Bauhaus-Universität Weimar
Fakultät Medien
Studiengang Mediensysteme

Ein Kaskadierendes Verfahren zur Detektierung von Suchsitzungen in Anfrage-Log-Dateien

Bachelorarbeit

Tino Rüb
Geboren am 28. September 1981 in Wuppertal

Matrikelnummer 21059

1. Gutachter: Prof. Dr. Benno Stein
Betreuer: N.N.

Datum der Abgabe: 26. Januar 2011

Erklärung

Hiermit versichere ich, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Weimar, den 26. Januar 2011

.....
Tino Rüb

Zusammenfassung

In eine Suchsitzung fallen eine oder mehrere Suchanfragen eines Benutzers, die auf die gleiche Intention zurückzuführen sind und in einem relativ kurzen Zeitraum erfolgen. Durch den, im Vergleich zu einer einzelnen Anfrage höheren Informationsgehalt einer Suchsitzung können hierauf aufbauende Algorithmen und Retrieval Verfahren eine bessere Güte erreichen. Eine aussagesichere Detektierung von Suchsitzungen ist hierbei die Voraussetzung, um aus mehreren Anfragen die Intention des Benutzers genauer bestimmen zu können.

Wir entwickeln auf Grundlage der geometrischen Methode ein kaskadierendes Verfahren, welches verschiedene Merkmale zur Detektierung einer Suchsitzung berücksichtigt. Unser primäres Ziel ist die Optimierung der konkurrierenden Eigenschaften *Güte* und *Laufzeit*. Zudem möchten wir Suchsitzungen mit einer maximalen Genauigkeit bestimmen, indem wir einen Teil der Suchsichtzungen ignorieren, die nicht mit einer hohen Aussagesicherheit detektiert wurden. Die Quantifizierung der einzelnen Merkmale erfolgt kaskadierend hinsichtlich des benötigten Aufwands. Wenn bereits ein „kostengünstiges“ Merkmal eine hohe Aussagesicherheit ermöglicht, werden keine weiteren Merkmale quantifiziert. Neben einigen Merkmalen, die die eigentliche Suchanfrage und den Zeitpunkt behandeln, quantifizieren wir ein Merkmal auf Basis der expliziten semantischen Analyse (*ESA*). In einem letzen Schritt ermitteln wir ein, dass Ähnlichkeitsmaß durch die erweiterte Repräsentation der Ergebnisse einer Suchmaschine gebildet wird.

Unser kaskadierendes Verfahren verbessert die erreichte Güte im Vergleich zu einem Referenzwert von $F_{\beta=1.5} = 0{,}918$ auf $F_{\beta=1.5} = 0{,}932$. Für den überwiegenden Anteil von 74,6% der Suchanfragen, kann ein F-Measure in der Höhe von 0,975 bestimmt werden. Des Weiteren konnten wir zeigen, dass die Aussagesicherheit einer *ESA* auf Basis der Wikipedia eine so hohe Güte erreicht, sodass ein Ähnlichkeitsmaß auf Basis der Ergebnisse einer Suchmaschine für die betrachtete Problemstellung nur noch eine marginale Verbesserung ermöglicht.

Inhaltsverzeichnis

Kapitel 1

Einleitung

Aktuelle Internetsuchmaschinen ermöglichen es uns durch ihre Technologien, die immer größer werdenden Dokumentenkollektionen des WWW erfassen und auswerten zu können. Von Dezember 2008 bis Dezember 2009 erhöhte sich die Anzahl der weltweiten Suchanfragen um 46%. Pro Tag wurden in diesem Beobachtungszeitraum durchschnittlich 4 Milliarden Suchanfragen gestellt [cI10]. Das lässt auf den Stellenwert von Suchmaschinen in unserem Umgang mit dem WWW schließen. Entsprechend ausgeprägt und vielfältig sind die Bestrebungen zur Optimierung von Suchmaschinen.

Das übergeordnete Ziel vieler Studien ist es, die Suche für den Benutzer so effektiv und effizient wie möglich zu gestalten und somit die Zufriedenheit bei der Nutzung einer Internetsuchmaschine zu maximieren. Für den überwiegenden Teil, 90–95% der Suchanfragen, können die heutigen Suchmaschinen Resultate in ausreichender Güte liefern [ZCBY10]. Für die restlichen 5–10% der Anfragen ist es nach wie vor schwierig, der Intention des Benutzers entsprechende Suchergebnisse zu erreichen.

Genauer betrachtet ist die Suche über eine computerbasierte Schnittstelle ein komplexer Prozess, der sich über mehrere Suchanfragen erstrecken kann. Dieser Prozess folgt meistens dem klassischen Prinzip „Versuch und Irrtum“. Ein Benutzer sucht so lange, bis entweder sein Ziel erreicht ist oder er die Suche abbricht [Swa77]. Aus jeder nicht zufriedenstellend beantworteten Anfrage kann der Benutzer schließen, dass er seine Suchanfrage verändern muss. Zum Beispiel ist ein gängiges Veränderungsmuster von Anfragen deren Umformulierung in Form einer Generalisierung oder einer Spezialisierung der vorherigen Suchanfrage. Im Fall der Generalisierung versucht der Benutzer, durch die Entfernung von Termen die Ergebnisanzahl zu erhöhen. Durch das Hinzufügen von Termen versucht er im Fall der Spezialisierung seine Anfrage zu konkretisieren, um so eine eingeschränktere Ergebnismenge zu erhalten.

Nach Silverstein et al. [SMHM99] besteht eine Suchsitzung[1] aus einer oder mehreren Anfragen, die in einem relativ kurzen Zeitraum erfolgen, und die einem spezifischen Informationsbedarf des Benutzers entsprechen.

Unter anderem ist diese Definition für die korrekte Validierung einer Suchmaschine maßgeblich. Wenn der Benutzer auf kein einziges der präsentierten Ergebnisse klickt, ist dies ein Indiz für eine nicht erfolgreich beantwortete Suchanfrage. Einzeln betrachtet ist das bei 30% aller Anfragen der Fall [Yu04]. Da nicht jede Suchanfrage einer neuen Intention des Benutzers entspricht, ist diese Betrachtung natürlich ungenau. Eine Validierung einer Suchmaschine ist demnach differenzierter und präziser, wenn mehrere zusammengehörende Anfragen auch zusammen betrachtet werden und ein wechselnder Informationsbedarf berücksichtigt wird.

Durch den höheren Informationsgehalt von mehreren, zusammengefassten Suchanfragen lässt sich die Intention des Benutzers im Vergleich zu einer einzelnen Anfrage besser ermitteln. Suchsitzungen können demzufolge als Grundlage für eine erweiterte Repräsentation einer Suchanfrage dienen. Somit ermöglicht das Modell einer Suchsitzung darauf aufbauende Funktionen und Algorithmen, wie zum Beispiel *relevance feedback* [RL03, SJÖ00], *maximum query* oder *query cover* [SH10], die gegebenenfalls für den Benutzer eine optimalere Ergebnismenge erreichen.

Eine möglichst korrekte Aufteilung von Anfragen in Suchsitzungen ist dafür allerdings die Voraussetzung. Die Qualität der hierauf aufbauenden Funktionen und Algorithmen zur Verbesserung der Suchergebnisse kann nur so gut sein, wie die Qualität der ihnen zugrunde liegenden Suchsitzungen. Wir widmen uns daher der wichtigen Frage, welche aufeinanderfolgenden Anfragen eines Benutzers in einem, beziehungsweise in keinem Zusammenhang stehen.

AnonID	Query	QueryTime	ItemRank	ClickURL
68501	cadaver museum	2006-03-27 17:31:10	7	http://news...
68501	museum of sience and industry in tampa florida	2006-03-27 17:38:25	1	http://www.m..
68501	clip art	2006-04-10 20:02:06	2	http://www.c..
68501	easter	2006-04-10 20:11:57		
68501	easter bunny clip art	2006-04-11 19:14:29	1	http://www.w..

Abbildung 1.1: Charakteristischer Auszug einer Anfrage-Log-Datei

Als Datenbasis dient eine Anfrage-Log-Datei der AOL-Suchmaschine aus dem Jahr 2006. Anhand eines exemplarischen Auszugs (Abb. 1.1) lässt sich erkennen, welche Informationen vorliegen. Es ist protokolliert, wann welcher Be-

[1] Im engl.: *search session*. Weitere Schlagwörter vergleichbarer Arbeiten: *session detection*, *task detection*, *session boundary*, *sessionization*, *session segmentation*, *query chains*.

nutzer welche Anfrage an die Suchmaschine gesendet hat. Weiterhin ist protokolliert, ob ein oder mehrere der präsentierten Ergebnisse vom Benutzer aufgerufen wurden. In diesem Fall wird die Domain der entsprechenden Ziel-URL sowie der Rang der Platzierung dieser URL innerhalb der Suchergebnisse angegeben.

Das Ziel dieser Arbeit ist die Umsetzung eines kaskadierenden Verfahrens, welches verschiedene Merkmale zur Detektierung einer Suchsitzung verwendet. Die Berücksichtigung des zeitlichen Abstands zweier Suchanfragen und ein simpler Vergleich beider Zeichenketten erfolgen in einem ersten Schritt. Als Basis zur Detektierung der Suchsitzungen anhand zweier aufeinanderfolgender Suchanfragen dient hierbei das *Geometrische Verfahren* von Gayo-Avello [Gay09], dass diese beiden Merkmale in ein Verhältnis setzt. Darauf aufbauend soll unser kaskadierendes Verfahren, je nach Fall, erweiterte Merkmale quantifizieren, falls der erste Schritt nicht zu einem zufriedenstellenden Ergebnis führt. Dazu werden verschiedene Merkmale und mögliche Ähnlichkeitsmaße evaluiert und gegenübergestellt. Ein Beispiel ist die Erweiterung einer Suchanfrage um die Ergebnisse einer Suchmaschine oder ein Ähnlichkeitsmaß basierend auf der expliziten semantischen Analyse (*ESA*). Die Laufzeit der jeweiligen Quantifizierung für die verschiedenen Merkmale wird in unserem Verfahren berücksichtigt, um eine möglichst effiziente Methode zu entwickeln. Entsprechend dem Verhältnis zwischen den jeweiligen „Kosten“ zur Quantifizierung eines Merkmals mit einer längeren Laufzeit und der erreichten Aussagesicherheit der bisherigen Merkmale soll entschieden werden, ob ein Abbruch oder eine Fortführung der Merkmalsquantifizierung für ein Anfragepaar erfolgt.

Das kaskadierende Verfahren wird auf der Grundlage einer umfangreichen Recherche und der Evaluierung bestehender Forschungsergebnisse entwickelt. Es wird untersucht, mit welcher jeweiligen Aussagesicherheit die einzelnen Suchsitzungen korrekt detektiert werden. Diese Information ermöglicht es den auf Suchsitzungen aufbauenden Verfahren zur Optimierung der Suchergebnisse, zum Beispiel nur die Suchsitzungen zu berücksichtigen, die mit einer ausreichend hohen Wahrscheinlichkeit richtig detektiert wurden.

Die Arbeit ist wie folgt gegliedert: Das nachfolgende Kapitel 2 zeigt vorhandene Überlegungen zur Datenaufbereitung sowie eine Übersicht der Merkmale und Verfahren in der Literatur. Eine ausführliche Erläuterung und Evaluierung des von uns entwickelten Verfahrens findet sich in Kapitel 3. Das Kapitel 4 fasst die Ergebnisse zusammen, diskutiert diese und schließt mit einem Ausblick die Arbeit ab.

Kapitel 2

Verwandte Arbeiten

2.1 Datenaufbereitung

Interessant für weitergehende Überlegungen sind in der Regel nur die von realen Menschen gestellten Suchanfragen. Computergenerierte Anfragen sollten dafür im Vorfeld aus einer Anfrage-Log-Datei entfernt werden. Diese werden automatisch, durch sogenannte *software agents* oder auch *bots*, aus unterschiedlichen Motiven generiert und abgesendet. Ebenso sollten die Suchanfragen von Meta-Suchmaschinen entfernt werden, da diese von vielen unterschiedlichen Benutzern erstellt und nicht eindeutig einzelnen Personen zugeordnet werden können.

Mit der Zielsetzung, nur die sinnvollen Anfragen für weitere Untersuchungen zu verwenden, wird in den nachfolgend beschriebenen Arbeiten versucht, diese beiden Gruppen zu entfernen. Einige Arbeiten basieren auf der Annahme, dass eine erhöhte Anfrageanzahl im Verhältnis zu einem zeitlichen Intervall auf eine nicht-menschliche Quelle schließen lässt. Die Größe des zeitlichen Intervalls basiert oft auf der Definition einer *search episode*. Dies entspricht üblicherweise der Menge aller Anfragen eines Benutzers über den Zeitraum eines Tages [JSB07], in der Form einer Gruppierung der Anfragen über ein Intervall von 24 Stunden.

Um die entsprechenden Anfragen zu entfernen, ignoriert Gayo-Avello beispielsweise alle *search episodes*, die mehr als 100 Anfragen aufweisen [Gay09]. Fraglich ist bei diesem Ansatz, warum sich die Filterung nur auf die Suchanfragen der entsprechenden *search episodes* eines Benutzers und nicht auf alle Anfragen eines Benutzers bezieht. Gayo-Avello weißt in seiner Arbeit allerdings auch darauf hin, dass nach seiner Filterung nach wie vor Anfragen von *software agents* enthalten sind. Bei Stichproben stößt er nach wie vor auf Anfragen, die eindeutig *software agents* zuzuordnen sind.

Bei Buzikashvili [Buz07] wird ein ähnlicher Ansatz verfolgt. Hier werden

mit dem sogenannten *Dynamic Sliding Window* Verfahren bei unterschiedlichen Intervall-Längen zwei Werte in einem zeitlichen Intervall untersucht. Die Anfragen eines Benutzers werden aussortiert, wenn eine gewisse Anzahl an Anfragen oder eine bestimmte Anzahl anderer Interaktionen[1] mit der Suchmaschine überschritten wird.

Zhang und Moffat [ZM06] verfolgen ebenfalls den Ansatz von maximal 100 Anfragen, allerdings innerhalb einer Suchsitzung. Zudem werden nur die Suchsitzungen verwendet, denen mindestens ein Klick auf ein Resultat der Suchergebnisseite folgt. Dieser Ansatz entfernt dadurch aber auch viele Anfragen, die von realen Personen stammen.

