausnutzen, um anderen Benutzern manipulierte Suchbegriffe zu empfehlen.

Bei der Verwendung des Verfahrens im PIA-System war festzustellen, dass die Güte der Vorschläge von Anfrage zu Anfrage sehr unterschiedlich war. Dies ist damit zu erklären, dass bei einer relativ geringen Anzahl von Benutzern für einige Themenbereiche viele Anfragen existieren, die für die Anfrageoptimierung herangezogen werden; für andere Bereiche aber nicht hinreichend viele Anfragen zur Verfügung stehen. Die durch stark schwankende Qualität der Vorschläge wird von vielen Benutzern als negativ bewertet.

8.4.2 Auswahl von Anfrageoptimierungsstrategien

Um zu bestimmen, welche Methode zum Generieren von Vorschlägen für eine vorliegende Anfrage zum Einsatz kommen sollte, wird im PIA-System eine regelbasierte Auswahlkomponente eingesetzt. Diese analysiert zunächst, für welche der Anfrageoptimierungsstrategien die jeweils notwendigen Daten zur Verfügung stehen. Dabei werden beispielsweise die Menge der zu einer Anfrage bewerteten Dokumente, die Anzahl der bisher in der Anfrage vorhandenen Terme sowie die im Benutzerprofil gespeicherten Informationen ausgewertet. Anhand von Regeln wird berechnet, wie gut die verfügbaren Daten zu den Anforderungen der jeweiligen Algorithmen passen. Es wird berücksichtigt, von welchen Algorithmen der Benutzer die Vorschläge bereits abgelehnt und von welchen Strategien der Benutzer die Vorschläge zur Erweiterung einer Anfrage angenommen hat.

Bei der Evaluation des Algorithmus zum Auswählen von Profiloptimierungsvorschlägen zeigt sich, dass viele Benutzer es bevorzugen, den Algorithmus manuell auszuwählen. Da die Mehrheit der Benutzer die Funktion zum Erklären der Vorschläge nicht nutzt, kann die automatische Wahl eines Vorschlagsalgorithmus dazu führen, dass die ermittelten Ergebnisse nicht den Erwartungen des Benutzers entsprechen (z.B. wenn ein kollaboratives Verfahren ausgewählt wird, der Benutzer aber auf Basis eines inhaltsbasierten

Verfahrens ermittelte Vorschläge erwartet). Darüber hinaus ergab die Evaluation, dass einige Benutzer Vorbehalte gegen einige Vorschlagsstrategien haben (z. B. Ablehnung von kollaborativen Verfahren). Es sollte daher im System die Möglichkeit gegeben werden, den jeweils zum Generieren von Optimierungsvorschlägen eingesetzten Algorithmus manuell zu wählen.

8.4.3 Erstellung von Anfragen aus Dokumenten

Als Ergänzung zum Optimieren von Anfragen bietet das entwickelte Framework die Möglichkeit, Anfragen auf Basis von Dokumenten zu erstellen. Es werden dabei aus einem gegebenen Dokument charakteristische Wörter und Terme extrahiert. Im Gegensatz zur Optimierung bereits bestehender Anfragen, muss beim Erstellen von neuen Anfragen, eine geeignete Anzahl von Termen gewählt sowie eine angemessene Verknüpfung der Begriffe definiert werden.

Die Funktionalität „Erstellen von Anfragen aus Dokumenten“ weist Ähnlichkeit mit der Funktion „Ähnliche Dokumente Finden“ auf, die von einigen Suchmaschinen (z. B. GOOGLE) angeboten wird. Das Erstellen von Suchanfragen auf Basis von Dokumenten (ohne das sofortige Starten einer Suche) hat den Vorteil, dass der Benutzer die erstellte Anfrage prüfen und ggf. anpassen kann. Die Funktion „Finden ähnlicher Dokumente“ wird damit auf die dem Benutzer bekannte Form der Suche zurückgeführt. Ein Beispiel für die Erstellung von Anfragen auf Basis eines Dokumentes im System PIA zeigt die Abbildung 8.10.

Zum Extrahieren von Suchbegriffen und Phrasen werden Verfahren zur inhaltsbasierten Dokumentenanalyse eingesetzt, die bereits im Zusammenhang mit der Optimierung von Anfragen erläutert wurden. Um eine präzise Anfrage zu erstellen, werden jeweils 5 Wörter oder Phrasen ausgewählt, die durch einen `OR`-Operator verknüpft werden. Die Evaluation im PIA-System hat gezeigt, dass die auf diese Weise auf Basis von Dokumenten erstellten Anfragen, eine gute Hilfestellung für eine neue Suchanfrage geben.

Von den Benutzern wurde die Funktion insbesondere dann als nützlich bewertet, wenn Dokumente zu einem Themengebiet ermittelt werden sollen, mit dem der Benutzer noch nicht vertraut ist. Die auf der Basis von Texten erstellten Suchanfragen helfen dem Benutzer, die für den neuen Themenbereich relevanten Begriffe zu finden. Die vorgeschlagenen Begriffe kann der Benutzer entsprechend seiner individuellen Präferenzen anpassen oder (unter Verwendung der Komponente zur Anfrageoptimierung) erweitern.

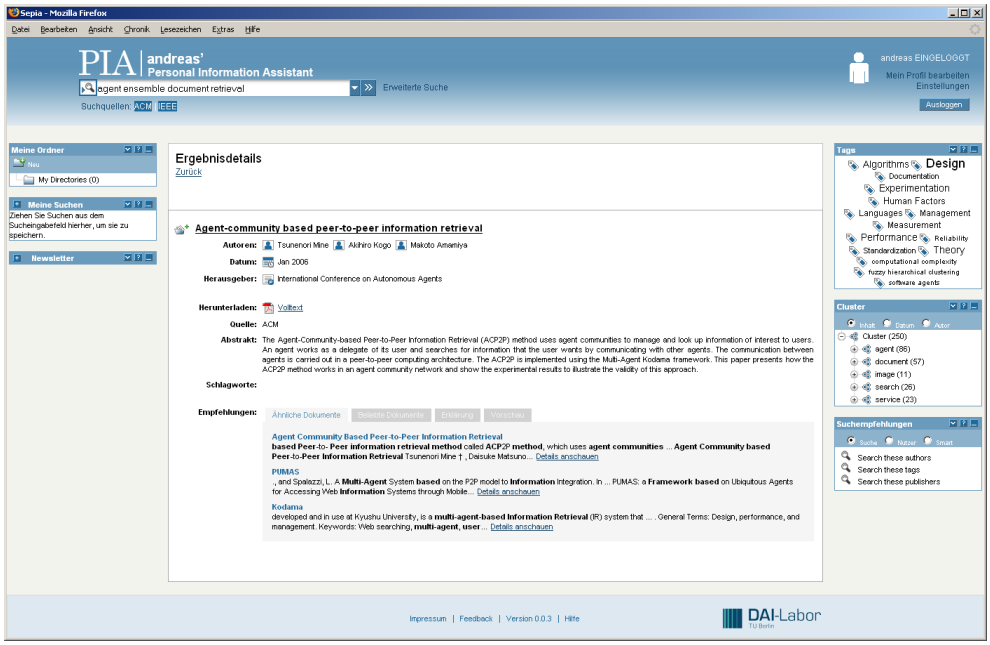


Abbildung 8.10: Die Abbildung visualisiert die Funktion „Ähnliche Dokumente“ am Beispiel des Informationssystems PIA.

8.4.4 Zusammenfassung und Bewertung

Anhand des PIA+COMM-Systems wurden die entwickelten Funktionen zum Erweitern von Anfragen erläutert. Die Evaluation auf der Basis von Benutzerbefragungen zeigte, dass eine Unterstützung bei der Erstellung und Optimierung von Anfragen als wichtig und hilfreich eingeschätzt wird. Besonders positiv wurde beurteilt, dass die vorgeschlagenen Begriffe nützliche Anregungen geben und dem Benutzer helfen, die verschiedenen Aspekte einer Anfrage zu erkennen. Für die Akzeptanz von Optimierungsvorschlägen ist es wichtig, dass der Benutzer versteht, warum ihm ein bestimmter Begriff vorgeschlagen wird. Das System sollte daher zu Anfrageoptimierungsvorschlägen Erläuterungen bereitstellen, die erklären, warum und in welcher Hinsicht ein Vorschlag zur Optimierung einer Anfrage geeignet ist. Dies kann dadurch erfolgen, dass dem Benutzer erläutert wird, welche Daten für die Generierung von Vorschlägen herangezogen und welcher Grundidee das Vorschlagsverfahren folgt.

Da einige Benutzer bestimmte Typen von Vorschlagsalgorithmen (z. B. kollaborative Algorithmen) ablehnen, sollte der Benutzer die Möglichkeit haben, die Menge der bei der Anfrageoptimierung berücksichtigen Algorithmen explizit festzulegen. Werden von einem Informationssystem eine große An-

zahl von Profiloptimierungsalgorithmen eingesetzt, wird dies von Benutzern teilweise als verwirrend empfunden, da dadurch bei der Profiloptimierung für den Benutzer unerwartete Vorschläge generiert werden können. Es wurde stattdessen gewünscht, den Algorithmus selbst auswählen zu können, um sicherzustellen, dass die Vorschläge jeweils von einem Algorithmus stammen, dem der Benutzer vertraut und den er verstanden hat.

Die Verfahren zum Ermitteln von Begriffen zur Anfrageoptimierung können anstatt für die interaktive Anfrageoptimierung auch im Hintergrund (d. h. für den Benutzer transparent) für das Erweitern von Anfragen eingesetzt werden. Das Erweitern von Anfragen kann helfen, sowohl die Filterpräzision als auch den Recall zu verbessern. Bei der Auswahl der dabei eingesetzten Algorithmen müssen das jeweilige Suchszenario und die Kenntnisse des Benutzers berücksichtigt werden. Der Nachteil der für den Benutzer transparenten Anfrageerweiterung besteht darin, dass damit die Nachvollziehbarkeit der zu einer Suchanfrage ermittelten Dokumente sinkt.

8.5 Filter- und Bewertungsstrategien

Datenquellen

Für die implementierten Informationssysteme wurden Anbindungen für eine große Anzahl von Datenquellen realisiert. Dabei wurden sowohl Datenbanken (z. B. wissenschaftliche Dokumente von IEEE[14] und CORDIS[15]), Social Tagging-Systeme (z. B. `del.icio.us`), Lexika (z. B. WIKIPEDIA) und Portale (z. B. `n-tv`[16]), Mailinglisten und Suchmaschinen (z. B. YAHOO[17]) berücksichtigt. Da einige Informationsquellen keine leistungsfähige Suchfunktion anbieten, wurde für einige der Quellen ein Suchindex erstellt und für das Bearbeiten von Anfragen genutzt. Für die Auswahl von Datenquellen werden in den implementierten Informationssystemen auf das jeweilige Szenario abgestimmte Regeln verwendet.

Bewertungsstrategien

Um die von den Informationsquellen ermittelten Dokumente umfassend zu bewerten, werden vielfältige Bewertungsstrategien eingesetzt. Da sich die auf Basis der entwickelten Architektur implementierten Informationssyste-

[14] `http://ieeexplore.ieee.org/`
[15] `http://www.cordis.lu`
[16] `http://www.n-tv.de`
[17] `http://yahoo.de`

me stark in Bezug auf das Suchszenario und die Benutzerzielgruppe unterscheiden, kommen auf das jeweilige System optimierte Bewertungsstrategien zum Einsatz. Faktoren, die bei der Auswahl von Filterstrategien berücksichtigt werden müssen, sind der Umfang von verfügbaren Benutzerprofilen (d. h. stehen hinreichend Daten zum Benutzer zur Verfügung, um personalisierte Bewertungen berechnen zu können), die für die Bearbeitung einer Suchanfrage zur Verfügung stehende Zeit (d. h. wie komplex können durchgeführte Berechnungen sein) sowie die Art der zu bewerteten Dokumente (z. B. Nachrichten oder wissenschaftliche Artikel). So kommen im System PIA, das auf die schnelle Bewertung von Dokumenten optimiert ist (die vorrangig von Suchmaschinen und Portalen stammen), Algorithmen zum Einsatz, die auf eine rasche Bearbeitung und die Erkennung von Spam optimiert sind. Bei der Zusammenstellung einer personalisierten Zeitung (im Rahmen des Projektes PZD) werden dagegen komplexere, auf die personalisierte Bewertung von Dokumenten optimierte Bewertungsstrategien eingesetzt.

Eine Auswahl der implementierten Bewertungsstrategien (Agenten) zeigt die Tabelle 8.1.

8.6 Zusammenstellen und Darstellen von Ergebnissen

Im Rahmen der verschiedenen Systeme kommen unterschiedliche Verfahren zum Zusammenstellen und zur Präsentation der Filterergebnismenge zum Einsatz.

Kombination von Bewertungsstrategien

Zum Zusammenführen von Bewertungen wird in den implementierten Systemen vorrangig die Strategie CombSUM (mit Berücksichtigung von Gewichten) verwendet, wobei die Gewichtungen für die Bewertungsagenten an das jeweilige Szenario angepasst werden. Darüber hinaus werden für das Zusammenstellen vorläufiger Filterergebnisse expertendefinierte Heuristiken eingesetzt, die auf das schnelle Erstellen von Ergebnismengen bei unvollständigen Bewertungen optimiert sind. Für das Zusammenstellen von Newslettern wurden Strategien der globalen Optimierung (Constraint-Server) gestestet.

Die Evaluation zeigt, dass die Bewertungskombinationsstrategie CombSUM das effiziente und für den Benutzer gut nachvollziehbare Zusammenführen von Bewertungen ermöglicht. Bei Informationssystemen, die Suchmaschinen als Datenquellen berücksichtigen (wie z. B. das PIA-System), erhalten die Agenten, die Dokumente bezüglich der Kriterien

Tabelle 8.1: Eine Auswahl der implementierten Bewertungsagenten

Name	Beschreibung
Erkennen von Internet-Shops	Um zu bestimmen, ob Dokumente von einem Internet-Shop stammen, wird ein Klassifikator eingesetzt. Dieser verwendet eine Liste von URLs bekannter Internet-Shops. Auf der Basis dieser Liste wird ein Naive-Bayes-Klassifikator erstellt, der anhand von in einem Dokument auftretenden Termen, Internet-Shops erkennt.
Erkennen von Spam	Zum Erkennen von Spam wird ein mehrstufiges Verfahren eingesetzt. Zunächst wird die Adresse geprüft. Anschließend wird für Spam-Dokumente typische Merkmale (z. B. versteckter Text) ermittelt und anhand eines Punktesystems, die Wahrscheinlichkeit berechnet, dass es sich bei einem Dokument um Spam handelt.
Unscharfe Übereinstimmung mit Suchbegriffen	Um auch dann Übereinstimmungen mit Suchbegriffen zu erkennen, wenn der Benutzer eine ungewöhnliche Schreibweise verwendet oder Rechtschreibfehler auftreten, kommen Verfahren des unscharfen Matchings zum Einsatz. Im PIA-System wird zum Suchen von Anfragebegriffen im Dokumententitel und in der Zusammenfassung die Levenshtein-Distanz berechnet.
Kostenpflichtiges Dokument	Von Suchmaschinen ermittelte Dokumente sind teilweise kostenpflichtig. Möchte der Benutzer aber ausschließlich kostenfrei erhältliche Dokumente erhalten, werden nur nach Bezahlung erreichbare Inhalte als sehr störend empfunden. Das entwickelte Filterframework bietet daher einen Agenten zum Erkennen von kostenpflichtigen Inhalten. Man setzt einen Klassifikator ein, der den Inhalt von HTML-basierten Dokumenten auf charakteristische Merkmale (z. B. Login zu Bezahldiensten) prüft.
Dokumenten-Domäne	Um sicherzustellen, dass ein Dokument zum Interessensbereich des Benutzers passt, wird die Domäne ermittelt, zu der das Dokument gehört. Es wird dazu ein auf C4.5 basierender Klassifikator eingesetzt, der anhand der im Dokument auftretenden Terme die Domäne ermittelt. Die zur Klassifikation verwendeten Terme werden dem Benutzer als Erklärung angeboten [Str06].
Ähnlichkeit mit vom Benutzer bewerteten Dokumenten	Es wird die Ähnlichkeit eines Dokumentes zu den n letzten vom Benutzer bewerteten Dokumenten ermittelt. Man erstellt einen Suchindex aus den letzten n bewerteten Dokumenten und verwendet das zu bewertende Dokument als Suchanfrage. Man berechnet den (mit der ermittelten Dokumentenähnlichkeit gewichteten) Durchschnitt über die vom Benutzer vergebenen Bewertungen.
Bewertung des Dateiformats	Auf der Basis eines Regelsystems wird bewertet, ob das Dateiformat eines Dokumentes zum aktuellen Szenario passt.