Bei Duskin und Feitelson [DF09] wird neben der Anfrageanzahl ein exzessives Anfrageverhalten untersucht. Alle Benutzer mit durschnittlich mehr als einer Anfrage pro Sekunde werden entfernt. Interessant ist hierbei die Überlegung, neben den beiden bekannten Klassifizierungen zwischen menschlichen und nicht-menschlichen Anfragen eine Dritte einzuführen: „unbekannt“. Diese Gruppe enthält die Anfragen, bei denen nicht sicher zwischen menschlichen und maschinellen Quellen unterschieden werden kann. In ihrer Arbeit untersuchen Duskin und Feitelson drei unterschiedliche Anfrage-Log-Dateien und bestimmen dabei für die „unbekannte“ Gruppe einen Anteil zwischen 3% und 17%. Ein dementsprechend hoher Anteil an Anfragen muss jeweils entfernt werden, um mit hoher Wahrscheinlichkeit nur menschliche, beziehungsweise nicht-menschliche Anfragen zu erhalten.

Die Arbeit von Jansen et al. [JMSP06] untersucht weitere Eigenschaften, wie zum Beispiel die Verwendung von Booleschen Ausdrücken (`OR`, `AND`, `NOT`) und Operatoren (“, `-`, `+`), die aber leider keine weiteren Rückschlüsse auf die Unterscheidung der Gruppen zulassen.

Wir untersuchen in dieser Arbeit die vorgestellten Ideen und experimentieren mit verschiedenen Filter-Regeln, um Anfragen zu entfernen und von der weiteren Betrachtung auszuschließen. Bei näherer Betrachtung wird sich zeigen, dass die beiden Filter-Regeln von Gayo-Avello [Gay09] und von Duskin und Feitelson [DF09] äquivalent sind, sodass nur die Quantifizierung einer dieser beiden Eigenschaften nötig ist. Des Weiteren entfernen wir Benutzer, deren Median der Zeichenanzahl der Anfragen sehr lang ist. Benutzer für die eine weitere Betrachtung uninteressant ist, werden ebenso entfernt. So sind zum Beispiel alle Benutzer, mit nur einer Suchanfrage für eine Untersuchung hinsichtlich der Gruppierung von Anfragen in Suchsitzungen irrelevant.

Von dem Ansatz, die Anfragen im Vorfeld in *search episodes* zu gruppieren sehen wir ab, da dieser zum Beispiel eine Zeitzonenverschiebung des Benutzers oder dessen mögliche tagesübergreifende Benutzung einer Suchmaschine

[1] Zum Beispiel der Aufruf eines Resultats aus der Suchergebnismenge.

nicht berücksichtigt. Die oben genannten Eigenschaften setzen wir hingegen mit der Gesamtzeitspanne seiner Aktivität in ein Verhältnis. Dies erlaubt uns eine Betrachtung, die den Charakter der Benutzung frei von Tageszeiten widerspiegelt. Um die Zeitspanne zu normalisieren, werden längere Pausen zwischen zwei Interaktionen eines Benutzer mit der Suchmaschine von der Gesamtzeitspanne subtrahiert. So können wir sicherstellen, dass zum Beispiel Benutzer, die täglich suchen, mit denen, die im Durchschnitt nur wöchentlich suchen, vergleichbar sind.

2.2 Merkmale und Verfahren zur Detektierung von Suchsitzungen

Zwischen je zwei aufeinanderfolgenden Anfragen eines Benutzers ist die Frage zu klären, ob es sich hierbei um eine *Sitzungsfortführung* oder um eine *neue Suchsitzung* handelt. Dazu können verschiedene Merkmale herangezogen werden, die im weiteren Verlauf dieses Abschnitts näher erläutert werden. Unterschiedliche Verfahren kombinieren diese Merkmale, durch heuristische Ansätze oder maschinelle Lernverfahren, um eine geeignete Lösung für diese Problemstellung zu evaluieren. Diese werden am Ende dieses Abschnitts vorgestellt.

2.2.1 Zeitlicher Abstand zweier Anfragen

Je größer der zeitliche Abstand zwischen zwei Anfragen eines Benutzers, umso wahrscheinlicher ist es, dass diese zwei Suchanfragen jeweils zwei unterschiedlichen Suchsitzungen zuzuordnen sind [SMHM99]. Dazu werden in der Literatur Schwellwerte, wie weit zwei aufeinanderfolgende Suchanfragen maximal auseinanderliegen dürfen, um sie noch in einer Suchsitzung zusammenfassen zu können, definiert oder ermittelt.

Statische Schwellwerte. Einige Studien führen Untersuchungen mit verschiedenen statischen Schwellwerten mit zum Teil variierenden Ergebnissen durch. Catledge und Pitkow [CP95] analysieren mit einem modifizierten Browser das Verhalten der Benutzer. Der durchschnittliche Abstand zwischen zwei Interaktionen liegt bei 9,3 Minuten. Mit der Annahme, dass die statistisch signifikanten Aktionen eine 1,5-fache Standardabweichung aufweisen, ermitteln Catledge und Pitkow 25,5 Minuten als Schwellwert.

Die folgendenen Studien beruhen auf unterschiedlichen Anfrage-Log-Dateien. He und Görker [HG00] evaluieren eine Serie von Schwellwerten zwischen 1 und 200 Minuten. Als bester Schwellwert werden 10 bis 15 Minuten angenommen. Downey et al. [DDH07] kommen zu dem Schluss, dass ein Schwell-

wert von 30 Minuten zu einem zufriedenstellenden Ergebnis führt. Es ist also nachvollziehbar, wenn Murray et al. [MLC06] feststellen, dass ein statischer Schwellwert zu wenig Signifikanz aufweist und mit unterschiedlichen Korpora variiert. Jones und Klinkner [JK08] erläutern zudem, dass unabhängig von dem gewählten Schwellwert dieses Merkmal eine Genauigkeit von maximal 70% erreichen kann.

Benutzerorientierte Schwellwerte. Murray et al. [MLC06] versuchen anhand der Verteilung der jeweiligen Abstände zweier Anfragen eines Benutzers, mithilfe des *hierarchical agglomerative clustering* (*HAC*), einen individuellen Schwellwert pro Benutzer zu ermitteln. Derjenige Abstand zweier Suchanfragen wird hierbei gewählt, der im Verhältnis zu der Verteilung der jeweils individuellen zeitlichen Abstände die maximale Varianz aufweist. Der nächstkleinere zeitliche Abstand wird als Schwellwert für diesen Benutzer definiert.

Murray et al. merken an, dass das Merkmal des zeitlichen Abstands nur die Aktivität des Benutzers widerspiegelt. Dies hat nur eine beschränkte Aussagekraft über die Fortführung oder den Beginn einer Suchsitzung. Somit ist es ohne weitere Merkmale nicht möglich, Suchsitzungen mit einer hohen Genauigkeit zu bestimmen. Daher verwenden wir in dieser Arbeit den zeitlichen Abstand in Kombination mit der lexikalischen Merkmalsquantifizierung (vgl. Abschnitt 2.2.2) und anderen Merkmalen (vgl. Abschnitt 2.2.3). Die Kombination mit weiteren Merkmalen ist auch häufig in der Literatur zu finden. Diese stellen wir in den Abschnitten 2.2.5 und 2.2.6 vor. Durch eine Gegenüberstellung einiger systemorientierter Schwellwerte mit dem Verfahren von Murray et al. [MLC06], kommen wir zu dem Schluss, dass das *hierarchical agglomerative clustering* die Aussagesicherheit gegenüber einem systemorientierten Schwellwert nicht verbessert.

2.2.2 Lexikalische Ähnlichkeit

Um einen Themenwechsel und damit eine neue Suchsitzung zu erkennen, analysieren einige Arbeiten den Inhalt der Anfragen. Der Vergleich der Zeichenketten zweier Suchanfragen kann mit unterschiedlichen Verfahren umgesetzt werden.

Veränderungsmuster. Viele Arbeiten beschreiben gängige Veränderungsmuster zweier aufeinanderfolgender Suchanfragen eines Benutzers und kategorisieren diese in verschiedene Klassen [Gay09, HGH02, LH99, ÖÖS08, Spi07]. Je nach den gegebenen Informationen variieren die Klassen geringfügig. Die möglichen Interaktionen eines Benutzers mit einer Suchmaschine sind in den jeweiligen Anfrage-Log-Dateien unterschiedlich detailliert erfasst. Zum Beispiel

sind in einigen Log-Dateien Ereignisse, wie das Besuchen einer weiteren Seite der Suchergebnisse, protokolliert. Eine weitere mögliche Information ist der Ursprung einer Anfrage aufgrund zusätzlicher Bedienungsmöglichkeiten, wie die Verwendung einer Vorschlag-Funktion. Da dies bei der AOL-Log-Datei nicht der Fall ist, verwenden wir nur die Klassen, die sich auf die Suchanfrage als solches beziehen:

1. *Wiederholung:* Die vorausgegangene Suchanfrage wird wiederholt.
2. *Generalisierung:* Durch das Weglassen von Termen aus der vorangegangenen Suchanfrage wird versucht, die Ergebnismenge zu vergrößern.
3. *Spezialisierung:* Durch Hinzufügen von Termen zu der vorangegangenen Anfrage versucht der Benutzer, die Ergebnisse einzuschränken.
4. *Umformulierung:* Die vorherige Anfrage wird mithilfe anderer Suchwörter umformuliert, da der Benutzer mit dem Suchergebnis nicht zufrieden ist.
5. *Neu:* Die Anfrage befasst sich mit einem neuen Informationsbedarf.

Aufeinanderfolgende Suchanfragen der ersten drei Klassen sind leicht zuzuordnen, da dies mit gängigen Ähnlichkeitsmaßen für Zeichenketten gut möglich ist (vgl. nächster Absatz). Die Differenzierung der beiden letzten Klassen stellt allerdings eine Herausforderung dar, wie auch das *vocabulary-mismatch* Problem aufzeigt [Gay09]. Zum Beispiel sind die beiden Synonyme „IR“ und „information retrieval“ semantisch sehr ähnlich, können allerdings nur schwer als solche ermittelt werden.

Ähnlichkeit von Zeichenketten. Um die Ähnlichkeit zweier Zeichenketten zu ermitteln, ist eine einfache Möglichkeit, die Terme der Zeichenketten miteinander zu vergleichen. Jansen et al. [JBS09, JSB07] und Özmutlu et al. [ÖÖS08] vermuten zum Beispiel eine Themenänderung, wenn kein einziger Term zweier Suchanfragen übereinstimmt.

Die in diesem Abschnitt vorgestellten Methoden werden von den jeweiligen Autoren häufig kombiniert und für verschiedene maschinelle Lernverfahren verwendet. Wie auch bei den zeitlichen Untersuchungen variieren die Ergebnisse.

Als ein passendes Ähnlichkeitsmaß für das Abändern von Zeichenketten wird die normalisierte Levenshtein-Distanz oft zum Vergleichen von Suchanfragen verwendet (etwa in [SLY+10, LOP+10]). Hierbei wird die Mindestanzahl von Operationen ermittelt, die notwendig ist, um eine Zeichenkette x in eine andere Zeichenkette y abzuändern. Jones und Klinkner [JK08] bewerten die

Levenshtein-Distanz als das aussagekräftigste Merkmal zum Vergleich zweier Zeichenketten im Rahmen von Suchanfragen.

Einfache Vergleiche von n-Grammen sind ein weiterer Ansatz. Zhang und Moffat [ZM06] ermitteln den Jaccard-Koeffizienten, indem sie die Anzahl übereinstimmender Trigramme zweier Suchanfragen zur Anzahl aller Trigramme beider Anfragen ins Verhältnis setzen. Sun et al. [SLY+10] lösen die Problemstellung vergleichbar und bewerten den Jaccard-Koeffizienten von n-Grammen als das signifikanteste lexikalische Merkmal. Je nach untersuchter Anfrage-Log-Datei bestimmen sie unterschiedliche Längen n. Für die in dieser Arbeit untersuchte AOL-Log-Datei wird die beste Länge der n-Gramme mit 6 angegeben. Gayo-Avello [Gay09] quantifiziert ein nicht näher spezifiziertes Verhältnis über zwei Vektorrepräsentationen der jeweiligen n-Gramme. Lucchese et al. [LOP+10] nehmen an, dass das Minimum der jeweils ermittelten Levenshtein-Distanz und des Jaccard-Koeffizienten von Trigrammen ein adäquates Mittel ist. Buzikashvili [Buz06a] verwendet ebenfalls n-Gramme für seine Untersuchungen, geht hierauf aber nicht weiter ein. Jones und Klinkner untersuchen hingegen nur einen Ansatz mit zwei restriktiveren Regeln, die lediglich die Anzahl gleicher Buchstaben am Anfang oder am Ende eines jeden Terms berücksichtigen.

Gayo-Avello hält fest, dass sich eher Vorteile durch die Berücksichtigung von n-Grammen ergeben als durch den Vergleich von Termen. Der übliche Ansatz, bei Textvergleichen Terme auf den Wortstamm zurückzuführen, entfällt so laut [Gay09] im Bezug auf die Ermittlung eines Ähnlichkeitsmaßes. Trotz fehlerhafter Eingaben, wie zum Beispiel bei verdrehter Reihenfolge von Buchstaben, kann eine hohe Ähnlichkeit festgestellt werden, da ein Vergleich mit n-Grammen durch geringfügig geänderte Zeichenketten nur gering beeinflusst wird.

In der ersten Stufe unseres kaskadierenden Verfahrens quantifizieren wir ein einfaches Merkmal wie die Generalisierung oder die Spezialisierung einer Suchanfrage. Falls eine Anfrage die darauffolgende enthält (oder im umgekehrten Fall) und diese beiden Anfragen einen maximalen zeitlichen Abstand nicht überschreiten, folgern wir eine Sitzungsfortführung. Als das beste lexikalische Merkmal der darauf aufbauenden geometrischen Methode ermitteln wir die Kosinusähnlichkeit von n- bis m-Grammen mit den Längen $n = 3$ und $m = 8$.

2.2.3 Erweiterte Ähnlichkeitsmaße

Da Anfragen sehr kurz sind und nur wenig Text aufweisen, ist eine Betrachtung der Zeichenketten für ein Ähnlichkeitsmaß oft nicht ausreichend [CC04]. Besonders die Differenzierung zwischen der Umformulierung vorheriger Anfragen und der Formulierung einer neuen Intention, ist eine Herausforderung.

Daher ist eine Repräsentation einer Anfrage wünschenswert, die einen höheren Informationsgehalt als die ursprüngliche Suchanfrage aufweist und somit eine geeignetere Grundlage für eine Unterscheidung bieten kann.

Erweiterung durch Teile der Ergebnismenge. Eine Idee ist es, die Ergebnisse einer Suchmaschine für ein Ähnlichkeitsmaß zu verwenden [MDM07]. Radlinski und Joachims [RJ05] vergleichen lediglich die URLs der ersten 10 Ergebnisse zweier Suchanfragen auf Übereinstimmung. Die Ergebnismenge beinhaltet noch weitere Daten, wie den Titel und einen kurzen Text-Auszug, die für einen Vergleich verwendet werden können. Zudem können die verlinkten Dokumente der Resultate analysiert werden.

Die Verfahren der nachfolgend genannten Arbeiten stimmen nahezu überein und variieren nur in der Anzahl der verwendeten Resultate. Von den Resultaten einer Anfrage werden die ersten 10 [Gay09], über 50 [Gay09, JK08, STZ05], 100 [CC04] bis zu 500 [MDM07] verwendet. Diesen Verfahren liegen jeweils stoppwortgefiltert die Titel und die Text-Auszüge der Ergebnis-Seiten zugrunde. Dabei werden Terme entfernt, die sehr häufig vorkommen und nur eine syntaktische oder ein grammatikalische Funktion besitzen. Zum Beispiel 'a', 'of', 'the', 'and' und weitere. Diese stoppwortgefilterten Inhalte können als eine erweiterte Repräsentation der Suchanfragen interpretiert werden. Die Ähnlichkeit zweier Anfragen wird ermittelt, indem die jeweiligen Repräsentationen $tf \cdot idf$-gewichtet über die Kosinusähnlichkeit verglichen werden. Wenn das erste Resultat auf einen Wikipedia-Artikel verlinkt, berücksichtigt Gayo-Avello [Gay09] zudem die ersten 4 KB dieses Artikels.

Für die erweiterte Repräsentation, die Sahami und Heilman [SH06] verwenden, wird hingegen der vollständige Inhalt der Dokumente berücksichtigt, die in den Ergebnissen der Suchmaschine verlinkt sind.

Gayo-Avello ermittelt seine untersuchte Erweiterung der Suchanfragen, im Vergleich zu anderen Ähnlichkeitsmaßen, als eines der aussagekräftigeren. Auch Jones und Klinkner evaluieren dieses Ähnlichkeitsmaß in ihrer Arbeit und stellen es anderen gegenüber. Als Resultat ihrer Experimente ermitteln sie es als das Beste.

Die Ermittlung der Daten erfolgt in der Regel über computergenerierte Anfragen an eine Suchmaschinen-API. Da Netzwerkanfragen über Intranetgrenzen hinaus sehr teuer sind, versuchen wir in unserem Verfahren, Teile der Ergebnismenge nur dann miteinzubeziehen, wenn günstigere Merkmale nicht genügend Aussagekraft aufweisen. Die drei unterschiedlichen Methoden, (1) nur die URLs, (2) den Inhalt der Ergebnisseiten oder (3) den Inhalt der Ergebnisdokumente für den Vergleich zu verwenden, werden von uns evaluiert. Zudem werden wir die bestmögliche Güte für die Anzahl der ersten 10, 20 und 50 Ergebnisse gegenüberstellen.

Erweiterung durch Konzepte. Gabrilovich und Markovitch erörtern, dass der aufbereitete Inhalt eines Wikipedia-Artikels als ein erweitertes semantisches Konzept für den eigentlichen Begriff des Artikel-Themas verwendet werden kann [GM07]. Die von Gabrilovich und Markovitch vorgestellte explizite semantische Analyse (*ESA*) basiert auf dem Vektorraummodell und ist ein empirisch begründetes, kollektionsrelatives Retrieval Modell [SA09]. Eine Term-Dokument-Matrix wird durch eine $tf \cdot idf$-gewichtete Indexierung von Wikipedia-Artikeln erstellt. Mit dieser vorbereiteten Datenbasis lässt sich eine Vektor-Repräsentation von gewichteten Konzept-Ähnlichkeiten eines gegebenen Textes erzeugen. Diese Vektor-Repräsentation für einen gegebenen Text verwenden Gabrilovich und Markovitch als ein Ähnlichkeitsmaß für Textfragmente und Zeichenketten, in dem sie die Kosinusähnlichkeit dieser ermitteln. Für unsere Problemstellung, dem Vergleich zweier Suchanfragen, wurde diese Idee bereits aufgegriffen. Lucchese et al. [LOP+10] evaluieren diese erweiterte Repräsentation ähnlicher Wikipedia-Artikel als ein Ähnlichkeitsmaß im Kontext der Detektierung von Suchsitzungen. Neben den Artikeln der Wikipedia verwenden sie zudem die Artikel der Wiktionary-Webseite. Nach der Idee von Jones und Klinkner [JK08] vergleichen Lucchese et al. alle Anfragen und gruppieren sie unabhängig der sequentiellen Abfolge. In ihren Experimenten können sie damit bessere Ergebnisse erreichen, als es im Vergleich zu n-Grammen oder der Levenshtein-Distanz der Fall ist.

Der Vorteil der *ESA* ist, dass die Merkmalsquantifizierung zur Laufzeit relativ kostengünstig erfolgen kann, wenn die Daten schon lokal aufbereitet vorliegen. In unserem Verfahren werden wir die *ESA* daher vor einer möglichen Anfrageerweiterung durch Teile der Suchergebnismenge berücksichtigen, falls uns die *ESA* für unsere Problemstellung eine verbesserte Aussagesicherheit ermöglicht. Um eine bestmögliche Güte in einem ausgewogenen Verhältnis zu den benötigten Ressourcen zu erreichen, werden wir mit einer verschiedenen Anzahl von Artikeln experimentieren. Nach Stein und Anderka [AS09, SA09] ist eine Kollektionsgröße zwischen 50.000 und 100.000 Artikeln für eine hohe Güte, und eine Größe von 1.000 Artikeln für eine hohe Effizienz nötig.

2.2.4 Weitere Betrachtung

Über die Ähnlichkeit der Zeichenketten und der erweiterten Repräsentationen hinaus lassen sich Suchanfragen nach der Intention der Benutzer kategorisieren. Unter Umständen können Rückschlüsse gezogen werden, die bei einer Betrachtung von Suchsitzungen relevant sind. Eine hierarchische Einordnung der Intention der jeweiligen Anfragen ermöglicht es, längere Suchsitzungen zu detektieren, die wiederum durch andere unterbrochen werden.

Kategorisierung der Intention. Nach Broder [Bro02] kann die Intention eines Benutzers in die folgenden 3 Kategorien unterteilt werden:

1. *Navigation:* Der Benutzer möchte eine bestimmte Webseite aufrufen.
2. *Information:* Er möchte einen bestimmten Informationsbedarf decken.
3. *Transaktion:* Der Benutzer möchte eine bestimmte Transaktion ausführen (zum Beispiel eine Datei herunterladen oder ein Produkt bestellen).

Aufbauend auf dieser Einteilung haben sich weitere Autoren mit diesem Thema beschäftigt [ZCBY10, RL04]. Allerdings ist diese Kategorisierung bisher bei keinem Verfahren verwendet worden, dass Suchsitzungen detektiert. Wir haben in dieser Arbeit analysiert, ob sich aus der Unterscheidung der einzelnen Kategorien auf einen Zusammenhang im Sinne von Suchsitzungen schließen lässt. Jedoch konnten wir keinen Zusammenhang ableiten.

Multitasking und hierarchische Relation. Spink et al. ermitteln in ihren Arbeiten zwischen 61% und 83% der Suchsitzungen (detektiert über zeitliche Schwellwerte) mit mehr als einer Intention und erläutern hieran den Begriff Multitasking [SPJP06, SÖÖ02]. Nach der Definition von Silverstein et al. [SMHM99], ziehen wir es allerdings vor, verschiedene Intentionen als einzelne Suchsitzungen aufzufassen.

Buzikashvili [Buz06a, Buz06b] bezeichnet die sequentielle Abfolge verschiedener Intentionen, wie es Spink et al. als Multitasking interpretieren, daher auch als „Pseudo-Multitasking“. Er unterteilt reales Multitasking in 2 Gruppen. Wenn ein Such-Ziel durch wenige andere Anfragen kurz unterbrochen und dann wieder fortgesetzt wird, definiert Buzikashvili eine „einhüllende“ Art. Er ermittelt für diese Gruppe nur einen geringen Anteil, von maximal 1% der untersuchten Suchsitzungen. Die zweite Gruppe beschreibt er als „zufälliges“ Multitasking, wobei sich hier zwei oder mehr Such-Ziele jeweils abwechseln. Innerhalb kurzer Untersuchungsperioden von einigen Stunden lässt sich diese Charakteristik der zweiten Gruppe nicht ermitteln. Doch je länger der analysierte Zeitraum, desto mehr Suchsitzungen lassen sich zusammenfassen. Das ist darauf zurückzuführen, dass Wiederholungen der gleichen Such-Ziele und -anfragen der Benutzer in einem größeren Betrachtungszeitraum häufiger auftreten.

Seco und Cardoso [SC06] betrachten hingegen nicht die sequentielle Reihenfolge der Suchsitzungen, sondern berücksichtigen alle vorherigen Sitzungen anhand eines zeitlichen Schwellwerts von 60 Minuten. Nach jedem neuen Anfragevergleich werden die Suchsitzungen dieser Gruppe überprüft, ob diese jeweils in Relation mit dem Zeitstempel der aktuell betrachteten Anfrage den

Schwellwert des zeitlichen Abstands überschreiten. Diese Sitzungen werden jeweils auf eine Termüberlappung mit den Suchsitzungen untersucht, die noch einen zeitlichen Abstand unterhalb des Schwellwerts von 60 Minuten aufweisen. Wenn mindestens ein Term übereinstimmt, werden jeweils zwei Sitzungen zusammengeführt. Der Anteil der so zusammengeführten Suchsitzungen liegt laut Seco und Cardoso bei 1.7%, was ebenfalls auf einen geringen Anteil zeitnahen Multitaskings schließen lässt.

Jones und Klinkner verfolgen einen anderen Ansatz und entwickeln eine Einteilung in Such-Ziele und übergeordnete Such-Missionen [LOP+10, JK08]. Hiermit ist gemeint, dass sich mehrere, unter Umständen nur wenig ähnliche Suchsitzungen, durch einen gemeinsamen Kontext zusammenfassen lassen. Zum Beispiel die einzelnen Such-Ziele mit den jeweiligen Intentionen (1) Übernachtungsmöglichkeiten an einem Urlaubsort zu erkunden, (2) einen Wechselkurs für Devisen zu ermitteln und (3) entsprechende Flüge zu buchen, lassen sich in der übergeordneten Such-Mission „Urlaubsplanung“ zusammenfassen.

Wir haben die Anfrage-Log-Datei ausgehend von Buzikashvilis Definition der „einhüllenden“ Art untersucht. Spink et al. und Buzikashvili verwenden für ihre Überlegungen nur ein lexikalisches Ähnlichkeitsmaß. Für unsere Analyse quantifizieren wir hingegen analog zu Jones und Klinkner weitere Merkmale. Leider blieben diese Experimente ohne Ergebnis, wobei allerdings erwähnt werden muss, dass die von uns zur Güteprüfung verwendete annotierte Teilmenge der Anfrage-Log-Datei ein mögliches Multitasking nicht berücksichtigt. Stichproben sowie ermittelte Ähnlichkeitsverteilungen lassen zwar ein sinnvolles Zusammenfallen von Suchsitzungen vermuten, sind aber ohne eine geeignete Evaluierung nicht zu verifizieren.

2.2.5 Heuristische Verfahren

Die simpleren Heuristiken verwenden nur ein einziges Merkmal, wie etwa den zeitlichen Abstand zweier Anfragen, und erzielen damit unzureichende Ergebnisse aufgrund der erläuterten Schwächen. In der Literatur finden sich jedoch auch einige heuristische Verfahren, die mehrere Merkmale zu einer Entscheidungsfindung kombinieren. Diese werden in diesem Abschnitt vorgestellt.

Regelbasiert. Huang und Efthimiadis [HE09] verwenden einen einfachen regelbasierten Klassifizierer[2], um für zwei aufeinanderfolgende Suchanfragen eines Benutzers zwischen einer Sitzungsfortführung und einer neuen Suchsitzung zu unterscheiden. Wenn eine der folgenden lexikalischen Regeln greift, wird die Sitzung fortgeführt. Sie beschreiben Ihre Merkmals-Regeln als eine

[2]Quellcode: http://jeffhuang.com/reformulationClassifier.py

„Taxonomie der Umformulierung“. Die Merkmalsquantifizierung erfolgt in den folgenden Schritten:

1. Änderung der Termreihenfolge.
2. Änderung von Leerzeichen oder geänderte Zeichensetzung.
3. Hinzufügen von Termen.
4. Löschen von Termen.
5. Löschen von URL-spezifischen Parametern.
6. Rückführung auf den Wortstamm von Termen.
7. Abkürzung von Termen.
8. Abkürzungen zu vollen Termen ausschreiben.
9. Löschen eines Präfix oder Suffix der Anfrage.
10. Hinzufügen eines Präfix oder Suffix zur Anfrage.
11. Ersetzung von Termen (Synonym, Hyponym, Hypernym, Meronym, Holonym) über Wordnet Database[3].
12. Levenshtein-Distanz <= 2.

Wenn ein Merkmal die beiden Suchanfragen eines Benutzers mit der entsprechenden Regel als gleiche Suchsitzung detektiert, werden die weiteren Merkmale nicht quantifiziert. Diese Implementierungsentscheidung von Huang und Efthimiadis wird hinsichtlich der Laufzeit leider nicht weiter diskutiert. Der zeitliche Abstand zweier Suchanfragen wird bei ihrem Verfahren nicht berücksichtigt.

Seco und Cardoso [SC06] kombinieren hingegen den zeitlichen Abstand zweier Suchanfragen eines Benutzers mit einem einfachen lexikalischen Ähnlichkeitsmaß. Die Terme der Anfragen werden hierzu auf den Wortstamm zurückgeführt. Wenn die Suchanfragen nicht weiter als 60 Minuten auseinanderliegen und mindestens ein Term überlappt, werden die Anfragen in einer Suchsitzung zusammengeführt. Des Weiteren werden in einem zweiten Schritt ähnliche Suchsitzungen zusammengeführt (vgl. Abschnitt 2.2.4).