„Spam“ sowie Zugehörigkeit zum gewünschten Themenbereich bewerten, hohe Gewichtungen, da Benutzer Werbung und nicht zum geforderten Themengebiet passende Dokumente als besonders störend beurteilen. Die jeweils initial durch einen Experten gewählten Gewichtungen können anhand von Benutzerfeedback adaptiert werden. Als zukünftige Erweiterung ist geplant, die Gewichtungen dynamisch an die Benutzerbewertungen anzupassen und dem Benutzer zu erlauben, Gewichte selbst anzupassen. Dies ermöglicht es, den Einfluss von kollaborativen, inhaltsbasierten und personalisierten Bewertungskriterien auf die individuellen Wünsche abzustimmen.

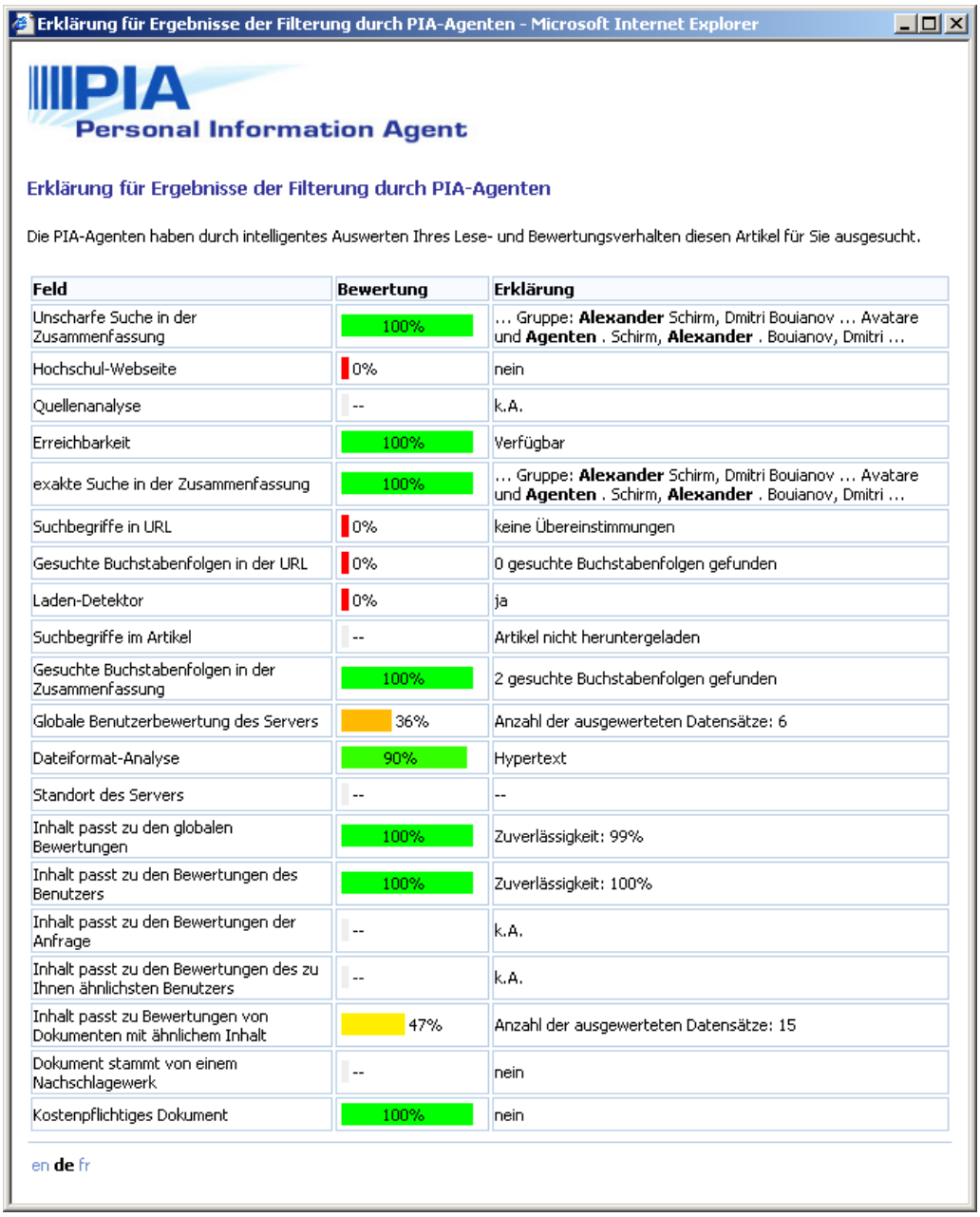
Präsentation

Zur Präsentation der Filterergebnisse verwenden die entwickelten Systeme größtenteils eine listenbasierte Darstellung, die jeweils nach unterschiedlichen Kriterien sortiert werden kann. Im System PIA+COMM können die wissenschaftlichen Dokumente auf Basis inhaltlicher Ähnlichkeiten hierarchisch gruppiert werden.

Um die Nachvollziehbarkeit der vom Informationssystem ermittelten Relevanzbewertungen sicherzustellen, besteht in den Systemen PIA und PIA+COMM die Möglichkeit, die von den berücksichtigten Bewertungsagenten ermittelten Relevanzbeurteilungen einzusehen. Abbildung 8.11 zeigt die Bewertungen der beim Ermitteln der Relevanz eines Dokumentes berücksichtigten Agenten. Zu jeder Bewertung wird zudem eine Erklärung angeboten, die das Zustandekommen der Bewertung erläutert. Das Einsehen der Bewertungen der Einzelagenten ermöglicht es zudem, Fehler frühzeitig zu erkennen und ggf. Bewertungsagenten zu optimieren. So können beispielsweise Bewertungsagenten auf Basis von Benutzerbewertungen nachtrainiert werden, wobei die von einem Agenten fehlerhaft bewerteten Dokumente (nach dem Boosting-Ansatz) eine besondere Gewichtung in der Trainingsmenge erhalten.

Diskussion und Bewertung

Die Evaluation der in den implementierten Informationssystemen Bewertungskombinationsverfahren zeigt, dass durch die auf das Suchszenario abgestimmte Kombination mehrerer Bewertungskriterien eine hohe Qualität der Filterergebnisse erreicht wird. Von den Benutzern wurde gelobt, dass erläutert wird, welche Faktoren für die Relevanzbewertung herangezogen werden und inwiefern ein Dokument die verschiedenen Kriterien erfüllt. Dies ist insbesondere dann von Bedeutung, wenn ein Dokument auf den ersten Blick nur begrenzt relevant erscheint. Allerdings gibt es auch Benutzer, die sich nur



Erklärung für Ergebnisse der Filterung durch PIA-Agenten - Microsoft Internet Explorer

PIA
Personal Information Agent

Erklärung für Ergebnisse der Filterung durch PIA-Agenten

Die PIA-Agenten haben durch intelligentes Auswerten Ihres Lese- und Bewertungsverhalten diesen Artikel für Sie ausgesucht.

Feld	Bewertung	Erklärung
Unscharfe Suche in der Zusammenfassung	100%	... Gruppe: **Alexander** Schirm, Dmitri Bouianov ... Avatare und **Agenten** . Schirm, **Alexander** . Bouianov, Dmitri ...
Hochschul-Webseite	0%	nein
Quellenanalyse	--	k.A.
Erreichbarkeit	100%	Verfügbar
exakte Suche in der Zusammenfassung	100%	... Gruppe: **Alexander** Schirm, Dmitri Bouianov ... Avatare und **Agenten** . Schirm, **Alexander** . Bouianov, Dmitri ...
Suchbegriffe in URL	0%	keine Übereinstimmungen
Gesuchte Buchstabenfolgen in der URL	0%	0 gesuchte Buchstabenfolgen gefunden
Laden-Detektor	0%	ja
Suchbegriffe im Artikel	--	Artikel nicht heruntergeladen
Gesuchte Buchstabenfolgen in der Zusammenfassung	100%	2 gesuchte Buchstabenfolgen gefunden
Globale Benutzerbewertung des Servers	36%	Anzahl der ausgewerteten Datensätze: 6
Dateiformat-Analyse	90%	Hypertext
Standort des Servers	--	--
Inhalt passt zu den globalen Bewertungen	100%	Zuverlässigkeit: 99%
Inhalt passt zu den Bewertungen des Benutzers	100%	Zuverlässigkeit: 100%
Inhalt passt zu den Bewertungen der Anfrage	--	k.A.
Inhalt passt zu den Bewertungen des zu Ihnen ähnlichsten Benutzers	--	k.A.
Inhalt passt zu Bewertungen von Dokumenten mit ähnlichem Inhalt	47%	Anzahl der ausgewerteten Datensätze: 15
Dokument stammt von einem Nachschlagewerk	--	nein
Kostenpflichtiges Dokument	100%	nein

en **de** fr

Abbildung 8.11: Die Abbildung zeigt die Einzelbewertungen der beim Berechnen der Relevanz eines Dokumentes berücksichtigten Agenten. Zu den Bewertungen wird jeweils eine kurze Erklärung bereitgestellt, die das Zustandekommen der Bewertung erläutert.

für die Dokumente interessieren und zusätzliche Erklärungen und detaillierte Bewertungen als störend bewerten. Die Benutzeroberfläche sollte hierbei so gestaltet werden können, dass die verschiedenen Wünsche und Erwartungen der Benutzer erfüllt werden.

Für die Administration von verteilten Informationssystemen hat sich die Verwendung einer Multi-Agentenplattform als günstig erwiesen. Insbesondere die durch die Plattform bereitgestellten Funktionen zur Realisierung eines verteilten Systems, dem Hinzufügen und Austauschen von Agenten zur Laufzeit, sowie die zentrale Überwachung und Protokollierung des auf mehrere Computer verteilten Informationssystem beschleunigt die Entwicklung und vereinfacht die Administration. Für den Benutzer bleibt die zur Realisierung des Informationssystems eingesetzte Multi-Agentenplattform transparent. Die Agenten-Metapher kann aber helfen, beim Benutzer ein besseres Verständnis für die komplexen Abläufe in einem Informationssystem zu schaffen.

8.7 Zusammenfassung

Die implementierten Informationssysteme zeigen, dass die entwickelte Architektur erfolgreich für die Realisierung von Informationssystemen eingesetzt werden kann. Es lassen sich damit sowohl für die themenbereichsübergreifende Informationssuche als auch auf ausgewählte Domänen (bzw. Szenarien) zugeschnittene Systeme realisieren. Es müssen jeweils die Datenquellen, Bewertungsstrategien und die Algorithmen zur Kombination von Bewertungen geeignet gewählt werden.

Flexibilität, Erweiterbarkeit Die Verwendung einer serviceorientierten Architektur ermöglicht es, die entwickelten Komponenten flexibel zu kombinieren und an das jeweilige Szenario anzupassen. Neue Agenten können zur Laufzeit des Systems hinzugefügt und bestehende Agenten ersetzt bzw. optimiert werden.

Die Kombination mehrerer, auf jeweils eine Aufgabe spezialisierter Agenten, anstatt von wenigen komplexen Filteragenten, vereinfacht die Wiederverwendung bestehender Agenten in unterschiedlichen Systemen und erleichtert damit das Erstellen von Informationssystemen für einen neuen Anwendungsbereich. Die Architektur stellt sicher, dass in ein Informationssystem neue Datenquellen und Bewertungsstrategien aufgenommen werden können. So ist es zum Beispiel möglich, die für ein themenbereichsspezifisches Informationssystem (wie z. B. PZD) entwickelten Bewertungsagenten auch in weiteren Informationssystemen (z. B. PIA) einzusetzen. Neue Strategien (wie

z. B. semantische Bewertungsalgorithmen) können effizient in ein Informationssystem integriert und mit bewährten Verfahren verknüpft werden, was Zukunftssicherheit schafft.

Das Bewerten von Dokumenten durch das Kombinieren der Ergebnisse mehrerer Filteragenten stellt sicher, dass die vom Informationssystem vorgeschlagenen Dokumente bezüglich mehrerer Kriterien zu den Interessen des Benutzers passen. Auf die individuellen Präferenzen des Benutzers trainierte Bewertungsagenten können berücksichtigt werden, was es ermöglicht, die Filterergebnismenge bezüglich der Präferenzen des Benutzers zu personalisieren. Das Bereitstellen von detaillierten Erläuterungen zu den ermittelten Bewertungen hilft dem Benutzer, die Relevanz von Dokumenten zu beurteilen und ggf. die verwendete Anfrage anzupassen oder die Gewichtungen für Bewertungsagenten zu adaptieren.

Unterstützen des Benutzers Die im Informationssystem PIA angebotenen Funktionen zum Erstellen und Optimieren von Anfragen wurden von den Benutzern als hilfreich und nützlich bewertet. Die Vorschläge zum Optimieren von Anfragen helfen dem Benutzer, die unterschiedlichen Aspekte einer Suchanfrage zu verstehen. Zudem motiviert diese Komponente die Benutzer, Dokumente explizit zu bewerten, da die Bewertungen zum Ermitteln personalisierter Vorschläge genutzt werden können. Im Vergleich mit alternativen Portalen (wie z. B. MetaGer-Web-Assoziator) wurde positiv bewertet, dass die Vorschläge auf die Bewertungen und Anfragen des Benutzers abgestimmt sind.

Präsentation der Filterergebnisse Die in den implementierten Informationssystemen verwendete Bewertungsstrategie CombSUM hat sich als leistungsfähige Strategie erwiesen. Die Kombination von Bewertungen unter Verwendung von CombSUM zeichnet sich durch eine hohe Qualität der Filterergebnisse bei geringerem Ressourcenbedarf und guter Nachvollvollziehbarkeit aus. Die im System PIA+COMM angebotene baumartigen Darstellung von Filterergebnissen ist für viele Benutzer neu; die Darstellung wird aber als sinnvoll bewertet, um die in der Dokumentenmenge berücksichtigten Themengebiete zu erkennen und um besonders interessante Dokumente schneller zu finden.

Administration Die während des Betriebs der implementierten Informationssysteme gesammelten Erfahrungen zeigen, dass die Verwendung einer Multi-Agentenplattform (wie JIAC IV) die Administration des Systems erleichtert. Es können die von der Agentenplattform bereitgestellten Dienste

zum Überwachen und Protokollieren der (auf mehrere Computer verteilten) Agenten genutzt werden. Zudem kann die durch das Agentenframework bereitgestellte Funktionalität zum Hinzufügen und Austauschen von Agenten genutzt werden. Die Verwendung von autonomen Filteragenten schafft zudem die Basis für die Skalierbarkeit der Anwendung, da überlastete Agenten durch das Hinzufügen weiterer Agenten entlastet werden können und zusätzliche Hardware in ein laufendes System integriert werden kann.

Kapitel 9

Zusammenfassung und Ausblick

Unter Verwendung von Multi-Agentensystemen lassen sich effektiv personalisierte Informationssysteme realisieren, die unterschiedliche Informationsquellen und Filterstrategien integrieren. Durch die adaptive Kombination von auf das Suchszenario und die individuellen Präferenzen des Benutzers abgestimmten Bewertungsmethoden und Strategien zum Kombinieren von Bewertungen werden zuverlässig hochqualitative Ergebnisse erreicht.

Anhand von verschiedenen für die Informationssuche charakteristischen Szenarien wurden die Anforderungen an ein personalisiertes Informationssystem identifiziert und eine Systemarchitektur abgeleitet. Es wurden die für die Elemente der Architektur entwickelten Komponenten und Algorithmen erläutert und die jeweiligen Besonderheiten diskutiert. Dabei wurden die Integration heterogener Datenquellen, die multidimensionale Bewertung von Dokumenten und die adaptive Kombination von Bewertungen unter Berücksichtigung szenariospezifischer Gütefunktionale im Detail analysiert. Die Leistungsfähigkeit der entwickelten Architektur und die Algorithmen wurden anhand mehrerer Informationssysteme gezeigt.

9.1 Multi-Agentensysteme

Die Architektur wurde auf der Basis einer Multi-Agentenplattform implementiert. Die Verwendung einer Multi-Agentenplattform ermöglicht die effiziente Realisierung eines offenen, verteilten Informationssystems. Die ontologiebasierte Kommunikation zwischen Agenten stellt sicher, dass Daten von den Agenten einheitlich interpretiert werden und neue Agenten in das System integriert werden können. Agentendienste können in Abhängigkeit von den individuellen Präferenzen des Benutzers verknüpft werden, womit eine Personalisierung des Informationssystems ermöglicht wird. Die Funktionen von

Multi-Agentensystemen zum Verschlüsseln von Daten und zum Prüfen der Identität von Agenten können genutzt werden, um die persönlichen Daten des Benutzers vor unberechtigten Zugriffen zu schützen. Die Funktionen zum Management von Agentensystemen ermöglichen das effektive Überwachen und Verwalten von verteilten Informationssystemen.

9.2 Systemarchitektur

Die entwickelten Informationssysteme verwenden eine serviceorientierten Architektur, die auf der Basis einer Multi-Agentenplattform implementiert werden kann. Die dienstbasierte Interaktion der Agenten ermöglicht es, Komponenten (Agenten) zur Laufzeit des Systems auszutauschen und hinzuzufügen sowie Ressourcen in mehreren Systemen gemeinsam zu nutzen. Durch die auf das jeweilige Suchszenario und die verfügbaren Systemressourcen abgestimmte Auswahl von Agenten können bei der Informationssuche die für eine Anfrage am besten geeigneten Informationsquellen, Bewertungsagenten und Algorithmen zum Zusammenstellen von optimalen Dokumentenmengen ausgewählt und kombiniert werden. Fehlerhafte Agenten können ersetzt und neue (oder verbesserte) Agenten in das System integriert werden. Die Nutzung der Infrastrukturdienste der Multi-Agentenplattform ermöglicht ein effektives Management einer großen Anzahl von intelligenten Agenten.

9.3 Kombination von Filterstrategien

Um auf die individuellen Präferenzen des Benutzers und das Suchszenario abgestimmte Filterergebnisse bereitzustellen, kombiniert man die Beurteilungen mehrerer Bewertungsagenten. Es wurden unterschiedliche Methoden zum Auswählen und Kombinieren von Bewertungen implementiert und bezüglich verschiedener Gütefunktionale evaluiert. Die durchgeführten Analysen zeigen, dass sowohl unter Verwendung inkrementell zusammengestellter Agentengremien als auch mit Agentengremien, die auf der Basis von Clustern erstellt wurden, hochqualitative Filterergebnisse erreicht werden können. Mit deutlich geringerem Berechnungsaufwand wird eine Ergebnisqualität erreicht, die die Filterpräzision der Benchmark-Algorithmen (z. B. RankSVM) übertrifft. Es wurde gezeigt, dass die Adaption der Agentengremien sowohl auf der Grundlage einer Menge von bewerteten Anfragen als auch auf der Basis von inkrementell abgegebenen Bewertungen erfolgen kann. Neben der Exaktheit von Filterergebnissen wurden alternative Gütemaße (wie z. B. die Verlässlichkeit oder die Übereinstimmung mit individuellen Präferenzen) analysiert. Die

durchgeführten Untersuchungen zeigen, dass die entwickelten Algorithmen das Erstellen von Gremien ermöglichen, die bezüglich dieser Gütefunktionale optimiert sind.

Die entwickelten Strategien zum Zusammenstellen von Agentengremien können in modifizierter Form genutzt werden, um Agentengremien an sich verändernde Benutzerpräferenzen anzupassen. Das Agentengremium wird dabei inkrementell an die Benutzerbewertungen oder an neue Bewertungsagenten adaptiert.

9.4 Personalisierung und Adaption

Die vorgestellte Architektur ermöglicht es, in Abhängigkeit von der Anfrage, dem Suchszenario und den Präferenzen des Benutzers optimierte Ergebnismengen zusammenzustellen. Es werden dabei unterschiedliche Relevanzkriterien (z. B. Relevanz auf Basis von Termstatistiken), vertrauensbasierte Kriterien (z. B. PageRank) mit aus dem Verhalten des Benutzers abgeleitete Relevanzkriterien (z. B. Ähnlichkeit mit bewerteten Dokumenten) kombiniert. Die gewichtete Kombination unterschiedlicher Arten von Relevanzbewertungen ermöglicht es, das Cold-Start-Problem von kollaborativen Informationssystemen zu umgehen, indem kollaborative Algorithmen erst dann berücksichtigt werden, wenn eine ausreichende Menge von Bewertungen vom aktuellen Benutzer vorliegen. Die Gewichtungen der zum Bewerten der Dokumente verwendeten Relevanzkriterien werden auf der Basis von Benutzerbewertungen angepasst.

9.5 Realisierte Informationssysteme

Die Evaluation der implementierten Informationssysteme zeigt, dass auf der Basis der entwickelten Architektur leistungsfähige Systeme realisiert werden können. Das Auswählen geeigneter Datenquellen und Bewertungsstrategien in Abhängigkeit vom jeweiligen Suchszenario ermöglicht das Bereitstellen hochqualitativer Filterergebnisse. Durch das Zusammenstellen von auf die Benutzerpräferenzen und das jeweilige Suchszenario optimierten Agentengremien können individuelle Benutzerwünsche und die besonderen Anforderungen des jeweiligen Szenarios berücksichtigt werden.

Durch die dienstbasierte Verknüpfung der Elemente des Informationssystems können bestehende Informationssysteme leicht an veränderte Anforderungen angepasst werden. Darüber hinaus können neue Informationssysteme auf der Basis bereits verfügbarer Komponenten realisiert werden. Die Aus-

tauschbarkeit von Komponenten zur Laufzeit ermöglicht es, Komponenten durch verbesserte Komponenten zu ersetzen oder für überlastete Komponenten zusätzliche Instanzen zu integrieren.

Die entwickelten Funktionen zum Optimieren von Anfragen unterstützen den Benutzer beim Verfeinern von unklaren Anfragen und beim Anpassen von Anfragen an die eigenen Interessengebiete. Die zu den Anfrageoptimierungsvorschlägen bereitgestellten Erläuterungen helfen dem Benutzer zu verstehen, warum ein vorgeschlagener Begriff zur Optimierung einer Anfrage geeignet ist. Dies erhöht die Akzeptanz der Anfrageoptimierungskomponente und motiviert den Benutzer, explizite Bewertungen abzugeben.

9.6 Alternative Anwendungsgebiete

Die entwickelten Verfahren zur Kombination von Bewertungsstrategien lassen sich in weitere Anwendungsgebiete übertragen. Auf die Präferenzen des Benutzers abgestimmte Agentengremien können immer dann zum Einsatz kommen, wenn verschiedene Faktoren beim Erstellen eines Ergebnisses berücksichtigt werden müssen. Das dynamische Zusammenstellen von Agentengremien ermöglicht es, bereits verfügbare (ggf. standardisierte) Komponenten entsprechend individueller Vorlieben (Gütefunktionale) zu kombinieren. Die entwickelten Verfahren stellen jeweils sicher, dass die berücksichtigten Faktoren aufeinander abgestimmt werden. Angewendet werden können optimierte Agentengremien somit immer dann, wenn bei der Entscheidungsfindung vielfältige Faktoren einfließen und die verwendete Gütefunktion dynamisch ist (z. B. Anpassung an Benutzerbewertungen). So können die Methoden beispielsweise im Rahmen von Empfehlungs- und Klassifikationssystemen eingesetzt werden [BL07, Pol06].

9.7 Ausblick

Das betrachtete Forschungsgebiet bietet vielfältige Möglichkeiten für weitergehende Untersuchungen. Ansätze, die detailliert analysiert werden sollten, werden im Folgenden erläutert.

Reinforcement-Learning: Reinforcement-Learning kann für unterschiedliche Optimierungsprobleme eingesetzt werden, wenn Benutzerbewertungen nicht direkt einer konkreten Lernaktion zugerechnet werden können („Learning based on delayed rewards“). Im Bereich des Ensemble-Learnings erscheint eine Anwendung von Reinforcement-Learning-Methoden aussichtsreich, um die Gewichtungen von Agenten zu optimieren und um eine kon-

tinuierliche Adaption an sich verändernde Benutzerpräferenzen zu erreichen [SB98, SDG+00, LA07].

Alternative Benchmarkdatensätze: Die verwendeten Benchmarkdatensätze (LETOR, TIPSTER) unterscheiden nur zwischen zwei Relevanzleveln („relevant", „irrelevant"). Um die Relevanz von Dokumenten differenzierter berücksichtigen zu können, sollten Datensätze mit Multi-Relevanzleveln betrachtet werden, wie sie bereits im Bereich von Empfehlungssystemen verwendet werden (z. B. JESTER [GDNG99]). Bei der Verwendung von Benchmarkdatensätzen mit mehreren Relevanzleveln müssen angepasste Gütefunktionale eingesetzt werden [ZZ07, SLA08]. Ziel für weitere Forschungsarbeiten sollte es sein, den Einfluss veränderter Gütefunktionale auf die Zusammensetzung von Filteragentengremien zu untersuchen.

Peer-To-Peer-Filterung: Die entwickelte Architektur ermöglicht die Realisierung von Informationssystemen als Peer-To-Peer-System, wobei über mehrere Computer verteilte Extraktions-, Bewertungs- und Filteragenten miteinander kooperieren. Es können dabei mehrere gleichberechtigte Agenten zum Steuern einer Informationssuche eingesetzt werden. Dies kann die Verfügbarkeit des Informationssystems verbessern, da bei Ausfall eines Agenten alternative Agenten herangezogen werden können. Das Management (das Registrieren und Finden von Agenten) sowie Protokolle zum Datenaustausch werden durch die Agentenplattform sichergestellt. Die Verteilung von Systemkomponenten auf unterschiedlich leistungsfähige und ggf. unzuverlässige Computer muss beim Zusammenstellen von Agentengremien berücksichtigt werden. Dies kann der Ausgangpunkt für weitergehende Untersuchungen sein.

Erweiterte Auswertung des Benutzerverhaltens: Durch eine Erweiterung der auf Basis der Architektur realisierten Informationssysteme können Daten über das Benutzerverhalten gewonnen werden. In Abhängigkeit von der jeweiligen Domäne können die aus dem Benutzerverhaltenen gewonnenen Informationen für eine bessere Anpassung an die individuellen Benutzerpräferenzen eingesetzt werden. Daten zum Verhalten des Benutzers können zum Verfeinern der verwendeten Bewertungsfunktionen genutzt werden, was die Grundlage für eine schnellere Adaption des Systems an Benutzerpräferenzen schafft.

Weitere Filterstrategien: Die entwickelte Architektur für ein offenes adaptives Informationssystem ermöglicht die Integration neuer Filterstrategien. Dabei ist insbesondere das Einbeziehen semantischer Bewertungsstrategien („semantic web") von Bedeutung [Mat08]. Interessant für weitergehende Forschungsarbeiten ist, inwieweit semantische Bewertungsstrategien in einem Gremium von Filteragenten die derzeit genutzten Agenten ergänzen bzw. ersetzen können.

Adaptive Bewertungsstrategien: In den betrachteten Szenarien wurde analysiert, wie Gremien von Agenten angepasst werden müssen, wenn neue Agenten hinzugefügt werden oder ein Agent nicht mehr verfügbar ist. Ein interessanter Spezialfall liegt vor, wenn Agenten ihre Filterstrategien selbständig anpassen. Bei der Zusammenstellung von Gremien von Filteragenten erfordert die Änderung der Filterstrategien eines Agenten auch die Anpassung der Zusammenstellung des Gremiums. In diesem Kontext sollte erforscht werden, wie die Strategien zum Adaptieren von Agentengremien das Optimieren einzelner Filterstrategien berücksichtigen können.

Anhang A

Strategien des LETOR-TD2004-Datensatzes

Tabelle A.1: Die Filterstrategien des LETOR-TD2004-Datensatzes

ID	Beschreibung
0	**BM25:** [RWB98]
1	dl **of body** [BYRN99]
2	dl **of anchor** [BYRN99]
3	dl **of title** [BYRN99]
4	dl **of URL** [BYRN99]
5	**HITS authority** [Kle99]
6	**HITS hub** [Kle99]
7	**HostRank** [XYZ+05]
8	Idf **of body** [BYRN99]
9	Idf **of anchor** [BYRN99]
10	Idf **of title** [BYRN99]
11	Idf **of URL** [BYRN99]
12	**Sitemap based feature propagation** [QLZ+05]
13	**PageRank** [PBMW98]
14	**LMIR.ABS of anchor** [ZL01]
15	**BM25 of anchor** [RWB98]
16	**LMIR.DIR of anchor** [ZL01]
17	**LMIR.JM of anchor** [ZL01]
18	**LMIR.ABS of extracted title** [ZL01]
19	**BM25 of extracted title** [RWB98]
20	**LMIR.DIR of extracted title** [ZL01]
21	**LMIR.JM auf Basis des adaptierten Dokumententitels** [ZL01]
22	**LMIR.ABS auf Basis des Dokumententitels** [ZL01]
23	**BM25 auf Basis des Dokumententitels** [RWB98]
24	**LMIR.DIR auf Basis des Dokumententitels** [ZL01]
25	**LMIR.JM auf Basis des Dokumententitels** [ZL01]
26	**Sitemap based feature propagation** [QLZ+05]
27	Tf **of body** [BYRN99]
28	Tf **of anchor** [BYRN99]
29	Tf **of title** [BYRN99]
30	Tf **of URL** [BYRN99]
31	Tf – Idf **of body** [BYRN99]
32	Tf – Idf **of anchor** [BYRN99]
33	Tf – Idf **of title** [BYRN99]
34	Tf – Idf **of URL** [BYRN99]
35	**Topical PageRank** [NDQ06]
36	**Topical HITS authority** [NDQ06]
37	**Topical HITS hub** [NDQ06]
38	**Hyperlink base score propagation: weighted in-link** [SZ03]
39	**Hyperlink base score propagation: weighted out-link** [SZ03]
40	**Hyperlink base score propagation: uniform out-link** [SZ03]
41	**Hyperlink base feature propagation: weighted in-link** [SZ03]
42	**Hyperlink base feature propagation: weighted out-link** [SZ03]
43	**Hyperlink base feature propagation: uniform out-link** [SZ03]

Anhang B

Glossar

Agent	Ein Agent ist eine autonom agierende Softwareentität, die in einem System von mehreren Agenten eine spezielle Funktionalität oder einen Dienst anbietet. Ein Agent interagiert mit anderen Agenten in einer formalen Agenten-Kommunikationssprache [Ses01].
Agenten-Migration	Agentenmigration beschreibt den Vorgang, bei dem ein Agent von einer Laufzeitumgebung in eine andere Laufzeitumgebung (z. B. auf einer neuen Maschine) wechselt. Man unterscheidet zwischen *starker Migration*, bei der der komplette Zustand des Agenten übertragen wird, und *schwacher Migration*, bei der lediglich der Programmcode, nicht aber der Zustand übertragen wird und somit der Agent in der neuen Laufzeitumgebung neu initialisiert werden muss.
AJAX	AJAX ist ein Akronym für **A**synchronous **J**avaScript **A**nd **X**ML. Es bezeichnet ein Konzept der asynchronen Datenübertragung zwischen einem Web-Server und dem Browser, das es ermöglicht, innerhalb von HTML-Seiten HTTP-Anfragen durchzuführen, ohne die HTML-Seite komplett neu laden zu müssen. Unter Verwendung von AJAX ist es möglich, Teile einer HTML-Seite dynamisch anzupassen oder Daten erst sukzessiv bei Bedarf nachzuladen.

Anfrage, Such-anfrage, Query	Im Bereich des IR wird das Finden und Bewerten von zu einer Fragestellung passenden Dokumenten untersucht. Die zur Suche verwendete formal formulierte Fragestellung wird dabei als Anfrage oder Query bezeichnet. Eine Anfrage kann unterschiedliche Merkmale beinhalten. Häufig zur Formulierung von Anfragen verwendete Kriterien sind: Wörter, Phrasen, Dokumente, bewertete Objekte sowie semantische beschriebene Daten.
Anytime-Algorithmus	Anytime-Algorithmen werden oft auch als unterbrechbare Algorithmen bezeichnet. Im Unterschied zu Algorithmen, die Ergebnisse erst nach dem kompletten Abarbeiten aller Schritte bereitstellen, können von Anytime-Algorithmen schon nach wenigen Berechnungsschritten Ergebnisse abgefragt werden. Damit ist es möglich diese Algorithmen vorzeitig abzubrechen und die bereits ermittelten (ggf. partiellen) Lösungen zu nutzen.
Benutzer-Profil	Bei der Realisierung von personalisierten Informationssystemen werden Benutzerprofile verwendet, um die über den Benutzer verfügbaren Informationen zu speichern. Das Benutzerprofil umfasst im Allgemeinen Daten zur Identität sowie den Interessen und Vorlieben des Benutzers. Darüber hinaus können im Benutzerprofil Daten zum Verhalten des Benutzers gespeichert werden (z. B. Häufigkeit der Systembenutzung, bevorzugt genutzte Funktionen). Im Rahmen von Informationssystemen können die im Benutzerprofil gespeicherten Informationen zu den Interessen und dem Verhalten des Benutzers eingesetzt werden, um Anfragen zu optimieren und die Bewertung von Dokumenten an die individuellen Benutzerinteressen anzupassen.
BDI-Agenten	Das BDI-Modell (Akronym für **B**elief, **D**esire, **I**ntention) wird hauptsächlich zur Realisierung deliberativen Agentenarchitekturen eingesetzt. Die Agenten erstellen Annahmen über ihre Umwelt, verfügen Wissen über erstrebenswerte Zustände und besitzen Absichten zum Ausführen von Aktionen. Der Einsatz des BDI-Modells soll es dem Agenten ermöglichen, ein eigenes Verhalten zu entwickeln und unter Berücksichtigung der Umgebung seine Ziele zu erreichen.