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

Dynamic Sliding Window. Buzikashvili [Buz07, Buz06b] verwendet dieses Verfahren lediglich zur Entfernung von nicht-menschlichen Suchanfragen (vgl. Abschnitt 2.1). Shi und Yang [SY06] nutzen es hingegen auch zur Segmentierung von Suchsitzungen. Wenn zwei Suchanfragen eines Benutzers weniger als 5 Minuten auseinanderliegen, und die Suchsitzung eine Gesamtlänge von 24 Stunden nicht überschreitet, fügen Shi und Yang diese Anfrage der Suchsitzung hinzu. Falls die Anfrage mit mehr als 5 und weniger als 60 Minuten Abstand auf die vorherige abgesendet wurde, wird eine (invertierte) normalisierte Levenshtein-Distanz auf Term-Basis ermittelt. Bei einem Wert größer als 0,4 wird angenommen, dass die Anfrage, unabhängig von der zuvor definierten Maximallänge einer Suchsitzung von 24 Stunden, der aktuellen Suchsitzung zuzuordnen ist.

Geometrische Methode. Gayo-Avello [Gay09] untersucht in seiner Arbeit einige Verfahren aus der Literatur (unter anderem [ÖÖS08, ÖÖB07, JSB07, MLC06, SC06, SY06, Buz06a, Buz06b, STZ05, ÖÇ05a, HGH02]) und vergleicht diese mit der von ihm vorgestellten Idee der geometrischen Methode. Seine Evaluierungen basieren auf mehreren unterschiedlichen Anfrage-Log-Dateien, wobei der Durchschnitt der erreichten Aussage-Sicherheit jeweils für einen Vergleich der verschiedenen Verfahren herangezogen wird. Als das aussagesicherste Verfahren ermittelt Gayo-Avello seine eigene geometrische Methode, welche wir deshalb als Grundlage dieser Arbeit verwenden.

Die geometrische Methode durchläuft sequentiell die nach Zeit sortierten Anfragen eines Benutzers aus der Anfrage-Log-Datei und quantifiziert zwei Merkmale für die aktuell untersuchte Anfrage in Bezug auf die aktuell betrachtete Suchsitzung. Die Suchsitzung beinhaltet eine oder mehrere der vorherigen Suchanfragen mit der Ausnahme des ersten Iterations-Schritts jedes Benutzers, da hier keine vorherige Anfrage vorhanden ist. In dem Fall der Detektierung einer Sitzungsfortführung wird die aktuelle Anfrage der Suchsitzung zugeordnet, und die nächste Suchanfrage des Benutzers mit der Suchsitzung verglichen. Wenn hingegen mit einer Suchanfrage eine neue Suchsitzung detektiert wird, gilt die aktuelle Suchsitzung als beendet. Die untersuchte Suchanfrage wird einer neu generierten Suchsitzung zugeordnet, und die nächste Suchanfrage mit dieser neuen Suchsitzung verglichen.

Die Problemstellung, bei jedem Iterations-Schritt zwischen einer Sitzungsfortführung und einer neuen Suchsitzung zu unterscheiden, löst Gayo-Avello wie folgt: Er ermittelt den zeitlichen Abstand zwischen der aktuell untersuchten Suchanfrage und der letzten Suchanfrage der aktuellen Suchsitzung eines Benutzers. Diesen Abstand setzt er in Relation mit einem lexikalischen Merkmal, das über ein Vektorraummodell die übereinstimmenden n-Gramme der aktuell untersuchten Suchanfrage im Bezug zu den n-Grammen der aktuellen

Suchsitzung in ein Verhältnis setzt. Gayo-Avello spezifiziert jedoch nicht, für welche Werte n er die n-Gramme quantifiziert.

Um diese beiden Merkmale in Relation betrachten zu können, werden diese auf eine gängige Maßzahl mit einem Wert von 0 bis 1 normalisiert. Für das zeitliche Merkmal gilt dabei, dass zwei nahezu zeitgleich abgesendete Suchanfragen den Ähnlichkeits-Wert 1, und zwei Suchanfragen mit einem zeitlichen Abstand größer einem definierten Maximum von 24 Stunden den Wert 0 erhalten. Das lexikalische Merkmal wird aus dem Verhältnis der Anzahl der übereinstimmenden n-Gramme im Bezug zur Gesamtanzahl der enthaltenen n-Gramme bestimmt. Wenn bei der untersuchten Suchanfrage kein einziges n-Gramm mit der aktuellen Suchsitzung überlappt, entspricht das einer lexikalischen Ähnlichkeit von 0. Nur für den Fall, dass alle n-Gramme der Suchsitzung mit allen n-Grammen der untersuchten Anfrage vollständig übereinstimmen, wird ein maximaler Wert von 1 ermittelt[4].

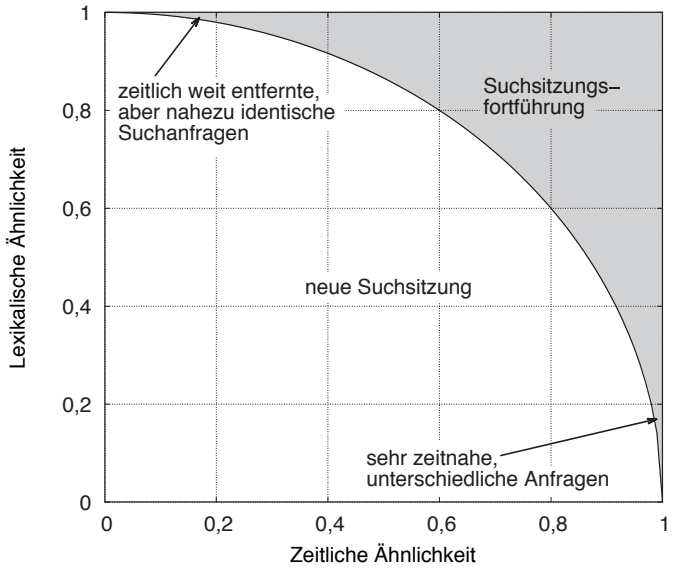

Abbildung 2.1: Klassifizierungsgrundlage der geometrischen Methode

Diese beiden quantifizierten Merkmale werden im $\mathbb{R}^2$ aufgetragen (vgl. Abbildung 2.1). Um bei der Analyse einer Anfrage eines Benutzers zwischen den beiden Fällen, also der Fortführung einer Suchsitzung und dem Beginn einer neuen Suchsitzung zu unterscheiden, wird von Gayo-Avello eine Funktion bestimmt. Er verwendet hierzu die Kreisfunktion mit einem Radius $r = 1$ und

[4]Die von Gayo-Avello [Gay09] definierten Merkmale haben wir invertiert. Wir ziehen eine Betrachtung vor, bei der ähnliche Objekte ein größeres Ähnlichkeitsmaß ergeben, als unähnliche Objekte.

mit dem Mittelpunkt des Kreises im Ursprung. Alle Punkte, die innerhalb des Kreises liegen, haben zu Folge, dass die aktuell untersuchte Anfrage eines Benutzer einer neuen Suchsitzung zugeordnet wird. Alle Punkte der Anfragen, die außerhalb der Funktion liegen, haben eine Sitzungsfortführung zur Folge.

Sun et al. [SYL+09] verwenden die geometrische Methode, um eine Anfrage-Log-Datei der chinesischen Suchmaschine Sogou in Suchsitzungen zu segmentieren. Abgesehen von einigen Abweichungen, die den Charakter des chinesischen Alphabets betreffen, machen sie drei konkrete Verbesserungsvorschläge und erzielen damit bessere Resultate.

Zunächst gewichten sie das zeitliche Merkmal mit $f(t) = 1-((1-t)^{1/k})$ und evaluieren diese Idee für $k = 2, 3, 4$. Diesen Schritt begründen sie damit, dass bei ihren Untersuchungen der Anteil von Suchanfragen mit einem sehr kleinen zeitlichen Abstand überproportional hoch ist. Für das lexikalische Merkmal verwenden Sun et al. alle n-Gramme mit $n \leq m$, wobei $m = 4$. Die einzelnen n-Gramme $w_1 \dots w_n$ gewichten Sun et al. über die Gewichtungs-Funktion

$$w(w_1 \dots w_n) = \begin{cases} \dfrac{1}{1 + 100 \cdot \log\left(\frac{count(w_1 \dots w_{n-1})}{count(w_1 \dots w_n)}\right)} & , n > 1 \\ \dfrac{1}{1 + 100 \cdot \log\left(\frac{total\_length}{count(w_1 \dots w_n)}\right)} & , n = 1 \end{cases}$$

mit dem Ziel, längere und insbesondere neue n-Gramme einer Suchanfrage stärker zu berücksichtigen. Die Funktion $count(w_1 \dots w_n)$ ermittelt das Vorkommen des n-Gramms in der aktuell betrachteten Suchsitzung und der betrachteten Anfrage. Der Parameter *total_length* entspricht der Gesamt-Zeichenanzahl der Suchsitzung und der Anfrage.

Für ihren dritten Verbesserungsvorschlag beziehen sie in ihre Überlegungen die Anzahl der Klicks auf die Resultate der Suchergebnisseite mit ein. Sie argumentieren, dass sehr wenige Klicks auf ein Ende der aktuellen Suchsitzung schließen lassen. Klickt der Benutzer hingegen auf keines der ihm präsentierten Ergebnisse oder aber auf sehr viele, deuten dieses Sun et al. als eine nicht zufriedenstellend beantwortete Anfrage und daher als eine höhere Wahrscheinlichkeit für eine Sitzungsfortführung. Um nun die Detektierung von neuen Suchsitzungen zu erhöhen (falls wenige Klicks erfolgten) vergrößern sie den Radius der Kreisfunktion um $c = \frac{0{,}1}{1+log(click\_count)}$. Hierbei entspricht *click_count* der Anzahl der Ergebnisse, die der Benutzer für eine Suchanfrage aufgerufen hat.

Aufbauend auf Gayo-Avellos Verfahren werden wir verschiedene Ideen evaluieren und versuchen, das geometrische Verfahren zu verbessern. Mit den drei unterschiedlichen von Sun et al. vorgestellten Gewichtsfunktionen ließen sich jedoch keine Verbesserungen erreichen.

2.2.6 Maschinelle Lernverfahren

Die vorliegende Arbeit beschäftigt sich überwiegend mit der Fragestellung, welche „teuer“ zu quantifizierenden Merkmale in welchen bestimmten Fällen zu einem aussagekräftigeren Ergebnis führen. So können wir bei einem Großteil der Suchanfragen darauf verzichten, da hierbei die günstig zu quantifizierenden Merkmale eine ausreichende Aussagesicherheit garantieren.

Viele der verschiedenen maschinellen Lernverfahren, die in der Literatur evaluiert werden, setzen die Quantifizierung aller Merkmale voraus und sind somit für unsere Überlegungen uninteressant. Zudem quantifizieren die meisten der vorgestellten Arbeiten nur den zeitlichen Abstand und ein einfaches lexikalisches Merkmal durch die simple Bedingung, dass mindest ein Term überlappt. Daher beschränken wir uns in dieser Arbeit darauf, diese Verfahren nur kurz aufzulisten.

Die Arbeiten von Özmutlu und Çavdur [ÖÇ05a], sowie He et al. [HGH02] basieren auf der Evidenztheorie nach Dempster und Shafer. Özmutlu et al. [ÖÖB07] ermitteln die Suchsitzung mithilfe einer Monte-Carlo-Simulation. Beiden Verfahren werden von Gayo-Avello mit seiner geometrischen Methode verglichen und als qualitativ schlechter bewertet [Gay09].

Özmutlu und Çavdur [ÖÇ05b] trainieren ein neuronales Netz, um Suchsitzungen zu detektieren. Özmutlu et al. [ÖÖS08] verwenden das Verfahren der multiplen linearen Regression. Die in der Literatur angegebene Genauigkeit dieser beiden Verfahren ist ebenfalls nicht so aussagekräftig wie die Ergebnisse der geometrischen Methode.

Lediglich Jones und Klinkner [JK08] quantifizieren weitere Merkmale über die oben genannten hinaus. Mithilfe des Verfahrens der logistischen Regression experimentieren sie mit vielen verschiedenen Merkmalen der verschiedenen Merkmals-Kategorien. Neben dem zeitlichen Abstand und einfachen lexikalischen Merkmalen, wie zum Beispiel der Levenshtein-Distanz oder die Untersuchung auf bestimmte Termüberlappung, verwenden sie mehrere Anfrage-Log-Datei-Merkmale sowie die Erweiterung durch Teile der Ergebnismenge einer Suchmaschine. Als Ergebnis bleiben je die acht aussagekräftigsten Merkmale differenziert, für die beiden Fälle der Suchsitzungsfortführung und zur Detektierung von neuen Suchsitzungen, übrig. Das Verfahren von Jones und Klinkner ist das einzige in der aktuellen Literatur, welches bessere Ergebnisse erzielt als die geometrische Methode von Gayo-Avello. Allerdings erläutern sie, dass die Quantifizierung dafür sehr aufwendig ist. Aus diesem Grund versuchen wir die geometrische Methode von Gayo-Avello in einem kaskadierenden Verfahren zu erweitern, um diese besonders „kostengünstig“ zu verbessern. Eine Entscheidung für einen ähnlichen Ansatz, wie Jones und Klinkner ihn gewählt haben, würde diese Zielsetzung nicht ermöglichen.

Kapitel 3

Detektierung von Suchsitzungen

3.1 Notation und Basisdefinitionen

Eine Anfrage-Log-Datei einer Suchmaschine ist eine Folge von Nutzeraktionen. Wir betrachten für unsere weiteren Überlegungen alle Aktionen $\mathcal{A}$ eines Benutzers. Dabei protokolliert a_i die i-te Aktion des Benutzers und ist ein Tupel der Form $a_i := \langle u_i, t_i, q_i \rangle$, bestehend aus der Nutzer-ID u_i, einem Zeitstempel t_i mit Datum und Uhrzeit der Aktion und der entsprechenden Suchanfrage q_i. Alle Aktionen, die einen Klick auf ein Ergebnis protokollieren, erweitern dieses 3-Tupel um die Domain d_i und den Rang r_i dieser Webseite innerhalb der präsentierten Ergebnisse. Folglich entspricht eine solche Klick-Aktion einem 5-Tupel von Werten der Form $a_i := \langle u_i, t_i, q_i, d_i, r_i \rangle$. Die Liste $\mathcal{A}$ ist nach der zeitlichen Abfolge der Aktionen, bestimmt durch den Zeitstempel t_i, sortiert. Ziel ist es nun, alle Aktionen $\mathcal{A}$ eines Benutzers nach der Definition von Silverstein et al. [SMHM99] in nicht überlappende Suchsitzungen $\mathcal{S} = s_1, \ldots, s_j, \ldots, s_{|\mathcal{S}|}$ zu gruppieren. Dabei beinhaltet eine Suchsitzung s_j eines Benutzers eine oder mehrere aufeinanderfolgende Aktionen. Somit ist eine Suchsitzung eine Konkatenation von Aktionen in der Form $s_j = a_k \circ \ldots \circ a_i \circ \ldots \circ a_l$ mit $l \geq k > 0$ und $l \leq |\mathcal{A}|$. Alle Aktionen $\mathcal{A}$ eines Benutzers können somit auch als eine Konkatenation seiner Suchsitzungen $\mathcal{A} = s_1 \circ \ldots \circ s_{|\mathcal{S}|}$ aufgefasst werden.