Blog, Web-Log	Ein We**blog** ist ein öffentlich einsehbares Tagebuch oder Journal. Blogs werden im Allgemeinen als eine Webseite realisiert, die periodisch neue Einträge enthält, wobei neue Einträge an oberster Stelle stehen; ältere Einträge folgen in umgekehrt chronologischer Reihenfolge. Blogs konzentrieren sich oft auf einen Themenbereich, z. B. Politik, Lokales oder Essen. Ein typisches Blog kombiniert Texte, Bilder und Referenzen zu anderen Blogs und Webseiten.
Boosting	Boosting beschreibt einen Meta-Klassifikationsalgorithmus, bei dem mehrere Klassifikatoren zu einem einzigen Klassifikator zusammengefasst werden. Beim Boosting ermittelt man eine gewichtete Kombination mehrerer (meist einfacher) Klassifikatoren. Die Optimierung der Gewichte erfolgt dabei iterativ, wobei der Einfluss der Elemente der Trainingsmenge in Abhängigkeit vom Trainingsfehler angepasst wird. Boosting-Algorithmen zeichnen sich im Allgemeinen dadurch aus, dass sie bei geringem Bedarf an Ressourcen gute Ergebnisse liefern, wobei die Klassifikationsgüte des durch Boosting erstellten Klassifikators über der Güte der bei der Kombination berücksichtigten einzelnen Klassifikatoren liegt [Bis06].
C4.5	C4.5 ist ein Entscheidungsbaum basierter Klassifikationsalgorithmus. Der Algorithmus unterstützt sowohl diskrete als auch reellwertige Objektattribute. Zum Aufbau des Entscheidungsbaumes wird eine Greedy-Strategie verwendet, die als Kriterium zur Auswahl des zu verwendenden Attributes für einen Knoten ein Maß für den Informationsgewinn verwendet („gain ratio“). Generierte Unterbäume werden auf ihren Nutzen geprüft und ggf. reduziert („Error Pruning“)[Qui93].

Cloaking	Cloaking ist eine Technik um die Platzierung von Dokumenten bei Suchmaschinen zu verbessern. Es wird dabei dem Indizierungsagenten („Robot“) eine andere Seite präsentiert, als die Benutzer sehen, die mit einem Browser auf die Internet-Seite zugreifen. Cloaking kann eingesetzt werden, um die richtige Einordnung der Seite für die Suchmaschine zu erleichtern, z. B. indem dem Indizierungsagenten eine textbasierte, strukturell optimierte HTML-Seite präsentiert wird, die mit speziellen Metainformationen (Meta-Tags) angereichert wird. Cloaking kann dazu missbraucht werden, um der Suchmaschine nicht vorhandene Inhalte vorzuspiegeln. Maßnahmen, um das Vortäuschen nicht vorhandener Inhalte einzuschränken, bestehen in der Verwendung verschiedener Indizierungsagenten (User-Agent-IDs) sowie dem Ausschließen von Informationsanbietern, die Cloaking zum Vortäuschen von Inhalten verwenden.
Clone, Klone	Multi-Agenten-Plattformen bieten im Allgemeinen die Möglichkeit, Agenten zu duplizieren. Diese Funktionalität kann genutzt werden, um eine Lastverteilung zu erreichen oder die Ausfallsicherheit zu erhöhen. Werden in Multi-Agenten Wahlverfahren realisiert, bei denen die alle Wähler dieselbe Gewichtung besitzen, kann bei einigen Wahlverfahren (z. B. CombSUM) durch das Duplizieren von Agenten das Ergebnis beeinflusst werden. Durch das Klonen eines Agenten kann dabei die Gewichtung des Agenten erhöht werden. Bei Verwendung eines Mehrheitswahlsystem und bei Verwendung der BORDA-Wahl kann durch das Duplizieren von Kandidaten das Ergebnis beeinflusst werden, wobei für geklonte Kandidaten die Chance sinkt, die Wahl zu gewinnen. Wahlen nach dem CONDORCET-Verfahren sind dagegen unempfindlich gegen das Duplizieren von Kandidaten.

Collection fusion	Das Erstellen einer einheitlichen Dokumentenmenge aus mehreren, untereinander disjunkten Dokumentenmengen bezeichnet man als *Collection fusion*. Dieser Anwendungsfall tritt im Bereich des IR auf, wenn Dokumente aus unterschiedlichen Datenbanken oder Informationsportalen zu einer Dokumentenmenge zusammengefasst werden sollen. Dabei stellt jede Datenquelle eine Menge relevanter Dokumente bereit. Auf Basis dieser Dokumente muss dann eine Ergebnismenge zusammengestellt werden. Die Schwierigkeit besteht dabei darin, dass die aus den verschiedenen Quellen stammenden Bewertungen unterschiedliche Bewertungen aufweisen, die untereinander nicht direkt vergleichbar sind.
Data fusion	Das Zusammenführen von unterschiedlichen Bewertungen für die Elemente eine Dokumentenmenge bezeichnet man als *data fusion*. *Data fusion* Verfahren werden im Allgemeinen dann eingesetzt, wenn die Bewertungen von unterschiedlichen Filterstrategien zu einer gemeinsamen Bewertung zusammengefasst werden müssen.
DMOZ	DMOZ ist ein Akronym für „**D**irectory at **Moz**illa“ und bezeichnet das Open Directory Project (ODP). Das Projekt verfolgt das Ziel, den umfangreichsten von Menschen erstellen Link-Katalog des World Wide Web anzulegen. Das Erstellen und Aktualisieren des Verzeichnisses erfolgt durch eine Gemeinschaft freiwilliger Editoren, die jeweils für bestimmte Themenbereiche zuständig sind. Die von DMOZ bereitgestellten Daten stehen als *Open Content* zur Verfügung und können im RDF-Format bezogen werden. Das Open Directory Project speichert über 4,8 Millionen Internetadressen und ca. 590 000 Kategorien (Oktober 2007).
Dokument	Der Begriff „Dokument“ bezeichnet „Datenobjekte“, die durch eine Menge von Attributen beschrieben werden. Ein Datenobjekt kann dabei einen Text, einen Film, ein Bild oder eine Konferenz repräsentieren. Im engeren Sinne werden unter Dokumenten Texte verstanden, die durch eine ID, den Inhalt und eine Menge von Meta-Daten (z. B. Worthäufigkeiten, Erstellungsdatum, Autor) beschrieben werden.

EPG	Der elektronische Programmführer (Electronic Program Guide – EPG) ist ein von vielen digital ausgestrahlten Fernsehsendern angebotener Dienst, der Informationen über das laufende und kommende Fernseh- oder Radioprogramm bereitstellt. Die Programmübersicht beinhaltet mindestens den Titel, die Uhrzeit und die Dauer jeder Sendung. Zusätzlich können zu den einzelnen Sendungen kurze Beschreibungen des Inhalts, die Programmart (Kategorie bzw. Klassifikation, „content descriptor"), Informationen zu einer Altersbeschränkung der Sendung sowie Informationen zum Bild- und Tonformat und der verwendeten Bitrate zur Verfügung gestellt werden.
Feature Extraction, Merkmalsextraktion	Feature Extraction beschreibt den Vorgang der Gewinnung von aussagekräftigen Merkmalen aus Dokumenten. Dabei wird (meist unter Verwendung von Daten über die Domäne) aus den Dokumentendaten Wissen (Merkmale) abgeleitet, was eine Verarbeitung der Dokumente erleichtert.
Filter, Filteragent	Ein Filter bzw. ein Filteragent ist eine Komponente, die Dokumente bezüglich der Relevanz bzw. Qualität bewertet. Der Filteragent ermittelt für jedes Element einer Dokumentenmenge eine Bewertung (z. B. in Form einer numerischen Relevanzbewertung, eines Rankings oder einer booleschen Bewertung). Die Entscheidung, welche Dokumente dem Benutzer präsentiert werden sollten, übernimmt ein Agent, der die Bewertungen mehrerer Filteragenten kombiniert und unter Berücksichtigung des jeweiligen Szenarios und der individuellen Benutzerpräferenzen eine optimierte Ergebnismenge zusammenstellt.
GermaNet	→ WordNet
ID3	ID3 ist ein von J. R. Quinlan entwickelter Entscheidungsbaum-Algorithmus. Der Algorithmus ID3 erstellt Entscheidungsbäume iterativ, wobei als Kriterium für die Auswahl des für die Erstellung eines neuen Knotens zu verwendenden Merkmals die Entropie auf den Trainingsdaten eingesetzt wird. ID3 wird vor allem für große Datenmengen eingesetzt, für die ohne große Berechnungen ein Entscheidungsbaum generiert werden soll. Eine Weiterentwicklung von ID3-Algorithmus ist der Klassifikator C4.5 [Mit97].

Index	Ein Index bezeichnet eine Datenstruktur, die einen schnellen Zugriff auf Daten in einer umfangreichen Datensammlung ermöglicht. Um bei der Suche nach einem Datensatz im ungünstigsten Fall nicht den kompletten Datenbestand durchsuchen zu müssen, erstellt man Indizes, die es ermöglichen, anhand von Attributwerten die Position des passenden Datensatzes innerhalb der Sammlung schnell zu bestimmen. Typische Datenstrukturen für das Erstellen eines Index sind B-Bäume, Hashtabellen und invertierte Listen.
Information Retrieval, IR	Information Retrieval (IR) bzw. Informationswiedergewinnung beschäftigt sich mit der computergestützten Suche in großen Datensammlungen. Im Gegensatz zur Suche von Informationen in fest strukturierten Datenbanken beschäftigt sich das IR mit der Behandlung von Vagheit (unscharfe Übereinstimmungen, unpräzise Anfragen) und Unsicherheiten (z. B. homographen Begriffen und Synonymen). Bei der Suche und Bewertung der Relevanz von Daten werden im Allgemeinen Verfahren zum effizienten Abspeichern von Daten (z. B. Indizes) mit Verfahren des maschinellen Lernens (z. B. Klassifikatoren, Künstliche Neuronale Netze) miteinander verknüpft.
Java-VM	Die Java Virtual Machine (JAVA VM oder JVM) ist der Teil der JAVA-Laufzeitumgebung (JRE) für JAVA-Programme, die für die Ausführung des JAVA-Bytecodes verantwortlich ist. Der andere Teil der Java-Laufzeitumgebung sind die Java-Klassenbibliotheken. Die JVM dient dabei als Schnittstelle zu der Maschine und dem Betriebssystem.
JIAC IV	JIAC IV (Akronym für „**J**ava **I**ntelligent **A**gent **C**omponentware Version 4“) ist ein JAVA-basierte Multi-Agentenframework zum Erstellung von dienstorientierten Multiagentensystemen. Die zentralen Konzepte der JIAC IV-Architektur sind die Unterstützung von Ontologien, der Aufbau von Agenten auf Basis von Komponenten sowie die Verwendung von Diensten als Grundlage für die Interaktion zwischen Agenten. JIAC IV ermöglicht die Realisierung offener, skalierbarer und flexibler Anwendungen, wobei zahlreiche bereits verfügbare Komponenten genutzt werden können.

Klassifikator	Klassifikationsverfahren sind Methoden zur Einteilung von Objekten in Klassen. Bei in Software implementierten Klassifikatoren spricht man von maschineller oder automatischer Klassifikation. Wichtige Anwendungsgebiete für Klassifikatoren sind die Mustererkennung, das maschinelle Lernen sowie das Information Retrieval. Bekannte Klassifikationsverfahren sind Abstandsklassifikatoren, Bayes-Klassifikatoren, Nächste-Nachbarn-Klassifikatoren, Künstliche neuronale Netze sowie Support-Vector-Maschinen.
Keyword Spamming	Keyword Spamming ist eine Technik um die Platzierung eines Dokumentes in der Ergebnisliste von Suchmaschinen zu verbessern. Es wird dabei versucht, einem Dokument populäre Suchbegriffe zuzuordnen. Methoden des Keyword-Spammings sind das Wiederholen von Begriffen in `HTML`-Kommentaren, im Seitentitel und in den `ALT`-Tags von Bildern.
Kreuz-entropie	Die Kreuzentropie wird in der Informationstheorie verwendet, um die Ähnlichkeit zwischen zwei Wahrscheinlichkeitsverteilungen zu beschreiben. Die Kreuzentropie zwischen zwei Wahrscheinlichkeitsverteilungen p, q über demselben Ereignishorizont gibt an, wie viele Bits durchschnittlich benötigt werden, um bezüglich einer Wahrscheinlichkeitsverteilung p verteilte Ereignisse mit einem Kodierungsschema zu beschreiben, das auf einer Wahrscheinlichkeitsverteilung q basiert.
LUCENE	APACHE LUCENE (`http://lucene.apache.org/`) ist eine für verschiedene Programmiersprachen (z. B. JAVA und C) Open-Source-Bibliothek zum Erstellen und Durchsuchen von Indizes. LUCENE bietet zahlreiche grundlegende Funktionen zum Verarbeiten und Indizieren von Texten (z. B. Zerlegung von Texten in Wörter, Entfernen von Stoppwörtern, Durchführen von Stemming) und bietet Schnittstellen für domänenspezifische Erweiterungen und Anpassungen.

Metadaten	Als Metadaten bezeichnet man Daten, die andere Daten beschreiben. Im Bereich des Dokumentenretrievals werden Metadaten zum Beschreiben der Eigenschaften von Dokumenten und Dokumentensammlungen verwendet. Metadaten, wie z. B. der Autor, die Sprache oder die Länge eines Textes bilden die Grundlage für eine umfassende Bewertung der Relevanz eines Dokumentes. Darüber hinaus können Metadaten eine verbesserte, auf die Präferenzen des Benutzers abgestimmte Präsentation von Dokumenten ermöglichen.
Metasuchmaschine	Eine Metasuchmaschine ist eine Suchmaschine, die Suchanfragen an mehrere andere Suchmaschinen weiterleitet, die Ergebnisse sammelt und aufbereitet. Die ermittelten Dokumente werden meist (wie die Ergebnisse einer Suchmaschine) in einer Liste dargestellt.
Middleware	Middleware bezeichnet Technologien, die Funktionen zur Vermittlung zwischen Anwendungen anbieten, so dass die Komplexität der zugrunde liegenden Applikationen und Infrastruktur verborgen wird [RMB00]. Unter Verwendung von Middleware können Softwarekomponenten auf mehreren Computern untereinander gekoppelt werden. Middleware kann dabei verschiedene Aufgaben übernehmen, wie z. B. den Transport komplexer Daten („Messaging“), das Registrieren und Verwalten von Diensten und Komponenten („Verzeichnisdienst“), das Vermitteln von Funktionsaufrufen („Remote Procedure Calls“) oder das Herstellen von Transaktionssicherheit.

Ontologie	In der analytischen Philosophie bilden Ontologien die Lehre von den grundlegenden Kategorien, d. h. allgemeinen Begriffen (wie Ding, Eigenschaft oder Ereignis) und Beziehungen (wie Teil und Ganzes sowie Abhängigkeiten zwischen Begriffen). Dabei steht im Mittelpunkt der Betrachtung, wie sich die verschiedenen Kategorien zueinander verhalten und was eine Kategorie ausmacht. In der Informatik (insbesondere im Bereich der Agententheorie) versteht man unter einer Ontologie die explizite Spezifikation der Struktur eines Bereiches (Domäne). Mit der Verwendung von Ontologien wird das Ziel verfolgt, die Entwicklung und Nutzung interoperabler Anwendungen zu ermöglichen. Ontologien umfassen ein kontrolliertes Vokabular (z. B. eine Menge von logischen Konstanten und Prädikaten), eine Menge von logischen Ausdrücken, die Constraints und Abhängigkeiten innerhalb der Domäne beschreiben. Ontologien stellen somit ein Vokabular zur Speicherung und Kommunikation von Wissen bereit und ermöglichen damit die explizite Repräsentation von Entitäten, Eigenschaften und Beziehungen.
Präzision, Precision, Exaktheit	Um zu bestimmen, welcher Anteil der Dokumente in einer Filterergebnismenge relevant ist, werden Maße zur Beschreibung der Exaktheit des Filters verwendet. Für unsortierte Dokumentenmengen, bei denen zwischen zwei Relevanzleveln („relevant“, „irrelevant“) unterschieden wird, verwendet man das Maß Precision: $$\mathrm{Prec}(R, D) = \frac{|\mathrm{Rel}(R)|}{|\mathrm{Rel}(D)|}$$ Weitere Maße zum Bewerten der Exaktheit eines Filters sind die Mean Average Precision (MAP) und der Normalized Discounted Cumulative Gain (NDCG).