Um die Aktionen in Suchsitzungen zu gruppieren, werden iterativ die Merkmale des Aktionspaares a_{i-1} und a_i quantifiziert. Während jedes Iterationsschritts muss also zwischen je zwei Aktionen entschieden werden, ob diese in zwei unterschiedliche Sitzungen fallen. Die aktuell betrachtete Aktion a_i wird somit entweder der Suchsitzung s_j, in welcher mindestens die Aktion a_{i-1} erfasst ist, oder einer neuen Suchsitzung s_{j+1} zugeordnet.

3.2 Datenbasis

Unseren Experimenten liegt eine Anfrage-Log-Datei der AOL-Suchmaschine aus dem Jahr 2006 zugrunde. Diese beinhaltet 36 Millionen Aktionen von 657 416 unterschiedlichen Benutzern in einem Zeitraum von 3 Monaten. Die Verteilung der Suchaktionen pro Benutzer kann der Abbildung 3.1 entnommen werden. Die Darstellung ist doppelt logarithmisch. Die Skaleninvarianz der Verteilung ähnelt erwartungsgemäß der *power law*-Verteilung. Dieser Verlauf ist darauf zurückzuführen, dass relativ wenige Benutzer relativ viele Aktionen, und relativ viele Benutzer relativ wenige Aktionen ausgeführt haben.

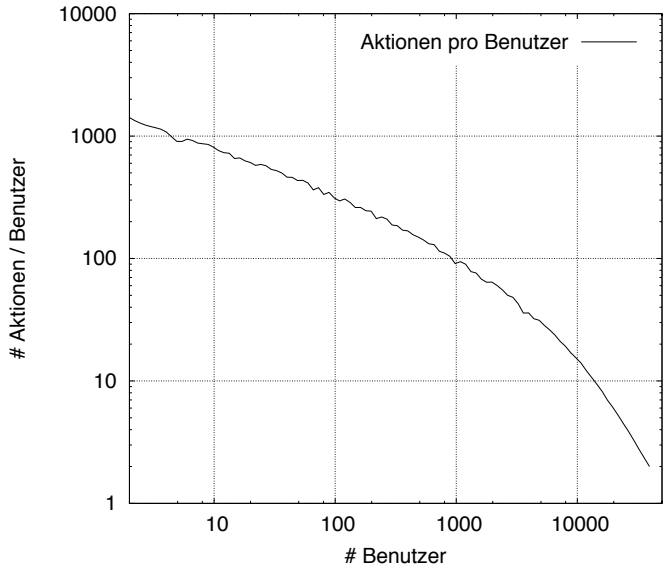

Abbildung 3.1: Verteilung der Anzahl Suchaktionen pro Benutzer

Eine Teilmenge dieser Anfrage-Log-Datei von 11.484 Aktionen von 214 unterschiedlichen Benutzern wurde von Gayo-Avello [Gay09] vorsegmentiert und steht uns zur Analyse zur Verfügung. Bei dieser manuellen Annotierung wurden anteilig nicht alle ursprünglichen Aktionen eines Benutzers erfasst. Allerdings ähnelt diese Teilmenge, im Sinne der Verteilung, den Ursprungsdaten. Die manuell segmentierten Suchsitzungen beinhalten durchschnittlich 2,7 Suchaktionen. Mit dieser Teilmenge werden wir unsere Experimente durchführen, um die Güte unseres Verfahrens zu prüfen. Eigentlich ist aber das Ziel, mit unserem Verfahren die gesamte Anfrage-Log-Datei zu segmentieren. Für diesen Schritt ist eine vorherige Datenaufbereitung nötig. Zum Beispiel haben 73.764 Benutzer im gesamten Untersuchungszeitraum nur eine einzige Suchanfrage abgesendet und sind somit für unsere Fragestellung uninteressant.

3.3 Datenaufbereitung

Ziel der Datenaufbereitung ist die Entfernung von unerwünschten Daten, um nur die von realen Menschen gestellten Suchanfragen zu berücksichtigen (vgl. Abschnitt 2.1). Des Weiteren werden im Vorfeld alle Aktionen entfernt, die für eine Detektierung von Suchsitzungen nicht relevant sind.

Für die weitergehenden Überlegungen betrachten wir jeweils alle einzelnen Aktionen a eines Benutzers. Somit werden alle Suchanfragen sowie auch alle Klicks auf die Webseiten der Ergebnisliste erfasst.

Von einer vorherigen Gruppierung der Aktionen in Form von *search episodes* über einem Intervall von 24 Stunden, wie es häufig in der Literatur angewendet wird [Gay09, JS06, JSB07, SYL+09], sehen wir ab. Um eine zeitliche Betrachtung im Verhältnis zur Aktionsanzahl eines Benutzer zu normieren, entfernen wir größere Aktivitätspausen. Der zeitliche Abstand zweier aufeinanderfolgender Aktionen wird hierzu durch

$$f_{\text{offset}}(a_i) = t_i - t_{i-1}$$

ermittelt. Falls der zeitliche Abstand einen Schwellwert überschreitet, wird dieser von der Gesamtzeitspanne der Benutzeraktivität $(t_{|\mathcal{A}|} - t_1)$ subtrahiert. Wenn wir die Summe aller in Frage kommenden zeitlichen Abstände ermitteln, erhalten wir einen Zeitraum, der die reale Nutzungszeit geeignet widerspiegelt:

$$f_{\text{activity}}(\mathcal{A}) = (t_{|\mathcal{A}|} - t_1) - \sum_{i=2}^{|\mathcal{A}|} \begin{cases} f_{\text{offset}}(a_i) & , f_{\text{offset}}(a_i) \geq T \\ 0 & , f_{\text{offset}}(a_i) < T \end{cases}$$

Dabei bestimmt der Schwellwert T den maximalen Abstand zwischen zwei aufeinanderfolgenden Aktionen, sodass dieser als aktiver Zeitraum mit berücksichtigt wird. Wir wählen für unsere Experimente einen maximalen Abstand von $T = 6$ Stunden, sodass alle längeren Aktivitätspausen abgezogen werden. Dieser relativ hohe Schwellwert begründet sich dadurch, dass es für eine unter den Benutzern vergleichbare Quantifizierung genügt, lediglich längere Abstände hierfür zu berücksichtigen.

Um schließlich die Aktionen zu entfernen, die als nicht-menschliche Anfrage-Quellen einzuordnen sind, werden in der Literatur zwei mögliche Eigenschaften beschrieben: Zum einen ein festgelegtes Maximum an Aktionen im Bezug zu einer Zeitspanne (zum Beispiel eines Tages) [Buz07, Gay09], zum anderen ein durchschnittlicher, minimaler Abstand zwischen zwei aufeinanderfolgenden Aktionen [DF09]. Da beide Werte durch ein Verhältnis zwischen einer zeitlichen Größe und der Anfrageanzahl ermittelt werden, erweisen sich beide

Eigenschaften als äquivalent. Daher ist nur die Quantifizierung einer dieser beiden Regeln nötig. Wir quantifizieren den durchschnittlichen Abstand zwischen allen Aktionen des Benutzers mit

$$f_{\text{avg\_offset}}(\mathcal{A}) = \frac{f_{\text{activity}}(\mathcal{A})}{|\mathcal{A}|}$$

und erhalten somit das Verhältnis der Anzahl der abgesendeten Aktionen zu einer angenäherten Nutzungszeit. Alle Benutzer, die einen bestimmten Schwellwert für $f_{\text{avg\_offset}}(\mathcal{A})$ unterschreiten, werden entfernt.

Mit der Zielsetzung, die Anfrage-Log-Datei in möglichst lange Suchsitzungen hoher Güte zu segmentieren, möchten wir auch die Benutzer entfernen, die nur sporadisch gesucht haben. Diese sind für die Detektierung von Suchsitzungen irrelevant. Durch die beschriebene Subtraktion der Aktivitätspausen von der Gesamtzeit entfernen wir mit dem $f_{\text{avg\_offset}}(\mathcal{A})$ Ansatz ebenfalls diese Benutzer, die zwar nicht unbedingt ein exzessives Interaktions-Verhalten aufweisen, aber ihre Aktion nur vereinzelt mit größeren Aktivitäts-Pausen abgesendet haben. Somit entfernen wir mit unserem Ansatz basierend auf $f_{\text{activity}}(\mathcal{A})$ deutlich mehr Aktionen, als wenn wir stattdessen die Gesamtzeitspanne $(t_{|\mathcal{A}|} - t_1)$ zugrunde legen würden. Die unterschiedliche Verteilung der aussortierten Aktionen je Benutzer, entsprechend den beiden Varianten, kann der Abbildung 3.2 entnommen werden. Das Diagramm ist einfach logarithmisch und verdeutlicht die unterschiedlichen Quantitäten der beiden Verteilungen für die gesamte AOL Anfrage-Log-Datei.

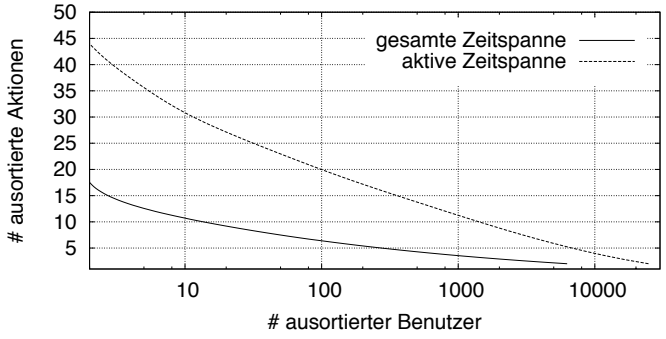

Abbildung 3.2: Verteilung aussortierter Aktionen je Benutzer für die gesamte Zeitspanne $(t_{|\mathcal{A}|} - t_1)$ im Vergleich zur aktiven Zeitspanne $f_{\text{avg\_offset}}(\mathcal{A})$

Nach Duskin und Feitelson [DF09] ist es für einen erheblichen Teil der Aktionen schwierig, mit Sicherheit zwischen menschlichen und nicht-menschlichen Quellen zu unterscheiden. Aufbauend auf dieser Argumentation wählen wir unsere Schwellwerte entsprechend, sodass infrage kommende Aktionen großzügig

entfernt werden. Zum Beispiel interpretieren wir das lokale Maximum in Abbildung 3.3 bei 2 Sekunden als ein klares Indiz für nicht menschliche Quellen. Das Diagramm zeigt für die jeweiligen Schwellwerte die durchschnittliche Anzahl an Aktionen je aussortierter Benutzer nach $f_{\text{avg\_offset}}(\mathcal{A})$. Ein Schwellwert bei 2 Sekunden betrifft allerdings nur einen sehr geringen Teil in Höhe von 0,7% der gesamten Aktionen. Um aber deutlich mehr Aktionen zu entfernen, wählen wir für $f_{\text{avg\_offset}}(\mathcal{A})$ einen entsprechend höheren Schwellwert in Höhe von 10 Sekunden. Hierbei ist die durchschnittliche Aktionsanzahl der aussortieren Benutzer entsprechend geringer.

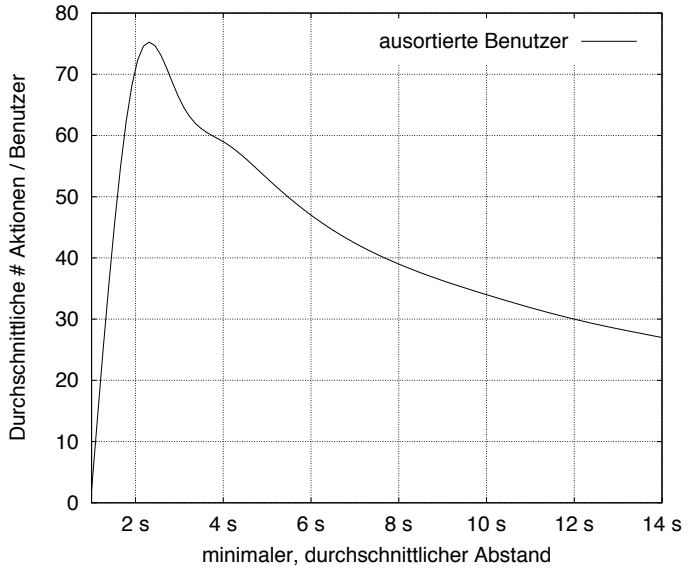

Abbildung 3.3: Verlauf der durchschnittlichen Aktionsanzahl der aussortierten Benutzer nach verschiedenen Schwellwerten für den minimalen durchschnittlichen Zeitabstand $f_{\text{avg\_offset}}(\mathcal{A})$

Falls ein Benutzer als eine nicht-menschliche oder als eine irrelevante Quelle klassifiziert wird, entfernen wir gleich alle Aktionen $\mathcal{A}$ des Benutzers. Gayo-Avellos Ansatz [Gay09], nur den Teil der in den betrachteten *search episodes* enthaltenen Aktionen eines Benutzers zu entfernen, ist uns nicht konsequent genug. Die Verteilung der aus dem gesamten AOL Anfrage-Log aussortierten Aktionen je Benutzer, basierend auf Gayo-Avellos Ansatz (vgl. Abbildung 3.4), zeigt deutlich, dass bei seiner Idee der überwiegende Teil der Nutzer nur partiell entfernt wird.

Neben der beschriebenen Filterregel $f_{\text{avg\_offset}}(\mathcal{A}) < 10$ Sekunden müssen weitere Regeln definiert werden. In einem ersten Schritt ignorieren wir, wie bereits weiter vorn erwähnt, alle Benutzer mit nur einer Aktion, da für die-

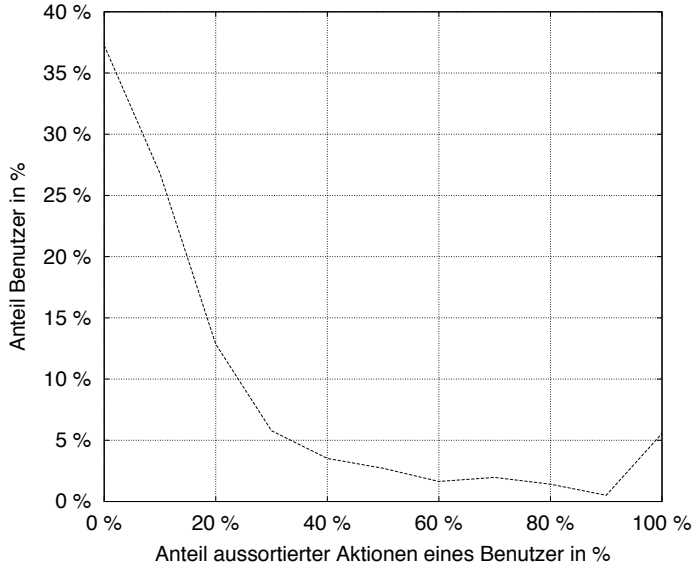

Abbildung 3.4: Anteil aussortierter Aktionen je Benutzer auf Basis von *search episodes* [Gay09]

se keine Suchsitzungen mit mehreren Aktionen entstehen können. Nach der Filterregel $f_{\text{avg\_offset}}(\mathcal{A}) < 10$ Sekunden entfernen wir abschließend alle Benutzer, für die der Median der Anfragelänge mehr als 100 Zeichen aufweist, da solche Benutzer auch eher als nicht-menschliche Quellen zu klassifizieren sind. Bei einer durchschnittlichen Länge in der englischen Sprache von 4,5 Zeichen je Wort, entsprechen 100 Zeichen einer Länge von ungefähr 20 Wörtern, was wir für das Verhalten eines menschlichen Benutzer als unrealistisch erachten. Für eine nach der Zeichenlänge sortierte Liste $\mathcal{Q} = \langle q_1, \ldots, q_n \rangle$ der Anfragen aller Aktionen eines Benutzers wird der Median der Anfragelänge wie folgt ermittelt:

$$f_{\text{med\_chars}}(\mathcal{Q}) = \begin{cases} count\_chars(q_{\frac{n+1}{2}}) & , n \text{ ungerade} \\ count\_chars(q_{\frac{n}{2}}) & , n \text{ gerade} \end{cases}$$

Dabei zählt die Funktion $count\_chars(q)$ die Zeichen der Anfragen. Der entsprechende Anteil aussortierter Aktionen (vgl. Tabelle 3.1) ist allerdings sehr klein.

Einige Anfragen enthalten offensichtliche Überreste einer Kodierung, zum Beispiel in der Form:

> johnson 20county 20community 20college 20kansas

Wir entfernen jeweils die übriggebliebenen Zeichenketten „20“ und verwenden diese Anfragen im weiteren Prozess unberücksichtigt dieser Korrektur.

Alle Eigenschaften, die wir für das Entfernen von Benutzern aus der AOL Anfrage-Log-Datei berücksichtigen, sind in der Tabelle 3.1 aufgeführt. Wir entfernen insgesamt 959.641 Aktionen (2,64%) und 130.292 Benutzer (21,7%).

	Beschreibung	Definition	% $\mathcal{A}$	% $\mathcal{U}$
1.	nur eine Aktion	$\lvert\mathcal{A}\rvert = 1$	0,16 %	9,49 %
2.	im Durchschnitt weniger als 10 Sekunden zwischen je zwei Aktionen	$f_{\text{avg\_offset}}(\mathcal{A}) < 10s$, $T = 6h$	2,48 %	12,17 %
3.	Median der Anfragenlänge größer als 100	$f_{\text{med\_chars}}(\mathcal{Q}) > 100$	< 0,01 %	0,04 %
		Gesamt:	2,64 %	21,70 %

Tabelle 3.1: Verwendete Filter-Regeln, um Benutzer zu entfernen und deren jeweils entfernter Anteil aus der AOL Anfrage-Log-Datei

3.4 Kaskadierendes Verfahren

Das von uns entwickelte kaskadierende Verfahren soll in mehreren Schritten die höchstmögliche Aussagesicherheit auf Basis der einzelnen Merkmale ermöglichen, ohne dabei unnötig viele Merkmale zu quantifizieren. Um zwischen der Güte der Ergebnisse und den Kosten, um diese zu ermitteln, ein ausgewogenes Verhältnis zu erreichen, werden die einzelnen Schritte in der Reihenfolge ihres Laufzeitverhaltens ausgeführt. Wenn uns die ermittelte Ähnlichkeit eines der Merkmale eine hohe Aussagesicherheit bei der Unterscheidung zwischen einer Suchsitzungsfortführung und dem Beginn einer neuen Sitzung ermöglicht, bricht die Quantifizierung der weiteren Merkmale für dieses Anfragepaar daher ab.

Mit einem ersten Schritt ermitteln wir einen Teil der einfacheren Fälle der Wiederholung, Generalisierung und Spezialisierung (vgl. Abschnitt 2.2.2) einer vorangegangenen Suchanfrage. Wenn eine der beiden Suchanfragen die andere vollständig beinhaltet und deren zeitlicher Abstand einen Schwellwert von 30 Minuten nicht überschreitet, können wir mit sehr hoher Sicherheit aussagen, dass diese in einer Sitzung zusammenfallen. Da für diese beiden Suchanfragen mit der nachfolgenden Merkmalsquantifizierung der geometrischen Methode ebenso das gleiche Resultat zu erwarten ist, führen wir die Güte dieser ersten Quantisierung nicht gesondert auf. Durch diesen Schritt werden 39,8% aller Anfragenpaare der annotierten Teilmenge berücksichtigt, sodass für diese keine weiteren Merkmale nötig sind.

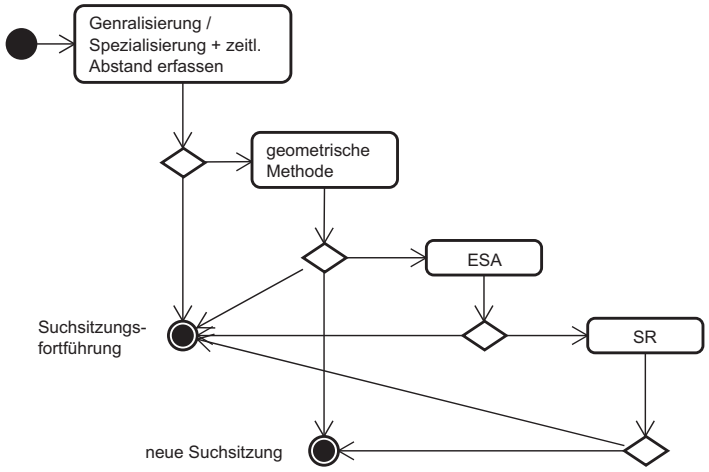

Abbildung 3.5: Idee des kaskadierenden Verfahrens

Die einzelnen Schritte unseres kaskadierenden Verfahrens sind in der Abbildung 3.5 skizziert. Nach dem erläuterten ersten Schritt, der die einfacheren Fälle der Generalisierung und Spezialisierung ermittelt, quantifizieren wir das Ähnlichkeitsmaß nach der geometrischen Methode. Wenn wir für bestimmte Anfragenpaare eine nicht ausreichende Aussagesicherheit bestimmen können, werden wir mit der expliziten semantischen Analyse (*ESA*) und der Erweiterung der Anfragen über Ergebnisse einer Suchmaschine versuchen, diese zu erhöhen. Wir betrachten die Schritte sequentiell und analysieren die jeweils erreichte Güte anhand der folgenden standardisierten Maßzahlen:

$$N_{\text{true\_shift}} \mathrel{\widehat{=}} \text{Anzahl Sitzungsgrenzen der Trainingsmenge.}$$

$$N_{\text{true\_cont}} \mathrel{\widehat{=}} \text{Anzahl Sitzungsfortführungen der Trainingsmenge.}$$

$$N_{\text{shift}} \mathrel{\widehat{=}} \text{Anzahl Sitzungsgrenzen der Hypothese.}$$

$$N_{\text{shift\&correct}} \mathrel{\widehat{=}} \text{Anzahl übereinstimmender Sitzungsgrenzen.}$$

$$N_{\text{Type-A-errors}} \mathrel{\widehat{=}} \text{Anzahl falsch hinzugefügter Sitzungsgrenzen.}$$

$$N_{\text{Type-B-errors}} \mathrel{\widehat{=}} \text{Anzahl falsch entfernter Sitzungsgrenzen.}$$

Diese werden dazu verwendet, die im Information Retrieval weitverbreiteten Güte-Maße

$$precision = \frac{N_{\text{shift\&correct}}}{N_{\text{shift}}}$$

$$recall = \frac{N_{\text{shift\&correct}}}{N_{\text{true\_shift}}}$$

$$F\text{-Measure}_{\beta} = \frac{(1+\beta^2) \cdot precision \cdot recall}{\beta^2 \cdot precision + recall}$$

zu ermitteln. Die Anzahl der falsch hinzugefügten oder entfernten Sitzungsgrenzen werden von den obigen Maßen allerdings unterbewertet, sodass wir zudem die beiden Fehler-Maße *error-per-response-fill* (*ERR*) und *slot error rate* (*SER*) [MKSW99] bestimmen:

$$ERR = \frac{N_{\text{true\_shift}} + N_{\text{shift}} - 2 \cdot N_{\text{shift\&correct}}}{N_{\text{true\_shift}} + N_{\text{shift}} - N_{\text{shift\&correct}}}$$

$$SER = \frac{N_{\text{true\_shift}} + N_{\text{shift}} - 2 \cdot N_{\text{shift\&correct}}}{N_{\text{true\_shift}}}$$

Des Weiteren bestimmen wir die allgemeine Genauigkeit (engl. *accuracy*) für die beiden Fälle einer Sitzungsfortführung ACC_{cont}, einer Sitzungsgrenze ACC_{shift} und deren Durchschnitt ACC_{avg}:

$$ACC_{\text{cont}} = 1 - \frac{N_{\text{Type-A-errors}}}{N_{\text{true\_cont}}}$$

$$ACC_{\text{shift}} = 1 - \frac{N_{\text{Type-B-errors}}}{N_{\text{true\_shift}}}$$

$$ACC_{\text{avg}} = \frac{N_{\text{true\_shift}} \cdot ACC_{\text{shift}} + N_{\text{true\_cont}} \cdot ACC_{\text{cont}}}{N_{\text{true\_shift}} + N_{\text{true\_cont}}}$$

Für unsere Experimente bestimmen wir als das maßgebliche Güte-Maß $F\text{-Measure}_{\beta=1,5}$, da wir falsch zusammengefügte Suchsitzungen als ein sehr viel höheres Problem ansehen, als fälschlich eingefügte Suchsitzungsgrenzen. Durch eine angestrebte Optimierung des F-Measure mit $\beta = 1{,}5$ werden

wir zwar kürzere Suchsitzungen ermitteln, aber dafür mit der höheren Aussagesicherheit, dass alle Aktionen einer Suchsitzung auch tatsächlich zusammen in diese Sitzung gehören.

3.4.1 Zeitlicher Abstand zweier Anfragen

Die Verwendung eines statischen, zeitlichen Merkmals zur Unterscheidung zwischen Sitzungsgrenze und Sitzungsfortführung ist simpel. Wenn der ermittelte Abstand $f_{\text{offset}}(a_i)$ zwischen den aufeinanderfolgenden Aktionen a_{i-1} und a_i einen Schwellwert T überschreitet, beginnt bei a_i eine neue Suchsitzung. Andernfalls wird diese Aktion a_i zur Suchsitzung von a_{i-1} hinzugefügt.

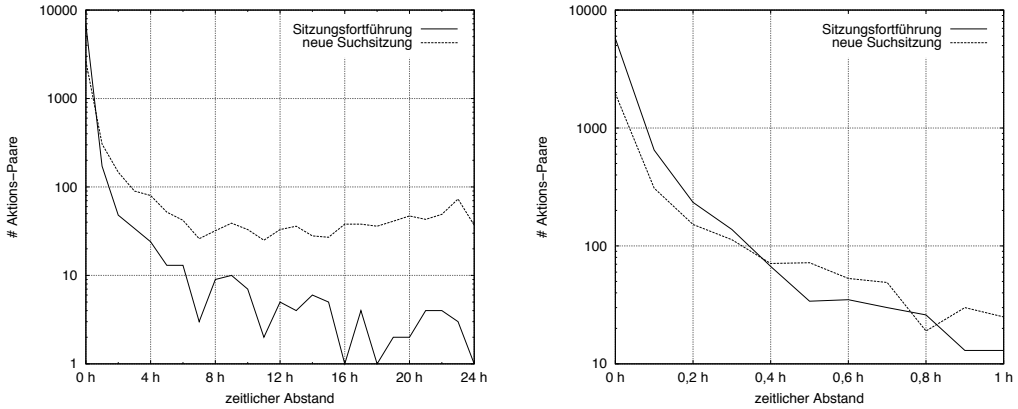

Abbildung 3.6: Verteilung der zeitlichen Ähnlichkeit $f_{\text{time}}(a_i)$

Um das Diskriminanzverhalten dieses Merkmals zu untersuchen, normalisieren wir die Abstände der Aktionen zu einem möglichem Maximum von 24 Stunden. Das heißt, dass alle Aktionspaare mit einem Abstand kleiner als 24 Stunden ein Ähnlichkeitsmaß in Höhe von

$$f_{\text{time}}(a_i) = \frac{f_{\text{offset}}(a_i)}{24\ \text{h}}$$

ergeben. Alle Aktionspaare mit einem größeren Abstand erhalten für das Ähnlichkeitsmaß $f_{\text{time}}(a_i) = 0$.