Ranking, Rangfolge, Rangliste	Ein Ranking definiert eine Ordnung von mehreren untereinander vergleichbaren Objekten. Im Bereich des IR werden Rankings verwendet, um Dokumente nach bestimmten Kriterien zu sortieren, z. B. in Hinsicht auf die Relevanz bezüglich einer Suchanfrage. Im Gegensatz zu Ratings ermöglichen Rankings keine absolute Aussage über die Güte eines Dokumentes, sondern lediglich ob ein Dokument besser oder schlechter (bezüglich des gewählten Kriteriums) ist als ein anderes Dokument.
Rating	Ein Rating beschreibt im Allgemeinen die Bewertung für ein Objekt. Im Bereich des Dokumentenretrievals wird unter einem Rating meist eine reellwertige Bewertung verstanden, die auf das Intervall $[0, 1]$ skaliert ist. Dabei steht ein Rating von 1 für ein sehr gutes, eine Rating von 0 für ein sehr schlechtes Dokument. Aus einer Menge von Dokumenten mit reellwertige Ratings kann ein Ranking von Dokumenten erstellt werden.
Recall	Der Recall definiert ein im Bereich des IR verwendetes Maß für die Vollständigkeit einer Ergebnismenge beim Filtern von Dokumenten. Der Recall ist definiert als das Verhältnis zwischen der Anzahl der relevanten Dokumente in der Ergebnismenge zur Gesamtzahl der relevanten Dokumente.
Rocchio-Algorithmus	Der Rocchio-Algorithmus ist ein statistisches Klassifikationsverfahren, das als Basis das Vektorraummodell verwendet. Die zu klassifizierenden Objekte werden dabei als (normierte) Attributvektoren beschrieben. Im Bereich des IR werden zur Beschreibung von Textdokumenten $\mathrm{Tf} - \mathrm{Idf}$-Vektoren verwendet. Voraussetzung für die Anwendung des Rocchio-Algorithmus ist das Vorliegen von Trainingsdokumenten D, die zu unterschiedlichen Klassen gehören. Auf Basis der Trainingsvektoren berechnet der Algorithmus für jede Klasse j einen Prototyp-Vektor c_j. $$c_j = \alpha \frac{1}{|C_j|} \sum_{d \in C} \frac{\mathrm{Tf} - \mathrm{Idf}(d)}{|\mathrm{Tf} - \mathrm{Idf}(d)|} - \beta \frac{1}{|D \backslash C_j|} \sum_{d \in D \backslash C_j} \frac{\mathrm{Tf} - \mathrm{Idf}(d)}{|\mathrm{Tf} - \mathrm{Idf}(d)|}$$

	Zur Klassifikation eines neuen Dokumentes wird die Kosinus-Distanz des Merkmalsvektors des Dokumentes zu den Prototyp-Vektoren der Klassen berechnet. Das neue Dokument wird der Klasse zugeordnet, zu der der Abstand am geringsten ist. Der Algorithmus kann auch iterativ angewendet werden, indem bei neu vorhandenen Bewertungen die ermittelten prototypischen Vektoren entsprechend angepasst werden, bis der neue Datensatz korrekt klassifiziert wird.
Service	Ein Service ist eine Leistung, die von einem Anbieter für einen Nutzer erbracht wird. Der Service wird durch eine Beschreibung der Leistung definiert, die davon abstrahiert, wie der Service erbracht wird. Im Kontext von serviceorientierten Architekturen versteht man unter einem Service eine Software-Komponente, die eine definierte Funktionalität über eine standardisierte Schnittstelle anderen Diensten oder Anwendungen zur Verfügung stellt. Für das Auffinden, die Identifizierung und Beschreibung eines Services werden im Allgemeinen Verzeichnisdienste und Aushandlungsprotokolle eingesetzt. Technologien und Protokolle, die für die Implementierung von Diensten häufig verwendet werden, sind z. B. XML, HTTP, WSDL, SOAP.
Social-Tagging-Dienst	Social-Tagging-Dienste stellen Funktionen bereit, die es unter Verwendung von Web-Browsern erlauben, Webseiten Schlagwörter („Tags“) zuzuordnen. Die Nutzer von Social Tagging Diensten können für neue Webinhalte Tags erstellen sowie bereits vergebene Tags erweitern, löschen und kommentieren. Social Tagging Dienste bieten zudem die Möglichkeit, die Tags anderer Benutzer einzusehen und sich über neue Inhalte mit Hilfe von RSS-Feeds benachrichtigen zu lassen. Bekannte Social Tagging Dienste sind `http://furl.net` und `http://del.icio.us`.
Stacking	Stacking ist eine Methode des Ensemble-Lernens. Es werden dabei mehrere Verfahren miteinander verbunden, so dass die Verknüpfung nach außen als eine Einheit wahrgenommen wird. Beim Stacking werden Verfahren nacheinander angewendet. Die durch die Verknüpfung mehrerer Verfahren gewonnenen Ergebnisse werden dabei wiederum als Eingabedaten für eine weitere Verknüpfung verwendet.

SVM	SVM (Abkürzung für **S**upport-**V**ektor-**M**aschine) ist ein Klassifikationsverfahren für das überwachte Lernen von Objekt-Klassen-Zuordnungen. Die Funktionsweise der SVM beruht auf dem Ermitteln einer optimal trennenden Hyperebene in einem hochdimensionalen Merkmalsraum, so dass um die Klassengrenzen herum ein möglichst breiter Bereich frei von Objekten bleibt („Large Margin Classifier“). Man versucht, eine passende Abbildung zu ermitteln („Kernel-Trick“), die die Daten so in den Raum projiziert, dass die Daten aus zwei Kategorien stets mit Hilfe einer Hyperebene separiert werden können.
$\mathrm{Tf}-\mathrm{Idf}$	Ein häufig verwendetes Maß, um die Relevanz eines Textes in Bezug auf einen Begriff zu beschreiben, ist $\mathrm{Tf}-\mathrm{Idf}$ (**t**erm **f**requency – **i**nverse **d**ocument **f**requency). Man berechnet das Verhältnis zwischen der Häufigkeit eines Begriffes im Text (tf) zur Häufigkeit des Begriffes in der Menge aller Texte (df). $$\mathrm{Tf}-\mathrm{Idf}(a,d,D)=\frac{\log(\mathrm{tf}(a,d))}{\log(\mathrm{df}(a,D))}$$ Ein hoher $\mathrm{Tf}-\mathrm{Idf}(a,d,D)$-Wert zeigt an, dass ein Begriff a in einem Dokument d häufig vorkommt und in der Gesamtheit der Dokumente D selten auftritt.
TIPSTER	Das TIPSTER Projekt verfolgt das Ziel, die Entwicklung von Methoden des Information Retrievals auf großen, in der Realität auftretenden Daten zu fördern. Das TIPSTER Projekt stellt einen Benchmark-Datensatz bereit, um die Leistungsfähigkeit von Text-Retrieval-Algorithmen zu bewerten. Die Datensammlung besteht aus SGML kodierten Dokumenten, Anfragen sowie Relevanzbewertungen (die definieren, ob ein Dokument für eine Anfrage relevant ist). Die Dokumentenmenge wird sowohl für das TIPSTER-Projekt als auch im Rahmen von TREC eingesetzt.
TREC	TREC ist eine Abkürzung für „**T**ext **RE**trieval **C**onference“. Schwerpunkte von TREC liegen im Bereich des Filtern und Bewerten von Textdokumenten. Im Rahmen der Konferenz wurde ein Datensatz entwickelt (Anfragen und passende Dokumente) anhand dessen Konferenzteilnehmer ihre Verfahren testen und bewerten können.

WIKIPEDIA	WIKIPEDIA ist eine kollaborativ erstellte Enzyklopädie, die jedem Interessierten die Möglichkeit gibt, Texte zu Wissensgebieten zu erstellen, zu ergänzen oder zu kürzen. Der Aufbau von Artikeln orientiert sich an einer einheitlichen Struktur, die hilft, Informationen schnell zu finden. WIKIPEDIA-Artikel können Kategorien zugeordnet werden (z. B. „exzellenter Artikel“, „Stub“) sowie mit Markierungen versehene werden (z. B. „kontrovers“ oder „überarbeitungsbedürftig“).
WORDNET	WORDNET ist ein kostenlos verfügbarer, seit 1985 am COGNITIVE SCIENCE LABORATORY der Princeton University (`http://wordnet.princeton.edu/`) entwickelter Wortschatz der englischen Sprache. WORDNET besteht aus einer lexikalischen Datenbank, die semantische und lexikalische Beziehungen zwischen Wörtern enthält. WORDNET wurde mit dem Ziel entwickelt, das automatische Erkennen der Semantik von Texten zu verbessern. Für die deutsche Sprache existiert das an der Universität Tübingen entwickelte GERMANET (`http://www.sfs.uni-tuebingen.de/lsd/`). Daten zu weiteren europäischen Sprachen stellt das Projekt EUROWORDNET (`http://www.illc.uva.nl/EuroWordNet/`) bereit.

Literaturverzeichnis

[ACD+07] Acharya, Anurag ; Cutts, Matt ; Dean, Jeffrey ; Haahr, Paul ; Henzinger, Monika ; Lawrence, Steve ; Pfleger, Karl ; Tong, Simon: Document scoring based on link-based criteria. In: *European Patent Office* (2007). – http://v3.espacenet.com/textdoc?DB=EPODOC&IDX=EP1668551&CY=ep

[AM00] Aslam, Javed A. ; Montague, Mark: Bayes optimal metasearch: A probabilistic model for combining the results of multiple retrieval systems. In: *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2000. – ISBN 1–58113–226–3, S. 379–381

[AM01] Aslam, Javed A. ; Montague, Mark: Models for metasearch. In: *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2001. – ISBN 1–58113–331–6, S. 276–284

[AM05] Kap. 1 In: Anand, Sarabjot S. ; Mobasher, Bamshad: *Intelligent Techniques for Web Personalization*. Springer Berlin, Heidelberg, 2005, S. 1–36. – ISBN 978-3-540-29846-5

[AMP05] Abrams, Zoë ; McGrew, Robert ; Plotkin, Serge: A non-manipulable trust system based on EigenTrust. In: *SIGecom Exch.* 5 (2005), Nr. 4, S. 21–30. – ISSN 1551–9031

[AMSWH06] Artemenko, Olga ; Mandl, Thomas ; Shramko, Margaryta ; Womser-Hacker, Christa: Evaluation of a Language Identification System for mono- and multilingual Text Documents. In: *SAC '06: Proceedings of the 2006 ACM symposium on Ap-*

plied computing. New York, NY, USA : ACM, 2006. – ISBN 1–59593–108–2, S. 859–860

[APS03] Aslam, Javed A. ; Pavlu, Virgiliu ; Savell, Robert: A unified model for metasearch, pooling, and system evaluation. In: *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management*. New York, NY, USA : ACM Press, 2003. – ISBN 1–58113–723–0, S. 484–491

[APY06] Aslam, Javed A. ; Pavlu, Virgil ; Yilmaz, Emine: A statistical method for system evaluation using incomplete judgments. In: *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2006. – ISBN 1–59593–369–7, S. 541–548

[ASST05] Adomavicius, Gediminas ; Sankaranarayanan, Ramesh ; Sen, Shahana ; Tuzhilin, Alexander: Incorporating contextual information in recommender systems using a multidimensional approach. In: *ACM Trans. Inf. Syst.* 23 (2005), Nr. 1, S. 103–145. – ISSN 1046–8188

[AWLM07] Albayrak, Sahin ; Wollny, Stefan ; Lommatzsch, Andreas ; Milosevic, Dragan: Agent Technology for Personalized Information Filtering: The PIA-System. In: *Scalable Computing: Practice and Experience* 8 (2007), S. 29–40

[AWV+05] Albayrak, Sahin ; Wollny, Stefan ; Varone, Nicolas ; Lommatzsch, Andreas ; Milosevic, Dragan: Agent technology for personalized information filtering: The PIA-system. In: *Proceedings of the 2005 ACM Symposium on Applied Computing (SAC), Santa Fe, New Mexico, USA, March 13-17, 2005*, ACM, 2005. – ISBN 1–58113–964–0, S. 54–59

[BBM02] Billhardt, Holger ; Borrajo, Daniel ; Maojo, Víctor: Using genetic algorithms to find suboptimal retrieval expert combinations. In: *SAC '02: Proceedings of the 2002 ACM symposium on Applied computing*. New York, NY, USA : ACM Press, 2002. – ISBN 1–58113–445–2, S. 657–662

[BCB94] Bartell, Brian T. ; Cottrell, Garrison W. ; Belew, Richard K.: Automatic combination of multiple ranked retrieval

systems. In: *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : Springer-Verlag New York, Inc., 1994. – ISBN 0–387–19889–X, S. 173–181

[BCCC93] Belkin, Nicholas J. ; Cool, C. ; Croft, W. B. ; Callan, James P.: The effect multiple query representations on information retrieval system performance. In: *SIGIR '93: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 1993. – ISBN 0–89791–605–0, S. 339–346

[Ber05] Berkhin, Pavel: Survey: A Survey on PageRank Computing. In: *Internet Mathematics* 2 (2005), Nr. 1

[BFJ+03] Beitzel, Steven M. ; Frieder, Ophir ; Jensen, Eric C. ; Grossman, David ; Chowdhury, Abdur ; Goharian, Nazli: Disproving the fusion hypothesis: an analysis of data fusion via effective information retrieval strategies. In: *SAC '03: Proceedings of the 2003 ACM symposium on Applied computing*. New York, NY, USA : ACM, 2003. – ISBN 1–58113–624–2, S. 823–827

[BHZ06] Bordeaux, Lucas ; Hamadi, Youssef ; Zhang, Lintao: Propositional Satisfiability and Constraint Programming: A comparative survey. In: *ACM Comput. Surv.* 38 (2006), Nr. 4, S. 12. – ISSN 0360–0300

[Bis06] Bishop, Christopher M.: *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, August 2006. – ISBN 0387310738

[BK07a] Bell, Robert M. ; Koren, Yehuda: Improved Neighborhood-based Collaborative Filtering. In: *Proceedings of the KDD Cup 2007*, ACM Press, 2007

[BK07b] Bell, Robert M. ; Koren, Yehuda: Lessons from the Netflix prize challenge. In: *SIGKDD Explor. Newsl.* 9 (2007), Nr. 2, S. 75–79. – ISSN 1931–0145

[BL07] Bennett, J. ; Lanning, S.: The Netflix Prize. In: *KDD-Cup and Workshop at the 13th ACM SIGKDD International*

Conference on Knowledge Discovery and Data Mining. San Jose, California, 08 2007

[BLG98] Bollacker, Kurt D. ; Lawrence, Steve ; Giles, C. L.: CiteSeer: an autonous Web agent for automatic retrieval and identification of interesting publications. In: *AGENTS '98: Proceedings of the second international conference on Autonomous agents*. New York, NY, USA : ACM Press, 1998. – ISBN 0–89791–983–1, S. 116–123

[BNW06] Byrd, Richard H. ; Nocedaly, Jorge ; Waltzy, Richard A.: Knitro: An Integrated Package for Nonlinear Optimization / Ziena Optimization, Inc. 2006. – Forschungsbericht

[Boi05] Boizet, Christophe: KartOO – The new way of searching / KartOO Techologies. 2005 (09). – Documentation. http://www.kartoo.net/e/eng/doc/newsletter0509.doc [06.08.2008]

[BP98] Brin, Sergey ; Page, Lawrence: The anatomy of a large-scale hypertextual Web search engine. In: *Computer Networks and ISDN Systems* 30 (1998), Nr. 1–7, S. 107–117

[BPR01a] Bellifemine, Fabio ; Poggi, Agostino ; Rimassa, Giovanni: Developing Multi-agent Systems with JADE. In: *ATAL '00: Proceedings of the 7th International Workshop on Intelligent Agents VII. Agent Theories Architectures and Languages*. London, UK : Springer-Verlag, 2001. – ISBN 3–540–42422–9, S. 89–103

[BPR01b] Bellifemine, Fabio ; Poggi, Agostino ; Rimassa, Giovanni: JADE: A FIPA2000 compliant Agent Development Environment. In: *AGENTS '01: Proceedings of the fifth international conference on Autonomous agents*. New York, NY, USA : ACM, 2001. – ISBN 1–58113–326–X, S. 216–217

[BR88] Bollmann, P. ; Raghavan, V. V.: A utility-theoretic analysis of expected search length. In: *SIGIR '88: Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 1988. – ISBN 2–7061–0309–4, S. 245–256

[BRHL99] Busetta, Paolo ; Ronnquist, Ralph ; Hodgson, Andrew ; Lucas, Andrew: Jack intelligent agents - components for intelligent agents in java / AgentLink. 1999. – Forschungsbericht

[BS97] Balabanović, Marko ; Shoham, Yoav: Fab: content-based, collaborative recommendation. In: *Commun. ACM* 40 (1997), Nr. 3, S. 66–72. – ISSN 0001–0782

[BSR+05] Burges, Chris ; Shaked, Tal ; Renshaw, Erin ; Lazier, Ari ; Deeds, Matt ; Hamilton, Nicole ; Hullender, Greg: Learning to rank using gradient descent. In: *ICML '05: Proceedings of the 22nd international conference on Machine learning*. New York, NY, USA : ACM, 2005. – ISBN 1–59593–180–5, S. 89–96

[Bun05] Bundesamt für Sicherheit in der Informationstechnik. *Deutsche IT-Sicherheitszertifikate*. 2005

[BV00] Buckley, Chris ; Voorhees, Ellen M.: Evaluating evaluation measure stability. In: *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2000. – ISBN 1–58113–226–3, S. 33–40

[BV04] Buckley, Chris ; Voorhees, Ellen M.: Retrieval evaluation with incomplete information. In: *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM, 2004. – ISBN 1–58113–881–4, S. 25–32

[BYRN99] Baeza-Yates, R. ; Ribeiro-Neto, B: *Modern Information Retrieval*. Addison Wesley, 1999. – 73–96 S

[CA07] Cissée, Richard ; Albayrak, Sahin: An agent-based approach for privacy-preserving recommender systems. In: *AAMAS '07: Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*. New York, NY, USA : ACM, 2007. – ISBN 978–81–904262–7–5, S. 1–8

[CCA89] Crouch, D. B. ; Crouch, C. J. ; Andreas, G.: The use of cluster hierarchies in hypertext information retrieval. In: *HYPERTEXT '89: Proceedings of the second annual ACM conference on Hypertext*. New York, NY, USA : ACM Press, 1989. – ISBN 0–89791–339–6, S. 225–237

[CDHW06] Chaudhuri, Surajit ; Das, Gautam ; Hristidis, Vagelis ; Weikum, Gerhard: Probabilistic information retrieval approach for ranking of database query results. In: *ACM Trans. Database Syst.* 31 (2006), Nr. 3, S. 1134–1168. – ISSN 0362–5915

[CFGM01] Chowdhury, Abdur ; Frieder, Ophir ; Grossman, David ; McCabe, Catherine: Analyses of multiple-evidence combinations for retrieval strategies. In: *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval.* New York, NY, USA : ACM Press, 2001. – ISBN 1–58113–331–6, S. 394–395

[CGMP96] Chang, Chen-Chuan K. ; García-Molina, Héctor ; Paepcke, Andreas: Boolean Query Mapping Across Heterogeneous Information Sources. In: *IEEE Transactions on Knowledge and Data Engineering* 8 (1996), Nr. 4, S. 515–521

[CHT99] Craswell, Nick ; Hawking, David ; Thistlewaite, Paul B.: Merging Results From Isolated Search Engines. In: *Australasian Database Conference*, 1999, S. 189–200

[CLRC98] Crestani, Fabio ; Lalmas, Mounia ; Rijsbergen, Cornelis J. V. ; Campbell, Iain: "Is this document relevant?... probably": a survey of probabilistic models in information retrieval. In: *ACM Comput. Surv.* 30 (1998), Nr. 4, S. 528–552. – ISSN 0360–0300

[CMK66] Cleverdon, C. W. ; Mills, J. ; Keen, E. M.: *Factors Determining the Performance of Indexing Systems. Vol. 1. Design, Parts 1 and 2; Vol. 2, Test Results (Cleverdon and Keen).* Aslib Cranfield Institute of Technology, 1966

[Coo68] Cooper, W. S.: Expected Search Length: A Single Measure of Retrieval Effectiveness Based on the Weak Ordering Action of Retrieval Systems. In: *American Documentation* 19 (1968), S. 30–41

[Coo91] Cooper, William S.: Some inconsistencies and misnomers in probabilistic information retrieval. In: *SIGIR '91: Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval.* New York, NY, USA : ACM Press, 1991. – ISBN 0–89791–448–1, S. 57–61

[CQL+07] Cao, Zhe ; Qin, Tao ; Liu, Tie-Yan ; Tsai, Ming-Feng ; Li, Hang: Learning to rank: from pairwise approach to listwise approach. In: *ICML '07: Proceedings of the 24th international conference on Machine learning*. New York, NY, USA : ACM, 2007. – ISBN 978–1–59593–793–3, S. 129–136

[Cro02] Kap. 1 In: Croft, W. B.: *Combining Approaches to Information Retrieval Advances in Information Retrieval*. Springer Netherlands, 2002, S. 1–36

[CS98] Chen, Liren ; Sycara, Katia: WebMate: A Personal Agent for Browsing and Searching. In: Sycara, Katia P. (Hrsg.) ; Wooldridge, Michael (Hrsg.): *Proceedings of the 2nd International Conference on Autonomous Agents (Agents'98)*, ACM Press, 1998. – ISBN 0–89791–983–1, S. 770–777

[CS02] Crammer, Koby ; Singer, Yoram: A new family of online algorithms for category ranking. In: *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2002. – ISBN 1–58113–561–0, S. 151–158

[CS07] Coyle, Maurice ; Smyth, Barry: Supporting intelligent Web search. In: *ACM Trans. Inter. Tech.* 7 (2007), Nr. 4, S. 20. – ISSN 1533–5399

[CSWZ98] Chang, Wendy ; Sheikholeslami, Gholamhosein ; Wang, Jia ; Zhang, Aidong: Data Resource Selection in Distributed Visual Information Systems. In: *IEEE Transactions on Knowledge and Data Engineering* 10 (1998), Nr. 6, S. 926–946. – ISSN 1041–4347

[DDH72] Dahl, O. J. (Hrsg.) ; Dijkstra, E. W. (Hrsg.) ; Hoare, C. A. R. (Hrsg.): *Structured programming*. London, UK, UK : Academic Press Ltd., 1972. – ISBN 0–12–200550–3

[DLR77] Dempster, A. P. ; Laird, N. M. ; Rubin, D. B.: Maximum likelihood from incomplete data via the EM algorithm. In: *Journal of the Royal Statistical Society* Series B (1977), S. 1–38

[Edl07] Edler, Ronny: *Koordination und Benchmarking von Filterstrategien*, Technische Universität Berlin, Diplomarbeit, 2007

[EV03] Eirinaki, Magdalini ; Vazirgiannis, Michalis: Web mining for web personalization. In: *ACM Trans. Interet Technol.* 3 (2003), Nr. 1, S. 1–27. – ISSN 1533–5399

[Far81] Farradane, J.: The basis for a theory of information retrieval. In: *SIGIR '81: Proceedings of the 4th annual international ACM SIGIR conference on Information storage and retrieval.* New York, NY, USA : ACM, 1981. – ISBN 0–89791–052–4, S. 1–3

[FBK+01] Fricke, Stefan ; Bsufka, Karsten ; Keiser, Jan ; Schmidt, Torge ; Sesseler, Ralf ; Albayrak, Sahin: Agent-based telematic services and telecom applications. In: *j-CACM* 44 (2001), April, Nr. 4, S. 43–48. – ISSN 0001–0782

[FG96] Franklin, S. ; Graesser, A.: Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents. In: *Intelligent Agents III. Agent Theories, Architectures and Languages (ATAL'96)* Bd. 1193. Berlin, Germany : Springer-Verlag, 1996

[FGL02] Fonseca, Steven P. ; Griss, Martin L. ; Letsinger, Reed: Agent behavior architectures a MAS framework comparison. In: *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems.* New York, NY, USA : ACM, 2002. – ISBN 1–58113–480–0, S. 86–87

[FISS03] Freund, Yoav ; Iyer, Raj ; Schapire, Robert E. ; Singer, Yoram: An efficient boosting algorithm for combining preferences. In: *J. Mach. Learn. Res.* 4 (2003), S. 933–969. – ISSN 1533–7928

[FKPT02] Fahrmeir, Ludwig ; Künstler, Rita ; Pigeot, Iris ; Tutz, Gerhard: *Statistik.* Springer, Aug 2002. – ISBN 3540678263

[Fli00] Flick, Uwe: *Qualitative Forschung : Theorie, Methoden, Anwendung in Psychologie und Sozialwissenschaften.* Rowohlt-Taschenbuch, 2000

[FP94] Fuhr, Norbert ; Pfeifer, Ulrich: Probabilistic information retrieval as a combination of abstraction, inductive learning, and probabilistic assumptions. In: *ACM Trans. Inf. Syst.* 12 (1994), Nr. 1, S. 92–115. – ISSN 1046–8188

[Fre99] Freund: An Adaptive Version of the Boost by Majority Algorithm. In: *COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers*, 1999

[FS94] Fox, E.A. ; Shaw, J.A.: Combination of multiple searches. In: *Proceedings of the 2nd Text REtrieval Conference (TREC-2)*, National Institute of Standards and Technology Special Publication, 500-215, 1994, S. 243–252

[FS95] Freund, Yoav ; Schapire, Robert E.: A decision-theoretic generalization of on-line learning and an application to boosting. In: *EuroCOLT '95: Proceedings of the Second European Conference on Computational Learning Theory*. London, UK : Springer-Verlag, 1995. – ISBN 3–540–59119–2, S. 23–37

[FS96] Freund, Yoav ; Schapire, Robert E.: Experiments with a New Boosting Algorithm. In: *International Conference on Machine Learning*, 1996, S. 148–156

[Fuh99] Fuhr, Norbert: A decision-theoretic approach to database selection in networked IR. In: *ACM Trans. Inf. Syst.* 17 (1999), Nr. 3, S. 229–249. – ISSN 1046–8188

[FV07] Farah, Mohamed ; Vanderpooten, Daniel: An outranking approach for rank aggregation in information retrieval. In: *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM, 2007. – ISBN 978–1–59593–597–7, S. 591–598

[FVC06] Fernandez, Miriam ; Vallet, David ; Castells, Pablo: Using historical data to enhance rank aggregation. In: *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2006. – ISBN 1–59593–369–7, S. 643–644

[FWRZ06] Fan, Weiguo ; Wallace, Linda ; Rich, Stephanie ; Zhang, Zhongju: Tapping the power of text mining. In: *Commun. ACM* 49 (2006), Nr. 9, S. 76–82. – ISSN 0001–0782

[GDNG99] Gupta, Dhruv ; Digiovanni, Mark ; Narita, Hiro ; Goldberg, Ken: Jester 2.0: Evaluation of a New Linear–Time Collaborative Filtering Algorithm. In: *In Proceedings of the 22nd*

Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (1999), August 15–19, S. 291–292

[Goo07] Google Inc.: *Corporate Information: Google Milestones*. Web, Google Inc., 1600 Amphitheatre Parkway, Mountain View, CA 94043, USA. 2007. – `http://www.google.com/corporate/history.html` [06.08.2008]

[GS03] Giunchiglia, Fausto ; Shvaiko, Pavel: Semantic matching. In: *Knowl. Eng. Rev.* 18 (2003), Nr. 3, S. 265–280. – ISSN 0269–8889

[GT05] Günther, Hans-Otto ; Tempelmeier, Horst: *Produktion und Logistik*. 6. Auflage. Springer, Berlin, 2005 (ISBN 354023246X)

[Har75] Hartigan, John A.: *Clustering Algorithms*. New York, NY, USA : John Wiley & Sons, Inc., 1975. – ISBN 047135645X

[Hau04] Haupt, Tobias C.: *Persönlichkeitstyp und Stresserleben*, Ludwig-Maximilians-Universität München, Diss., 2004

[HB97] Hofmann, Thomas ; Buhmann, Joachim M.: Pairwise Data Clustering by Deterministic Annealing. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 19 (1997), Nr. 1, S. 1–14. – ISSN 0162–8828

[HBG+98] Han, Eui-Hong ; Boley, Daniel ; Gini, Maria ; Gross, Robert ; Hastings, Kyle ; Karypis, George ; Kumar, Vipin ; Mobasher, Bamshad ; Moore, Jerome: WebACE: A Web Agent for Document Categorization and Exploration. In: Sycara, Katia P. (Hrsg.) ; Wooldridge, Michael (Hrsg.): *Proceedings of the 2nd International Conference on Autonomous Agents (Agents'98)*. New York : ACM Press, 9–13, 1998. – ISBN 0–89791–983–1, S. 408–415

[HBLH94] Hersh, William ; Buckley, Chris ; Leone, T. J. ; Hickam, David: OHSUMED: an interactive retrieval evaluation and new large test collection for research. In: *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : Springer-Verlag New York, Inc., 1994. – ISBN 0–387–19889–X, S. 192–201

[HBV01] Halkidi, Maria ; Batistakis, Yannis ; Vazirgiannis, Michalis: On clustering validation techniques. In: *Journal of Intelligent Information Systems* 17 (2001), S. 107–145

[Hec06] Heck, Mike: Vivisimo Races Ahead of the Search Pack. In: *Info World* (2006), 09, Nr. 37

[Her97] Herrmann, Jürgen: *Maschinelles Lernen und Wissensbasierte Systeme*. Springer, 1997

[HGO00] Herbrich, Ralf ; Graepel, Thore ; Obermayer, Klaus ; Smola (Hrsg.) ; Bartlett (Hrsg.) ; Schoelkopf (Hrsg.) ; Schuurmans (Hrsg.): *Large margin rank boundaries for ordinal regression*. MIT Press, Cambridge, MA, 2000

[HHOW05] Hebrard, Emmanuel ; Hnich, Brahim ; O'Sullivan, Barry ; Walsh, Toby: Finding Diverse and Similar Solutions in Constraint Programming. In: *AAAI*, 2005, S. 372–377

[HJ07] Hahne, Michael ; Jung, Corinna: Auswertung der Voruntersuchung des PIA-Systems / Zentrum Technik und Gesellschaft, TU Berlin. 2007. – Forschungsbericht

[HJJ+07] Hahne, Michael ; Jung, Corinna ; Jérôme, Jérôme ; Lommatzsch, Andreas ; Paus, André: Ein gradualisiertes Communitymodell zur Bildung wissenschaftlicher Gemeinschaften und seine Anwendung auf die Entwicklung von Social Software. In: Gronau, Norbert (Hrsg.) ; Müller, Claudia (Hrsg.): *Bildung von sozialen Netzwerken in Anwendungen der Social Software*. GITO Verlag – Expertenwissen für die industrielle Praxis, 2007

[HKGM08] Heymann, Paul ; Koutrika, Georgia ; Garcia-Molina, Hector: Can social bookmarking improve web search? In: *WSDM '08: Proceedings of the international conference on Web search and web data mining*. New York, NY, USA : ACM, 2008. – ISBN 978–1–59593–927–9, S. 195–206

[HKHA06] Hirsch, Benjamin ; Konnerth, Thomas ; Hessler, Axel ; Albayrak, Sahin: *A Serviceware Framework for Designing Ambient Services*. Springer, 2006. – 124–136 S. – ISBN 978–2–287–47469–9 (Print) 978–2–287–47610–5 (Online)

[HLWZ03] Helsinger, Aaron ; Lazarus, Richard ; Wright, William ; Zinky, John: Tools and techniques for performance measurement of large distributed multiagent systems. In: *AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems*. New York, NY, USA : ACM Press, 2003. – ISBN 1–58113–683–8, S. 843–850

[HMK03] Hansen, N. ; Muller, S.D. ; Koumoutsakos, P.: Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). In: *Evolutionary Computation* 11 (2003), Nr. 1, S. 1–18

[HO96] Hansen, N. ; Ostermeier, A.: Adapting arbitrary normal mutation distributions in evolution strategies: the covariance matrix adaptation. In: *Evolutionary Computation, Proceedings of IEEE International Conference on Evolutionary Computation (ICEC'96)*, IEEE Press, NY, 1996. – DOI: 10.1109/ICEC.1996.542381, S. 312–317

[HO06] Hochreiter, Sepp ; Obermayer, Klaus: Support Vector Machines for Dyadic Data. In: *Neural Computation* 18 (2006), S. 1472–1510

[Hos05] Hosanagar, Kartik: A utility theoretic approach to determining optimal wait times in distributed information retrieval. In: *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2005. – ISBN 1–59593–034–5, S. 91–97

[HPS96] Hull, David A. ; Pedersen, Jan O. ; Schütze, Hinrich: Method combination for document filtering. In: *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 1996. – ISBN 0–89791–792–8, S. 279–287

[Hul93] Hull, David: Using statistical testing in the evaluation of retrieval experiments. In: *SIGIR '93: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 1993. – ISBN 0–89791–605–0, S. 329–338

[Iwa00] Iwayama, Makoto: Relevance feedback with a small number of relevance judgements: incremental relevance feedback vs. document clustering. In: *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2000. – ISBN 1–58113–226–3, S. 10–16

[JB05] Jo Bager, c't: *Google vergrößert den Suchindex*. Web. 09 2005. – `http://www.heise.de/newsticker/Google-vergroessert-den-Suchindex--/meldung/64319` [06.08.2008]

[Jen00] Jennings, Nicholas R.: On agent-based software engineering. In: *Artificial Intelligence* 177 (2000), Nr. 2, S. 277–296

[JFM97] Joachims, Thorsten ; Freitag, Dayne ; Mitchell, Tom M.: Web Watcher: A Tour Guide for the World Wide Web. In: *Proceedings of the IJCAI 1997*, 1997, S. 770–777

[JK02] Järvelin, K. ; Kekäläinen, J.: Cumulated Gain-based Evaluation of IR Techniques. In: *ACM Trans. Inf. Syst,* 20 (2002), S. 422–446