Die Verteilungen in Abbildung 3.6 basieren auf der von Gayo-Avello [Gay09] manuell annotierten Teilmenge der AOL Anfrage-Log-Datei. Dabei sind alle Aktionspaare separat für die Klassen „neue Suchsitzungen“ und „Sitzungsfortführung“ aufgetragen. Es lässt sich erkennen, dass dieses Merkmal nur eine begrenzte Aussagesicherheit besitzt. Unsere Experimente bestätigen nochmals

die bereits in der Literatur [JK08, MLC06] erläuterte Schwäche des zeitlichen Merkmals als alleiniges Merkmal für eine Entscheidung. Die Ergebnisse für verschiedene Schwellwerte T können den Tabellen 3.2 und 3.3 entnommen werden.

	precision	*recall*	$F_{\beta=1}$	$F_{\beta=1.5}$	*ERR*	*SER*
$T = 1$ h	0,91156	0,68903	0,78483	0,74499	0,35414	0,37782
$T = 30$ min	0,88232	0,73880	0,80420	0,77772	0,32747	0,35974
$T = 15$ min	0,80669	0,82347	0,81500	0,81824	0,31224	0,37386
$T = 5$ min	0,75448	0,87571	0,81059	0,83446	0,31850	0,40926

Tabelle 3.2: Güte des zeitlichen Merkmals $f_{\text{offset}}(a_i)$

	ACC_{shift}	ACC_{cont}	ACC_{avg}
$T = 1$ h	0,68903	0,96373	0,892258
$T = 30$ min	0,73880	0,94654	0,89249
$T = 15$ min	0,79351	0,91672	0,88466
$T = 5$ min	0,87571	0,84540	0,85329

Tabelle 3.3: Genauigkeit des zeitlichen Merkmals $f_{\text{offset}}(a_i)$

Zusätzlich zu einem statischen, zeitlichen Merkmal experimentieren wir mit dem von Murray et al. [MLC06] (vgl. Abschnitt 2.1) vorgestellten *hierarchical agglomerative clustering* (*HAC*). Das *HAC* ermittelt über die Verteilung der zeitlichen Abstände eines Benutzers einen Schwellwert, der für diesen Benutzer als Entscheidungsgrundlage dient. Zusätzlich definieren wir mit HAC_{max} und mit HAC_{min} eine Ober- und Unterschranke für den individuell ermittelten zeitlichen Abstand. Leider konnte keines der Ergebnisse überzeugen. Eine ähnliche Güte, im Vergleich zu einem systemorientierten Schwellwert, wurde nur erreicht, wenn der Zeitraum zwischen HAC_{min} und HAC_{max} vergleichsweise klein gewählt wurde. Damit stimmen unsere Untersuchungen mit denen von Gayo-Avello [Gay09] insofern überein, als dass mit dem Verfahren von Murray et al. [MLC06] keine besseren Ergebnisse erreicht werden können.

3.4.2 Lexikalische Ähnlichkeit

Die lexikalischen Ähnlichkeitsmaße basieren auf dem Anfrage-Inhalt q der Aktionen, wobei die beiden Zeichenketten zweier aufeinanderfolgender Aktionen in der Literatur auf unterschiedliche Weise miteinander verglichen werden. Ein einfach zu quantifizierendes Merkmal ist die Termüberlappung in der Form

$$f_{\text{termOver}}(a_i) = \frac{count\_terms\_in\_common(q_{i-1}, q_i)}{count\_terms(q_i)},$$

wobei $count\_terms\_in\_common(q, q')$ die übereinstimmenden Terme der Anfragen q und q_{prime} zählt. Die Funktion $count\_terms(q)$ ermittelt die jeweilige Term-Anzahl der Zeichenkette q. Ein Ähnlichkeitsmaß basierend auf der Levensthein-Distanz ist naheliegend, da eine anteilige Umformulierung von Suchanfragen auf eine Suchsitzungsfortführung schließen lässt (vgl. Abschnitt 2.2.2). Die Funktion

$$f_{\text{levDist}}(a_i) = 1 - \frac{levDistance(q_{i-1}, q_i)}{count\_chars(q_{i-1}) + count\_chars(q_i)}$$

ermittelt dabei ein normalisiertes Ähnlichkeitsmaß. Dabei bestimmt die Funktion $levDistance(q, q')$ die eigentliche Levenshtein-Distanz und $count\_chars(q)$ die Anzahl der jeweiligen Zeichen.

Für dieses Ähnlichkeitsmaß ist nur die Quantifizierung eines Aktionspaars (a_{i-1}, a_i) sinnvoll. Für die Termüberlappung, sowie für die nachfolgenden lexikalischen Ähnlichkeitsmaße, wurde alternativ auch mit der Quantifizierung aller Aktionen $a \in s_j$ der aktuell betrachteten Suchsitzung (anstatt nur der Aktion a_{i-1}) experimentiert und den Ergebnissen der ursprünglichen Betrachtung auf Basis der Aktion a_{i-1} gegenübergestellt. Für die Termüberlappung erreicht die Quantifizierung gegenüber der Aktion a_{i-1}, im Vergleich zu der Quantifizierung gegenüber der gesamten Suchsitzung, die bessere Güte. Für die beiden anderen Ähnlichkeitsmaße ist hingegen der Vergleich der Suchsitzung mit der aktuell betrachteten Aktion aussagekräftiger. Diese beiden weiteren Ähnlichkeitsmaße der Literatur basieren auf n-m-Grammen. Zum einen wird der Jaccard-Koeffizient bestimmt,

$$f_{\text{jaccKoef}}(s_j, a_i) = \frac{count\_in\_common(ngramm_{n,m}(s_j), ngramm_{n,m}(a_i))}{\|ngramm_{n,m}(a_i)\|}$$

zum anderen die Kosinusähnlichkeit:

$$f_{\text{cosSim}}(s_j, a_i) = \frac{ngramm_{n,m}(s_j)^T \cdot ngramm_{n,m}(a_i)}{\|ngramm_{n,m}(s_j)\| \cdot \|ngramm_{n,m}(a_i)\|}$$

Mit $ngramm_{n,m}(...)$ werden alle möglichen n- bis m-Gramme einer Aktion oder Sitzung in Form eines Vektors ermittelt. Übereinstimmenden n-m-Gramme zählt die Funktion $count\_in\_common(ngramm_{n,m}(s_j), ngramm_{n,m}(a_i))$.

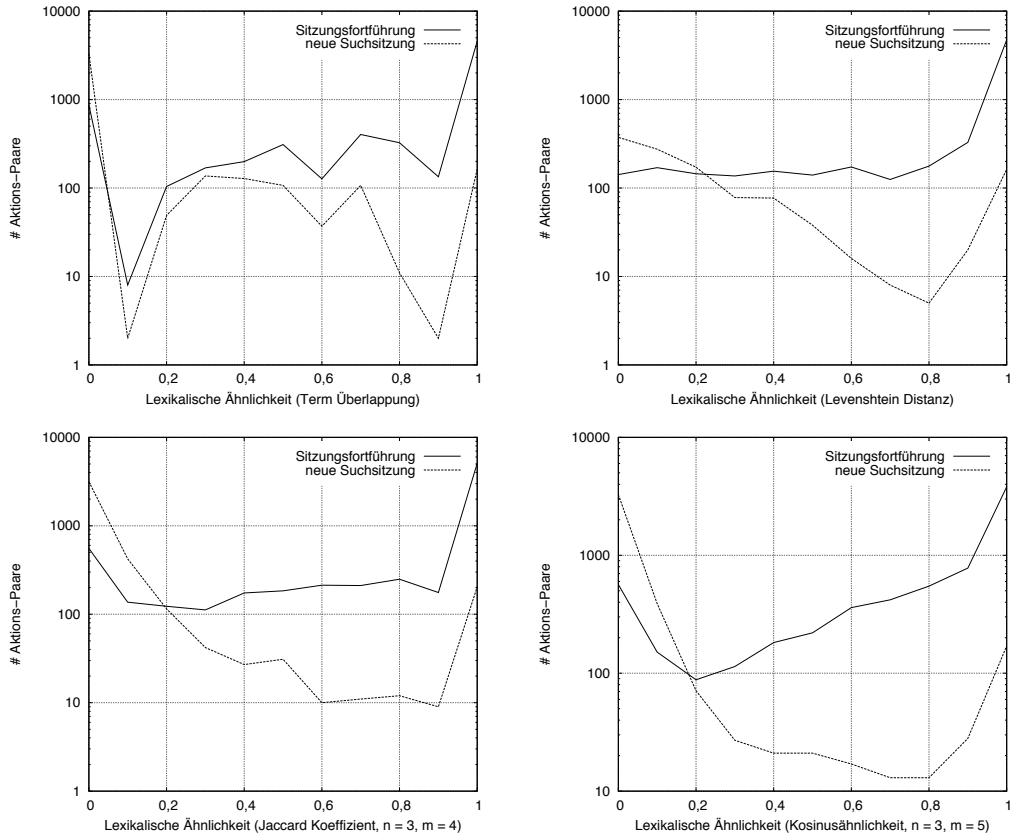

Abbildung 3.7: Ähnlichkeits-Verteilungen der verschiedenen lexikalischen Merkmale

Wir haben für verschiedene Werte n und m jeweils die beiden Merkmale $f_{\text{jaccKoef}}(s_j, a_i)$ und $f_{\text{cosSim}}(s_j, a_i)$ quantifiziert und gegenübergestellt. Zudem wurde für alle Ähnlichkeitsmaße ein angenähertes Optimum für die jeweiligen Schwellwerte durch Stichproben ermittelt. Die Termüberlappung diskriminiert am besten bei einem Schwellwert von 0,6. Das heißt, dass mindestens 60% der Terme übereinstimmen müssen, sodass diese Aktionen in einer Suchsitzung zusammenfallen. Für das auf der Levensthein-Distanz basierende Ähnlichkeitsmaß ermitteln wir einen Schwellwert von 0,5. Von allen probierten Längen $n, m = 1, \ldots, 20$ mit $m \geq n$ erreichte der Jaccard-Koeffizient der n-m-Gramme das beste Ergebnis mit $n = 3, m = 4$ bei einem Schwellwert von 0,3. Für die Kosinusähnlichkeit sind 3 bis 5-Gramme und ein Schwellwert von 0,25 die Parameter mit der höchsten Güte. Die Verteilungen auf Basis der von Gayo-Avello [Gay09] manuell annotierten Teilmenge der AOL Anfrage-Log-Datei für die verschiedenen Ähnlichkeitsmaße sind in Abbildung 3.7 gegenübergestellt. Die jeweils bestmöglichen Ergebnisse sind in den Tabellen 3.4

und 3.5 aufgeführt.

	precision	*recall*	$F_{\beta=1}$	$F_{\beta=1.5}$	*ERR*	*SER*
$f_{\text{termOver}}(a_i)$	0,68081	0,92944	0,78593	0,83555	0,35265	0,50631
$f_{\text{levDist}}(a_i)$	0,70431	0,94182	0,80593	0,85328	0,32505	0,45358
$f_{\text{jaccKoef}}(s_j, a_i)$	0,81197	0,92053	0,86285	0,88415	0,24122	0,29265
$f_{\text{cosSim}}(s_j, a_i)$	0,82310	0,92275	0,87008	0,88961	0,22996	0,27556

Tabelle 3.4: Güte der lexikalischen Merkmale

	ACC_{shift}	ACC_{cont}	ACC_{avg}
$f_{\text{termOver}}(a_i)$	0,92944	0,76360	0,80675
$f_{\text{levDist}}(a_i)$	0,94182	0,78549	0,82617
$f_{\text{jaccKoef}}(s_j, a_i)$	0,92053	0,88435	0,89376
$f_{\text{cosSim}}(s_j, a_i)$	0,92275	0,89241	0.90031

Tabelle 3.5: Genauigkeit der lexikalischen Merkmale

3.4.3 Geometrische Methode

Die geometrische Methode kombiniert das lexikalisches Merkmal $f_{\text{cosSim}}(s_j, a_i)$ und das zeitliche Merkmal $f_{\text{time}}(a_i)$. Diese beiden Ähnlichkeitsmaße in der Form des Vektors

$$v = \begin{pmatrix} f_{\text{cosSim}}(s_j, a_i) \\ f_{\text{time}}(a_i) \end{pmatrix}$$

bilden die Entscheidungsgrundlage für die geometrische Methode. Eine Länge des Vektors $v \geq 1$ entspricht dabei einer Abtragung der Werte im $\mathbb{R}^2$ außerhalb des Einheitskreises und laut [Gay09] einer Sitzungsfortführung. Ein resultierender Vektor mit einer Länge < 1 liegt innerhalb des Einheitskreises und entspricht einer neuen Suchsitzung (vgl. Abbildung 3.8). Somit gilt für die Ermittlung des Ähnlichkeitsmaßes der geometrischen Methode

$$f_{\text{geom}}(s_j, a_i) = \begin{cases} \sqrt{f_{\text{cosSim}}(s_j, a_i)^2 + f_{\text{time}}(a_i)^2} & , \ |s_j| > 0 \\ 1 & , \ |s_j| = 0\,, \end{cases}$$

wobei der Fall $|s_j| = 0$ immer nur für den ersten Iterations-Schritt eines Benutzers gilt. Bei einem Wert für $f_{\text{geom}}(s_j, a_i) \geq 1$ wird die Aktion a_i der Suchsitzung s_j zugeordnet. Falls der ermittelte Wert unter 1 fällt, beginnt mit der

Aktion a_i eine neue Suchsitzung s_{j+1}. Für den nächsten Iterationsschritt wird somit in diesem Fall, neben dem obligatorischen Heraufzählen der Laufvariable i, die Laufvariable j der aktuell betrachteten Suchsitzung um 1 erhöht.

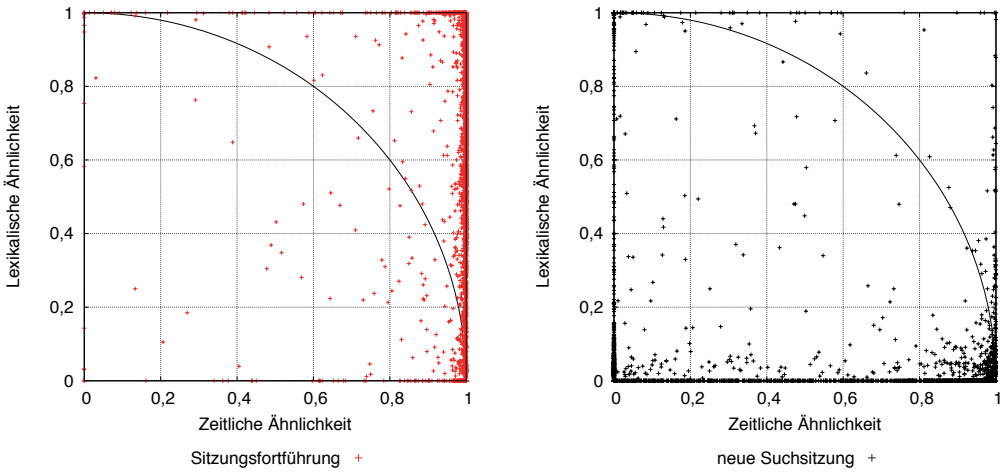

Abbildung 3.8: Verteilung der lexikalischen Ähnlichkeit im Verhältnis zur zeitlichen Ähnlichkeit

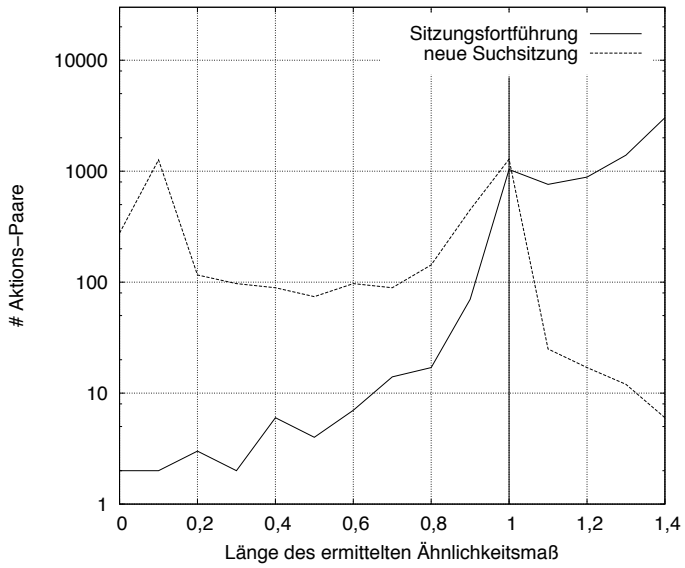

Abbildung 3.9: Verteilung der Entscheidungsgrundlage der geometrischen Methode

In den Diagrammen in Abbildung 3.8 sind für alle Aktionen das jeweilige lexikalische dem zeitlichen Ähnlichkeitsmaß gegenübergestellt. Die beiden

Klassen, Suchsitzungsfortführung (rot) und neue Suchsitzung (schwarz), sind separat aufgeführt, um bei möglichen Überlagerungen der Punktwolken eine fehlerfreie Darstellung zu ermöglichen.