[JMF99] Jain, A. K. ; Murty, M. N. ; Flynn, P. J.: Data clustering: a review. In: *ACM Comput. Surv.* 31 (1999), Nr. 3, S. 264–323. – ISSN 0360–0300

[Joa97] Joachims, Thorsten: A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization. In: Fisher, Douglas H. (Hrsg.): *Proceedings of ICML-97, 14th International Conference on Machine Learning*. Nashville, US : Morgan Kaufmann Publishers, San Francisco, US, 1997, S. 143–151

[Joa02] Joachims, Thorsten: Optimizing search engines using clickthrough data. In: *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA : ACM, 2002. – ISBN 1–58113–567–X, S. 133–142

[Jon98] Jones, Steve: Dynamic Query Result Previews for a Digital Library. In: *ACM DL*, 1998, S. 291–292

[Ken38] Kendall, M. G.: A new measure of rank correlation. In: *Biometrika* 30 (1938), S. 81–93

[Kle99] KLEINBERG, Jon M.: Authoritative sources in a hyperlinked environment. In: *Journal of the ACM* 46 (1999), Nr. 5, S. 604–632

[KLR+04] KUMMAMURU, Krishna ; LOTLIKAR, Rohit ; ROY, Shourya ; SINGAL, Karan ; KRISHNAPURAM, Raghu: A hierarchical monothetic document clustering algorithm for summarization and browsing search results. In: *WWW '04: Proceedings of the 13th international conference on World Wide Web*. New York, NY, USA : ACM, 2004. – ISBN 1–58113–844–X, S. 658–665

[Knu73] KNUTH, Donald: *The Art Of Computer Programming, vol. 3: Sorting And Searching*. Addison-Wesley, 1973. – S. 391 – 392

[KPK01] KOLCZ, Aleksander ; PRABAKARMURTHI, Vidya ; KALITA, Jugal: Summarization as feature selection for text categorization. In: *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*. New York, NY, USA : ACM, 2001. – ISBN 1–58113–436–3, S. 365–370

[KR04] KAMPS, Jaap ; DE RIJKE, Maarten: The effectiveness of combining information retrieval strategies for European languages. In: *SAC '04: Proceedings of the 2004 ACM symposium on Applied computing*. New York, NY, USA : ACM Press, 2004. – ISBN 1–58113–812–1, S. 1073–1077

[Kun06] KUNEGIS, Jérôme: *Using Integer Linear Programming for Search Results Optimization*, TU Berlin, Diplomarbeit, 2006

[Kwo95] KWOK, K. L.: A network approach to probabilistic information retrieval. In: *ACM Trans. Inf. Syst.* 13 (1995), Nr. 3, S. 324–353. – ISSN 1046–8188

[KWSD00] KUNCHEVA, L. I. ; WHITAKER, C. J. ; SHIPP, C.A. ; DUIN, R. P. W.: Is independence good for combining classifiers? In: *Proceedings of the 15th International Conference on Pattern Recognition* Bd. 2, 2000, S. 168–171

[KZ06] KLEINER, A. ; ZIPARO, V.A.: RoboCupRescue - Simulation League Team RescueRobots Freiburg (Germany). In: *RoboCup 2006 (CDROM Proceedings), Team Description Paper, Rescue Simulation League*. Bremen, Germany, 2006

[LA07] Lommatzsch, Andreas ; Albayrak, Sahin: Using multi-agent systems for learning optimal policies for complex problems. In: *ACM-SE 45: Proceedings of the 45th annual southeast regional conference*. New York, NY, USA : ACM, 2007. – ISBN 978–1–59593–629–5, S. 244–249

[LCC00] Larkey, Leah S. ; Connell, Margaret E. ; Callan, Jamie: Collection selection and results merging with topically organized U.S. patents and TREC data. In: *CIKM '00: Proceedings of the ninth international conference on Information and knowledge management*. New York, NY, USA : ACM Press, 2000. – ISBN 1–58113–320–0, S. 282–289

[LDV99] Lieberman, Henry ; Dyke, Neil W. V. ; Vivacqua, Adrian S.: Let's browse: a collaborative Web browsing agent. In: *Proceedings of the 1999 International Conference on Intelligent User Interfaces (IUI'99)*, ACM Press, 1999, S. 65–68

[Lee95] Lee, Joon H.: Combining multiple evidence from different properties of weighting schemes. In: *SIGIR '95: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 1995. – ISBN 0–89791–714–6, S. 180–188

[Lee97] Lee, Joon H.: Analyses of multiple evidence combination. In: *SIGIR '97: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 1997. – ISBN 0–89791–836–3, S. 267–276

[Leu01] Leuski, Anton: Evaluating document clustering for interactive information retrieval. In: *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*. New York, NY, USA : ACM Press, 2001. – ISBN 1–58113–436–3, S. 33–40

[Lev66] Levenshtein, Vladimir I.: Binary codes capable of correcting deletions, insertions, and reversals. In: *Soviet Physics Doklady* 10 (1966), Nr. 8, S. 707–710

[Lew92] Lewis, David D.: An evaluation of phrasal and clustered representations on a text categorization task. In: *SIGIR '92:*

Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval. New York, NY, USA : ACM Press, 1992. – ISBN 0–89791–523–2, S. 37–50

[LG94] Lewis, David D. ; Gale, William A.: A sequential algorithm for training text classifiers. In: *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : Springer-Verlag New York, Inc., 1994. – ISBN 0–387–19889–X, S. 3–12

[Lie95] Lieberman, Henry: Letizia: An Agent That Assists Web Browsing. In: Mellish, Chris S. (Hrsg.): *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)*. Montreal, Quebec, Canada : Morgan Kaufmann publishers Inc.: San Mateo, CA, USA, 1995. – ISBN 1–55860–363–8, S. 924–929

[LLC03] Losee, Robert M. ; Lewis Church, Jr.: Information Retrieval with Distributed Databases: Analytic Models of Performance. In: *IEEE Trans. on Parallel and Distributed Systems* 14 (2003), Nr. 12, S. 18 – 27

[LLS08] Lee, Wai-Lung ; Lommatzsch, Andreas ; Scheel, Christian: Feed Distillation Using AdaBoost and Topic Maps. In: Voorhees, E. M. (Hrsg.) ; Buckland, Lori P. (Hrsg.): *The Sixteenth Text REtrieval Conference (TREC 2007) Proceedings*, 2008

[LMK07] Lommatzsch, Andreas ; Mehlitz, Martin ; Kunegis, Jérôme: A Multi-Agent Framework for Personalized Information Filtering. In: *Proceedings of German e-Science 2007 (GES'07)*, 2007

[LRWW98] Lagarias, Jeffrey C. ; Reeds, James A. ; Wright, Margaret H. ; Wright, Paul E.: Convergence Properties of the Nelder–Mead Simplex Method in Low Dimensions. In: *SIAM J. on Optimization* 9 (1998), Nr. 1, S. 112–147. – ISSN 1052–6234

[LXQ+07] Liu, Tie-Yan ; Xu, Jun ; Qin, Tao ; Xiong, Wenying ; Li, Hang: LETOR: Benchmark Dataset for Research on Learning

to Rank for Information Retrieval. In: *LR4IR 2007, in conjunction with SIGIR 2007*, 2007

[MA01a] Montague, Mark ; Aslam, Javed A.: Metasearch consistency. In: *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2001. – ISBN 1–58113–331–6, S. 386–387

[MA01b] Montague, Mark ; Aslam, Javed A.: Relevance score normalization for metasearch. In: *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*. New York, NY, USA : ACM Press, 2001. – ISBN 1–58113–436–3, S. 427–433

[MA02] Montague, Mark ; Aslam, Javed A.: Condorcet fusion for improved retrieval. In: *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*. New York, NY, USA : ACM Press, 2002. – ISBN 1–58113–492–4, S. 538–548

[Mad96] Madani, O.: Efficient information gathering on the Internet. In: *FOCS '96: Proceedings of the 37th Annual Symposium on Foundations of Computer Science*. Washington, DC, USA : IEEE Computer Society, 1996, S. 234

[Mar52] Markowitz, Harry: Portfolio Selection. In: *The Journal of Finance* 7 (1952), März, Nr. 1, S. 77–91

[Mat08] Mattern, Friedemann (Hrsg.): *Wie arbeiten die Suchmaschinen von morgen?* Fraunhofer IRB Verlag, 2008 (acatech DISKUTIERT). – ISBN 978-3-8167-7526-3

[MCGF99] McCabe, M. C. ; Chowdhury, Abdur ; Grossman, David A. ; Frieder, Ophir: A unified environment for fusion of information retrieval approaches. In: *CIKM '99: Proceedings of the eighth international conference on Information and knowledge management*. New York, NY, USA : ACM Press, 1999. – ISBN 1–58113–146–1, S. 330–334

[MCM99] Martin, D. ; Cheyer, A. ; Moran, D.: The Open Agent Architecture: a framework for building distributed software systems. In: *Applied Artificial Intelligence* 13 (1999), Nr. 1/2, S. 91–128

[Men06] MENG, Xiannong: A Comparative Study of Performance Measures for Information Retrieval Systems. In: *Information Technology: New Generations, 2006. ITNG 2006. Third International Conference on*, IEEE Computer Society, 2006, S. 578–579

[Mil05] MILOSEVIC, Dragan: *Coordinating Filtering Strategies in Cooperative Agent Communities*, Technische Universität Berlin, Diss., 2005

[Mit97] MITCHEL, T. M.: *Machine Learning*. Boston, Massachusetts : McGraw-Hill, 1997. – ISBN 0070428077

[MKA07] MEHLITZ, Martin ; KUNEGIS, Jérôme ; ALBAYRAK, Sahin: Using Novel IR Measures to Learn Optimal Cluster Structures for Web Information Retrieval. In: *WI '07: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*. Washington, DC, USA : IEEE Computer Society, 2007. – ISBN 0–7695–3026–5, S. 312–316

[MKS02] MCGOWAN, J. P. ; KUSHMERICK, Nicholas ; SMYTH, Barry: Who Do You Want to Be Today? Web Personae for Personalised Information Access. In: *AH '02: Proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*. London, UK : Springer-Verlag, 2002. – ISBN 3–540–43737–1, S. 514–517

[MKSW99] MAKHOUL, John ; KUBALA, Francis ; SCHWARTZ, Richard ; WEISCHEDEL, Ralph: Performance measures for information extraction. In: *In Proceedings of DARPA Broadcast News Workshop*, 1999, S. 249–252

[MM03] MELVILLE, Prem ; MOONEY, Ray: Constructing diverse classifier ensembles using artificial training examples. In: *Proceeding of 18th Intl. Joint Conference on Artificial Intelligence*. Acapulco, Mexico, August 2003, S. 505–510

[MRF01] MANMATHA, R. ; RATH, T. ; FENG, F.: Modeling score distributions for combining the outputs of search engines. In: *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2001. – ISBN 1–58113–331–6, S. 267–275

[MU05] Mitzenmacher, Michael ; Upfal, Eli: *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. New York, NY, USA : Cambridge University Press, 2005. – ISBN 0521835402

[NDQ06] Nie, Lan ; Davison, Brian D. ; Qi, Xiaoguang: Topical link analysis for web search. In: *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM, 2006. – ISBN 1–59593–369–7, S. 91–98

[NZT07] Najork, Marc A. ; Zaragoza, Hugo ; Taylor, Michael J.: Hits on the web: how does it compare? In: *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM, 2007. – ISBN 978–1–59593–597–7, S. 471–478

[Pat05] Patrick, Paul: Impact of SOA on enterprise information architectures. In: *SIGMOD '05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data*. New York, NY, USA : ACM Press, 2005. – ISBN 1–59593–060–4, S. 844–848

[PBMW98] Page, Lawrence ; Brin, Sergey ; Motwani, Rajeev ; Winograd, Terry: The PageRank Citation Ranking: Bringing Order to the Web / Stanford Digital Library Technologies Project. 1998. – Forschungsbericht

[PH07] Papazoglou, Mike P. ; Heuvel, Willem-Jan: Service oriented architectures: approaches, technologies and research issues. In: *The VLDB Journal* 16 (2007), Nr. 3, S. 389–415. – ISSN 1066–8888

[PMB96] Pazzani, Michael J. ; Muramatsu, Jack ; Billsus, Daniel: Syskill Webert: Identifying Interesting Web Sites. In: *AAAI/IAAI, Vol. 1*, 1996, S. 54–61

[Pol06] Polikar, Robi: Ensemble based systems in decision making. In: *Circuits and Systems Magazine, IEEE* 6 (2006), 9, Nr. 3, S. 21 – 45. – 1531-636X

[Por97] PORTER, M. F.: *An algorithm for suffix stripping.* San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1997. – S. 313–316. – ISBN 1–55860–454–5

[QLL07] QIN, Tao ; LIU, Tie-Yan ; LI, Hang. *The TREC Datasets in LETOR.* Part of the TREC datasets in LETOR description. May 2007

[QLZ+05] QIN, Tao ; LIU, Tie-Yan ; ZHANG, Xu-Dong ; CHEN, Zheng ; MA, Wei-Ying: A study of relevance propagation for web search. In: *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval.* New York, NY, USA : ACM, 2005. – ISBN 1–59593–034–5, S. 408–415

[Qui93] QUINLAN, J. R.: *C4.5: programs for machine learning.* San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1993. – ISBN 1–55860–238–0

[RAS01] RASOLOFO, Yves ; ABBACI, Faïza ; SAVOY, Jacques: Approaches to collection selection and results merging for distributed information retrieval. In: *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management.* New York, NY, USA : ACM Press, 2001. – ISBN 1–58113–436–3, S. 191–198

[RBJ89a] RAGHAVAN, V. V. ; BOLLMANN, P. ; JUNG, G. S.: Retrieval system evaluation using recall and precision: problems and answers. In: *SIGIR '89: Proceedings of the 12th annual international ACM SIGIR conference on Research and development in information retrieval.* New York, NY, USA : ACM Press, 1989. – ISBN 0–89791–321–3, S. 59–68

[RBJ89b] RAGHAVAN, Vijay ; BOLLMANN, Peter ; JUNG, Gwang S.: A critical investigation of recall and precision as measures of retrieval system performance. In: *ACM Trans. Inf. Syst.* 7 (1989), Nr. 3, S. 205–229. – ISSN 1046–8188

[Rij79] VAN RIJSBERGEN, C. J.: *Information retrieval, Second Edition.* Newton, MA, USA : Butterworth-Heinemann, 1979. – ISBN 0408709294

[Rij83] VAN RIJSBERGEN, C. J.: Information retrieval: new directions: old solutions. In: *SIGIR '83: Proceedings of the 6th annual*

international ACM SIGIR conference on Research and development in information retrieval. New York, NY, USA : ACM Press, 1983. – ISBN 0–89791–107–5, S. 264–265

[Rij86] VAN RIJSBERGEN, C. J.: (invited paper) A new theoretical framework for information retrieval. In: *SIGIR '86: Proceedings of the 9th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 1986. – ISBN 0–89791–187–3, S. 194–200

[RIS+94] RESNICK, Paul ; IACOVOU, Neophytos ; SUCHAK, Mitesh ; BERGSTROM, Peter ; RIEDL, John: GroupLens: an open architecture for collaborative filtering of netnews. In: *CSCW '94: Proceedings of the 1994 ACM conference on Computer supported cooperative work*. New York, NY, USA : ACM, 1994. – ISBN 0–89791–689–1, S. 175–186

[Ris05] RISSE, Mathias: Why the count de Borda cannot beat the Marquis de Condorcet. In: *Social Choice and Welfare* 25 (2005), October, Nr. 1, S. 95–113

[RMB00] RUH, William A. ; MAGINNIS, Francis X. ; BROWN, William J.: *Enterprise Application Integration: A Wiley Tech Brief*. 1st edition. Wiley, October 13 2000. – ISBN 978-0471376415

[RMS91] RITTER, Helge ; MARTINEZ, Thomas ; SCHULTEN, Klaus: *Reihe künstliche Intelligenz*. Bd. 2., erw. Aufl.: *Neuronale Netze. Eine Einführung in die Neuroinformatik selbstorganisierender Netzwerke*. Bonn ; München ; Reading, Mass. : Addison Wesley, 1991

[RN95] RUSSEL, S. ; NORVIG, P.: *Artificial Intelligence, a Modern Approach*. Prentice Hall, 1995

[RPB06] RICHARDSON, Matthew ; PRAKASH, Amit ; BRILL, Eric: Beyond PageRank: Machine learning for static ranking. In: *WWW '06: Proceedings of the 15th international conference on World Wide Web*. New York, NY, USA : ACM Press, 2006. – ISBN 1–59593–323–9, S. 707–715

[RPP05] ROTH, Volker ; PINSDORF, Ulrich ; PETERS, Jan: A distributed content-based search engine based on mobile code. In:

SAC '05: Proceedings of the 2005 ACM symposium on Applied computing. New York, NY, USA : ACM, 2005. – ISBN 1–58113–964–0, S. 66–73

[RS03] Renda, M. E. ; Straccia, Umberto: Web metasearch: rank vs. score based rank aggregation methods. In: *SAC '03: Proceedings of the 2003 ACM symposium on Applied computing*. New York, NY, USA : ACM, 2003. – ISBN 1–58113–624–2, S. 841–846

[RWB98] Robertson, S. E. ; Walker, S. ; Beaulieu, M.: Okapi at TREC-7: Automatic Ad Hoc, Filtering, VLC and Interactive track. In: *TREC*, 1998, S. 199–210

[Sal86] Salton, Gerard: Another look at automatic text-retrieval systems. In: *Commun. ACM* 29 (1986), Nr. 7, S. 648–656. – ISSN 0001–0782

[SB98] Sutton, R. S. ; Barto, A. G.: *Reinforcement Learning – An Introduction*. MIT Press, 1998

[SBY82] Salton, G. ; Buckley, C. ; Yu, C. T.: An evaluation of term dependence models in information retrieval. In: *SIGIR '82: Proceedings of the 5th annual ACM conference on Research and development in information retrieval*. New York, NY, USA : Springer-Verlag New York, Inc., 1982. – ISBN 0–387–11978–7, S. 151–173

[SC02] Si, Luo ; Callan, Jamie: Using sampled data and regression to merge search engine results. In: *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2002. – ISBN 1–58113–561–0, S. 19–26

[Sch02] Schmidt, Torge: *ASITA: Advanced Security Infrastructure for Multi-Agent-Applications in the Telematic Area*, TU Berlin, Dissertation, 2002

[Sch03] Schulze, Markus: A New Monotonic and Clone-Independent Single-Winner Election Method. In: *Voting Matters* 11 (2003), S. 9–19

[SDG+00] STEPHAN, V. ; DEBES, K. ; GROSS, H.-M. ; WINTRICH, F. ; WINTRICH, H.: A Reinforcement Learning based Neural Multi-Agent-System for Control of a Combustion Process. IV (2000), S. 217–222

[Ses01] SESSELER, Ralf: Building agents for service provisioning out of components. In: MÜLLER, Jörg P. (Hrsg.) ; ANDRE, Elisabeth (Hrsg.) ; SEN, Sandip (Hrsg.) ; FRASSON, Claude (Hrsg.): *Proceedings of the Fifth International Conference on Autonomous Agents*. Montreal, Canada : ACM Press, Mai 2001, S. 218–219

[Ses02] SESSELER, R.: *Eine modulare Architektur für dienstbasierte Interaktionen zwischen Agenten*, Technische Universität Berlin, Diss., 2002

[SFW83] SALTON, Gerard ; FOX, Edward A. ; WU, Harry: Extended Boolean information retrieval. In: *Commun. ACM* 26 (1983), Nr. 11, S. 1022–1036. – ISSN 0001–0782

[SGMB08] SHEPITSEN, Andriy ; GEMMELL, Jonathan ; MOBASHER, Bamshad ; BURKE, Robin: Personalized recommendation in social tagging systems using hierarchical clustering. In: *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*. New York, NY, USA : ACM, 2008. – ISBN 978–1–60558–093–7, S. 259–266

[Sin05] SINGHAL, Amit: Challenges in running a commercial search engine. In: *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 2005. – ISBN 1–59593–034–5, S. 432–432

[Six06] SIXTUS, Mario: *Google: Marktmacht*. c't 10/2006, S. 162. 10 2006. – `http://www.heise.de/ct/06/10/162/` [06.08.2008]

[SJCO02] SI, Luo ; JIN, Rong ; CALLAN, Jamie ; OGILVIE, Paul: A language modeling framework for resource selection and results merging. In: *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*. New York, NY, USA : ACM Press, 2002. – ISBN 1–58113–492–4, S. 391–397

[SKR02] SCHAFER, J. B. ; KONSTAN, Joseph A. ; RIEDL, John: Meta-recommendation systems: user-controlled integration of diverse

recommendations. In: *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*. New York, NY, USA : ACM, 2002. – ISBN 1–58113–492–4, S. 43–51

[SL65] Salton, G. ; Lesk, M. E.: The SMART automatic document retrieval systems - an illustration. In: *Commun. ACM* 8 (1965), Nr. 6, S. 391–398. – ISSN 0001–0782

[SLA08] Scheel, Christian ; Lommatzsch, Andreas ; Albayrak, Sahin: Extended Performance Measures in Information Retrieval for Multiple Levels of Relevance / DAI-Labor, TU Berlin. 2008. – Forschungsbericht

[SM01] Smyth, Barry ; McClave, Paul: Similarity vs. Diversity. In: *ICCBR '01: Proceedings of the 4th International Conference on Case-Based Reasoning*. London, UK : Springer-Verlag, 2001. – ISBN 3–540–42358–3, S. 347–361

[SMRM04] Smyth, Barry ; McGinty, Lorraine ; Reilly, James ; McCarthy, Kevin: Compound Critiques for Conversational Recommender Systems. In: *WI '04: Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*. Washington, DC, USA : IEEE Computer Society, 2004. – ISBN 0–7695–2100–2, S. 145–151

[SNL+07] Scheel, Christian ; Neubauer, Nicolas ; Lommatzsch, Andreas ; Obermayer, Klaus ; Albayrak, Sahin: Efficient Query Delegation by Detecting Redundant Retrieval Strategies. In: *Proceedings of SIGIR 2007 Workshop: Learning to Rank for Information Retrieval*, 2007

[Spe04] Spearman, C.: The proof and measurement of association between two things. In: *Am. J. Psychol* 15 (1904), S. 72–101

[SSS98] Schapire, Robert E. ; Singer, Yoram ; Singhal, Amit: Boosting and Rocchio applied to text filtering. In: *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM Press, 1998. – ISBN 1–58113–015–5, S. 215–223

[Sto05] Stojanovic, Nenad: On the role of a user's knowledge gap in an information retrieval process. In: *K-CAP '05: Proceedings*

of the 3rd international conference on Knowledge capture. New York, NY, USA : ACM, 2005. – ISBN 1–59593–163–5, S. 83–90

[Str06] STREIBEL, Olga: *Klassifikation und Erstellung von benutzerspezifischen Domänen mit Methoden und Algorithmen des maschinellen Lernens*, Technische Universität Berlin, Diplomarbeit, 2006

[Sul98] SULLIVAN, Danny: *GoTo Going Strong*. Web. Jul 1998. – `http://www.goto.com/` [06.08.2008]

[SW81] SALTON, Gerard ; WU, Harry: A term weighting model based on utility theory. In: *SIGIR '80: Proceedings of the 3rd annual ACM conference on Research and development in information retrieval*. Kent, UK : Butterworth & Co., 1981. – ISBN 0–408–10775–8, S. 9–22

[Swe69] SWETS, J. A.: Effectiveness of Information Retrieval Methods. In: *American Documentation* 20 (1969), Nr. 1, S. 72–89

[SWY75] SALTON, G. ; WONG, A. ; YANG, C. S.: A vector space model for automatic indexing. In: *Commun. ACM* 18 (1975), Nr. 11, S. 613–620. – ISSN 0001–0782

[Syc98] SYCARA, Katia P.: Multiagent Systems. In: *The AI magazine* 19 (1998), Nr. 2, S. 79–92. – ISSN 0738–4602

[SZ03] SHAKERY, Azadeh ; ZHAI, ChengXiang: Relevance Propagation for Topic Distillation, UIUC TREC 2003 Web Track Experiments. In: *The Twelfth Text REtrieval Conference (TREC 2003)*, 2003, S. 673–677

[TCH+06] TANG, Thanh T. ; CRASWELL, Nick ; HAWKING, David ; GRIFFITHS, Kathy ; CHRISTENSEN, Helen: Quality and relevance of domain-specific search: A case study in mental health. In: *Inf. Retr.* 9 (2006), Nr. 2, S. 207–225. – ISSN 1386–4564

[Tik63] TIKHONOV, A.N.: Solution of incorrectly formulated problems and the regularization method. In: *Soviet Math. Dokl. 4* (1963), S. 1035–1038. – English translation of Dokl. Akad. Nauk SSSR 151 (1963) 501-504

[TIS+07] Tweedale, J. ; Ichalkaranje, N. ; Sioutis, C. ; Jarvis, B. ; Consoli, A. ; Phillips-Wren, G.: Innovations in multi-agent systems. In: *J. Netw. Comput. Appl.* 30 (2007), Nr. 3, S. 1089–1115. – ISSN 1084–8045

[TLQ+07] Tsai, Ming-Feng ; Liu, Tie-Yan ; Qin, Tao ; Chen, Hsin-Hsi ; Ma, Wei-Ying: FRank: a ranking method with fidelity loss. In: *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA : ACM, 2007. – ISBN 978–1–59593–597–7, S. 383–390

[TP07] Tiemann, Marco ; Pauws, Steffen: Towards ensemble learning for hybrid music recommendation. In: *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*. New York, NY, USA : ACM, 2007. – ISBN 978–1–59593–730–8, S. 177–178

[TRE04] TREC@nist.gov: *Text Retrieval Conference (TREC) - Overview*. http://trec.nist.gov/overview.html: Information Accesss Division, Information Technology Laboratory, National Institute of Standards and Technology, U.S. Department of Commerce, Jan 2004

[TS99] Tabarrok, A. ; Spector, L.: Would the Borda Count have avoided the civil war? In: *Journal of Theoretical Politics* (1999), Nr. 11, S. 261–288

[TVR02] Tombros, Anastasios ; Villa, Robert ; Rijsbergen, C. J. V.: The effectiveness of query-specific hierarchic clustering in information retrieval. In: *Inf. Process. Manage.* 38 (2002), Nr. 4, S. 559–582. – ISSN 0306–4573

[Val84] Valiant, L. G.: A theory of the learnable. In: *Commun. ACM* 27 (1984), Nr. 11, S. 1134–1142. – ISSN 0001–0782

[VGJL94] Voorhees, Ellen M. ; Gupta, Narendra K. ; Johnson-Laird, Ben: The Collection Fusion Problem. In: Harman, D. K. (Hrsg.): *The Third Text REtrieval Conference (TREC-3)*, National Institute of Standards and Technology, 1994, S. 500–225

[VGJL95] Voorhees, Ellen M. ; Gupta, Narendra K. ; Johnson-Laird, Ben: Learning collection fusion strategies. In: *SIGIR*

'95: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval. New York, NY, USA : ACM Press, 1995. – ISBN 0–89791–714–6, S. 172–179

[Vor04] VORHEES, Ellen: NIST Special Publication: SP 500-255. In: *The Twelfth Text Retrieval Conference (TREC 2003)* NIST, 2004. – `http://trec.nist.gov/pubs/trec12/t12_proceedings.html`, S. 841–845

[WAB+07] WETZKER, Robert ; ALPCAN, Tansu ; BAUCKHAGE, Christian ; UMBRATH, Winfried ; ALBAYRAK, Sahin: An unsupervised hierarchical method for automated document categorization. In: *Proceedings of the IEEE/WIC/ACM Web Intelligence 2007*, IEEE Computer Society Press, 2007

[WC03] WU, Shengli ; CRESTANI, Fabio: Distributed information retrieval: A multi-objective resource selection approach. In: *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* 11 (2003), Nr. Supplement, S. 83–99. – ISSN 0218–4885

[WC04] WU, Shengli ; CRESTANI, Fabio: Shadow document methods of resutls merging. In: *SAC '04: Proceedings of the 2004 ACM symposium on Applied computing*. New York, NY, USA : ACM Press, 2004. – ISBN 1–58113–812–1, S. 1067–1072

[web08] WEBHITS INTERNET DESIGN GMBH: *Web-Barometer*. Web. 03 2008. – `http://www.webhits.de/deutsch/webstats.html#engines` [06.08.2008]

[Wei05] WEISSTEIN, Eric W.: *Sample Mean*. From MathWorld–A Wolfram Web Resource. `http://mathworld.wolfram.com/SampleMean.html`, 2005

[WF05] WITTEN, Ian H. ; FRANK, Eibe ; KAUFMANN, Morgan (Hrsg.): *Data Mining: Practical machine learning tools and techniques*. 2nd Edition. San Francisco : Diane Cerra, 2005

[Whi04] WHITE, Tom: *Can't beat Jazzy – Introducing the Java platform's Jazzy new spell checker API*. WEB. Sep 2004. – `http://www.ibm.com/developerworks/java/library/j-jazzy/` [06.08.2008]

[WJ95] Wooldridge, M. ; Jennings, N. R.: Intelligent Agents: Theory and Practice. In: *The Knowledge Engineering Review* 10 (1995), Nr. 2, S. 115–152

[WLP04] Wenyao Liu, Zhaohui W. ; Pan, Gang: An Entropy-Based Diversity Measure for Classifier Combining and Its Application to Face Classifier Ensemble Thinning. In: *Advances in Biometric Person Authentication* Bd. 3338/2004, Springer Berlin, Heidelberg, 2004, S. 118–124

[Wol05] Wolling, Jens: Suchmaschinen? - Selektiermaschinen! In: Krömker, Heidi (Hrsg.) ; Klimsa, Paul (Hrsg.): *Handbuch Medienproduktion. Produktion von Film, Fernsehen, Hörfunk, Print, Internet, Mobilfunk und Musik.* Wiesbaden : Verlag für Sozialwissenschaften, 2005, S. 529–537

[WR84] Wong, S. K. M. ; Raghavan, Vijay V.: Vector space model of information retrieval: a reevaluation. In: *SIGIR '84: Proceedings of the 7th annual international ACM SIGIR conference on Research and development in information retrieval.* Swinton, UK : British Computer Society, 1984. – ISBN 0–521–26865–6, S. 167–185

[WRJR05] White, Ryen W. ; Ruthven, Ian ; Jose, Joemon M. ; Rijsbergen, C. J. V.: Evaluating implicit feedback models using searcher simulations. In: *ACM Trans. Inf. Syst.* 23 (2005), Nr. 3, S. 325–361. – ISSN 1046–8188

[WY95] Wong, S. K. M. ; Yao, Y. Y.: On modeling information retrieval with probabilistic inference. In: *ACM Trans. Inf. Syst.* 13 (1995), Nr. 1, S. 38–68. – ISSN 1046–8188

[WZB08] Wetzker, Robert ; Zimmermann, Carsten ; Bauckhage, Christian: Analyzing Social Bookmarking Systems: A del.icio.us Cookbook. In: *Proceedings of the ECAI 2008 Mining Social Data Workshop*, IOS Press, 2008, S. 26–30

[WZRW87] Wong, S. K. ; Ziarko, W. ; Raghavan, V. V. ; Wong, P. C.: On modeling of information retrieval concepts in vector spaces. In: *ACM Trans. Database Syst.* 12 (1987), Nr. 2, S. 299–321. – ISSN 0362–5915

[XLL07] Xu, Jun ; Liu, Tie-Yan ; Li, Hang. *The OHSUMED Datasets in LETOR.* Part of the OHSUMED dataset in LETOR description. May 2007

[XYZ+05] Xue, Gui-Rong ; Yang, Qiang ; Zeng, Hua-Jun ; Yu, Yong ; Chen, Zheng: Exploiting the hierarchical structure for link analysis. In: *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval.* New York, NY, USA : ACM, 2005. – ISBN 1–59593–034–5, S. 186–193

[YA06] Yilmaz, Emine ; Aslam, Javed A.: Estimating average precision with incomplete and imperfect judgments. In: *CIKM '06: Proceedings of the 15th ACM international conference on Information and knowledge management.* New York, NY, USA : ACM Press, 2006. – ISBN 1–59593–433–2, S. 102–111

[YMI02] Y. Matsuo, H. F. ; Ishizuka, M.: Browsing support by highlighting keywords based on a user's browsing history. In: *Systems, Man and Cybernetics, 2002 IEEE International Conference*, ACM Press, Oct 2002. – ISBN 0–7803–7437–1, S. 6–12

[YS02] Yu, Bin ; Singh, Munindar P.: An evidential model of distributed reputation management. In: *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems.* New York, NY, USA : ACM Press, 2002. – ISBN 1–58113–480–0, S. 294–301

[ZE98] Zamir, Oren ; Etzioni, Oren: Web Document Clustering: A Feasibility Demonstration. In: *Research and Development in Information Retrieval*, 1998, S. 46–54

[ZL01] Zhai, Chengxiang ; Lafferty, John: A study of smoothing methods for language models applied to Ad Hoc information retrieval. In: *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval.* New York, NY, USA : ACM, 2001. – ISBN 1–58113–331–6, S. 334–342

[Zob98] Zobel, Justin: How reliable are the results of large-scale information retrieval experiments? In: *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval.* New York, NY, USA : ACM Press, 1998. – ISBN 1–58113–015–5, S. 307–314

[ZZ07] Zhang, Min-Ling ; Zhou, Zhi-Hua: Multi-Label Learning by Instance Differentiation. In: *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*, AAAI Press, 2007. – ISBN 978–1–57735–323–2, S. 669–674