Die Verteilung der ermittelten Ähnlichkeit nach $f_{\text{geom}}(s_j, a_i)$ kann der Abbildung 3.9 entnommen werden. Das lokale Maximum der Verteilung beider Klassen bei 1 ist darauf zurückzuführen, dass in vielen Fällen eines der beiden zugrunde liegenden Ähnlichkeitsmaße zum Maximum 1, das andere Maß dabei hingegen gegen 0 tendiert. Wie auch Abbildung 3.8 zeigt, ist der erheblich größere Anteil dabei zeitlich sehr nah, aber lexikalisch kaum ähnlich.

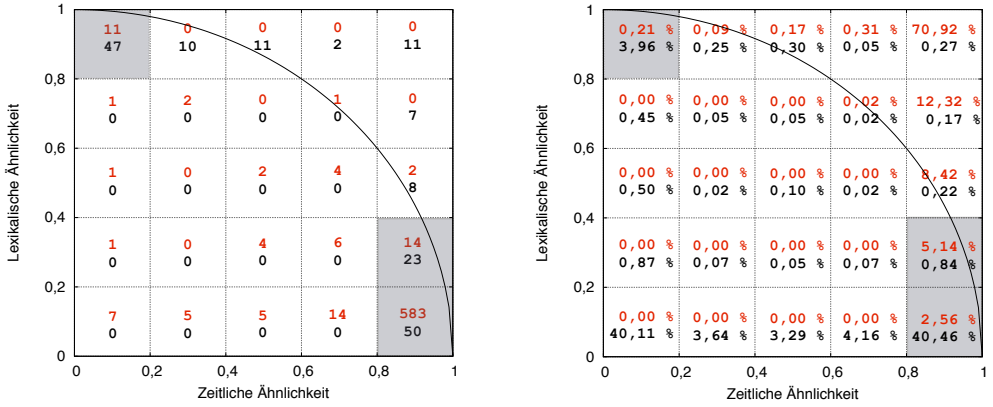

Abbildung 3.10: # Falsch hinzugefügte / entfernte Sitzungsgrenzen im Vergleich zur prozentualen Verteilung

Die Diagramme in Abbildung 3.10 veranschaulichen diesen Zusammenhang im Detail. Beide Diagramme sind partiell in Quadranten unterteilt. Im linken Diagramm ist die Anzahl der fehlerhaften klassifizierten Sitzungsfortführungen sowie die Anzahl der fehlerhaften klassifizierten Suchsitzungsgrenzen je Quadrant abgetragen. Das rechte Diagramm veranschaulicht den jeweiligen Anteil aller entsprechend quantifizierten Aktionen in Prozent.

In den oben links markierten Eckbereich in Abbildung 3.10 fallen nur wenige Anfragen. Unsere Untersuchungen haben gezeigt, dass wir das bestmögliche Ergebnis für diesen Teil der Aktionen ermitteln können, wenn wir alle Aktionen mit einer zeitlichen Ähnlichkeit $f_{\text{time}}(a_i) < 0{,}1$ als den Beginn einer neuen Suchsitzung interpretieren. Somit gilt

$$f_{\text{geom\_cut}}(s_j, a_i) = \begin{cases} f_{\text{geom}}(s_j, a_i) & , f_{\text{time}}(a_i) \geq 0,1 \\ 0 & , f_{\text{time}}(a_i) < 0,1 \end{cases}$$

für unsere manuelle Optimierung des Verfahrens.

Der unten rechts markierte Problembereich enthält einen erheblichen Anteil der Anfragen, bei denen es schwierig ist, mit Sicherheit zwischen einer Sitzungsfortführung und einer neuen Suchsitzung zu unterscheiden. Für den großen Anteil an Aktionen (41,3%), die in zwei verschiedene Suchsitzungen gehören und in diese 2 Quadranten fallen, ist die Aussagesicherheit mit 73 fehlerhaften Aktionen noch passabel. Für die Aktionen (7,7%), die hierbei einer Suchsitzung zuzuordnen sind, ist der Fehler mit 597 betroffenen Aktionen hinsichtlich der Gesamtgüte gravierend.

Aufgrund dieser Ergebnisse werden wir für Aktionen, die in diesen Bereich fallen, in weiteren kaskadierenden Schritten weitere Merkmale zur Entscheidung heranziehen, um die Aussagesicherheit bestmöglich zu erhöhen. Sollte kein weiteres Merkmal zu einer gewünschten Verbesserung führen, werden wir die in diesem Bereich liegenden Suchsitzungen bei der Segmentierung der gesamten Log-Datei in einem Nachbereitungsschritt entfernen. Dabei ist das Ziel, eine Teilmenge von Suchsitzungen von sehr hoher Güte zu bestimmen. Es ist daher nicht notwendig, alle Sitzungen zu bestimmen.

Da in dem Problembereich rechts unten das lexikalische Ähnlichkeitsmaß relativ schwach ist, liegt die Vermutung nahe, dass es sich dabei um Aktionen handelt, die entweder dem Veränderungsmuster *Neu* oder *Umformulierung* zuzuordnen sind (vgl. Abschnitt 2.2.2). Daher erachten wir die explizite semantische Analyse und den Vergleich der Ergebnisse einer Suchmaschine als adäquate weitere Ähnlichkeitsmaße, deren Ergebnisse in den nächsten beiden Abschnitten erläutert werden.

	precision	*recall*	$F_{\beta=1}$	$F_{\beta=1.5}$	*ERR*	*SER*
$f_{\text{geom\_org}}(s_j, a_i)$	0,86726	0,94306	0,90357	0,91836	0,17590	0,20129
$f_{\text{geom}}(s_j, a_i)$	0,85574	0,95172	0,90118	0,91997	0,17986	0,20871
$f_{\text{geom\_cut}}(s_j, a_i)$	0,85533	0,96608	0,90734	0,92907	0,16961	0,19733
$f_{\text{geom\_sample}}(s_j, a_i)$	0,96742	0,97845	0,97291	0,97503	0,05276	0,05450

Tabelle 3.6: Güte der geometrischen Methode

	ACC_{shift}	ACC_{cont}	ACC_{avg}
$f_{\text{geom\_org}}(s_j, a_i)$	0,94306	0,91936	0,92785
$f_{\text{geom}}(s_j, a_i)$	0,95172	0,91037	0,92113
$f_{\text{geom\_cut}}(s_j, a_i)$	0,96608	0,90871	0,92364
$f_{\text{geom\_sample}}(s_j, a_i)$	0,97845	0,98730	0,98579

Tabelle 3.7: Genauigkeit der geometrischen Methode

Die einzelnen Ergebnisse der geometrischen Methode sind in den beiden Tabellen 3.6 und 3.7 erfasst. Dabei entspricht $f_{\text{geom\_org}}$ den originalen Werten von Gayo-Avello [Gay09]. Die Funktion $f_{\text{geom}}(s_j, a_i)$ entspricht unserem Ergebnis für unsere gewählten n-m-Gramme. Darauf aufbauend folgt $f_{\text{geom\_cut}}(s_j, a_i)$. Als letztes Ergebnis ermitteln wir mit $f_{\text{geom\_sample}}$ das Optimum, wobei der untere Problembereich nicht berücksichtigt wird.

3.4.4 Explizite semantische Analyse (*ESA*)

Die explizite semantische Analyse (*ESA*) ist ein kollektionsrelatives Retrieval Verfahren. Vorgestellt von Gabrilovich und Markovitch [GM07], beruht die Idee der *ESA* auf der Charakteristik der Wikipedia beziehungsweise eines Lexikons im Allgemeinen. Nach ihrer Hypothese kann eine Vektorrepräsentation eines Wikipedia-Artikels als ein semantisches Konzept für das beschriebene Artikelthema dienen.

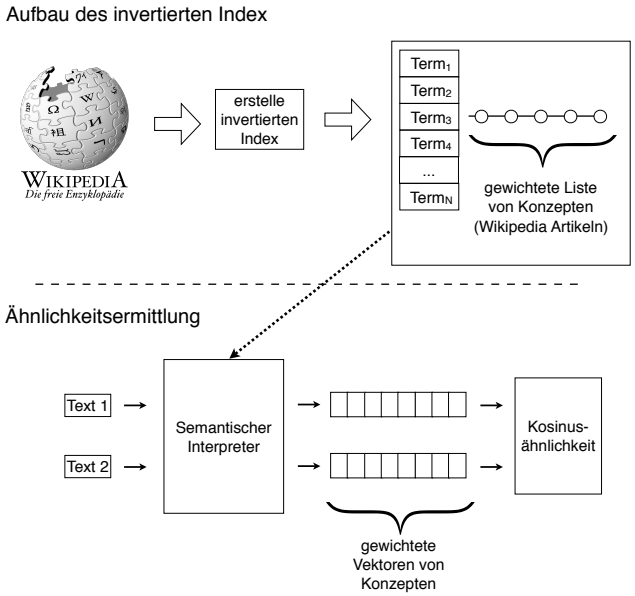

Abbildung 3.11: Schematische Darstellung der *ESA* nach [GM07]

Das Verfahren bedarf eines Aufbereitungsschritts, wie die Abbildung 3.11 erläutert. Auf Basis einer Auswahl von Wikipedia-Artikeln wird eine $tf \cdot idf$-gewichtete Term-Dokument-Matrix erzeugt. Diese dient als Basis für einen invertierten Index. Für zwei zu vergleichende Texte wird entsprechend des Vokabulars je eine transponierte Vektorrepräsentation erstellt und mit der dem

Index zugrunde liegenden Matrix multipliziert. Diese beiden Produkte bilden je einen Vektor, der nach der Ähnlichkeit zum Eingabetext gewichtete Konzepte enthält. Mit dem Kosinus des Winkels zwischen den beiden Vektoren wird schließlich das Ähnlichkeitsmaß $\varphi_{\text{esa}}(s_j, a_i)$ quantifiziert.

Wie im vorherigen Abschnitt 3.4.3 erläutert, werden wir nur für den Teil der Anfragen die in einen bestimmten Problembereich fallen, das Ähnlichkeitsmaß $\varphi_{\text{esa}}(s_j, a_i)$ quantifizieren. Falls die Bedingungen

$$f_{\text{time}}(a_i) > 0{,}8 \text{ und}$$

$$f_{\text{cosSim}}(s_j, a_i) < 0{,}4$$

erfüllt sind, quantifizieren wir dieses Merkmal wie folgt:

$$f_{\text{casc\_esa}}(s_j, a_i) = \begin{cases} 1 & , \varphi_{\text{esa}}(s_i, a_i) > T_{\text{esa}} \\ f_{\text{geom\_cut}}(s_j, a_i) & , \varphi_{\text{esa}}(s_i, a_i) \leq T_{\text{esa}} \end{cases}$$

Ist eine hohe Aussagesicherheit mit $\varphi_{\text{esa}}(s_i, a_i) \leq T_{\text{esa}}$ nicht gegeben, wird das zuvor quantifizierte Merkmal $f_{\text{geom\_cut}}(s_j, a_i)$ herangezogen. Die Verteilung der quantifizierten Ähnlichkeit ist in der Abbildung 3.12 dargestellt.

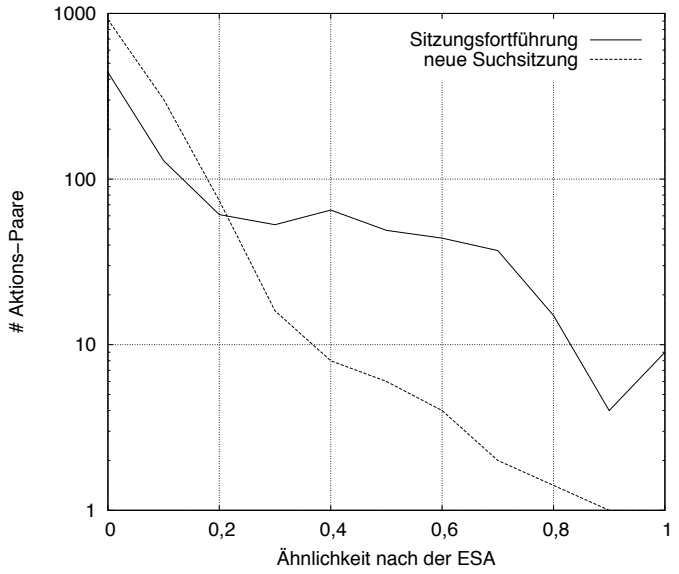

Abbildung 3.12: Verteilung der Ähnlichkeit für ein Merkmal auf Basis der *ESA*

Da die explizite semantische Analyse in das generalisierten Vektorraummodell transformiert werden kann [SA09], sind die einzelnen Terme nach ihrer

Kookkurrenz gewichtet. Wir haben die *ESA* für verschiedene Kollektionsgrößen evaluiert. Nach Anderka und Stein [AS09] variiert mit einer unterschiedlichen zugrunde liegenden Dokumentanzahl die Güte nur gering. Wie den Tabellen 3.8 und 3.9 zu entnehmen ist, können wir dieses bestätigen. Da eine Verarbeitung einer Kollektionsgröße von 100.000 Dokumenten keine besondere Herausforderung für aktuelle Computer-Systeme darstellt, und diese eine marginal höhere Güte aufweist, habe wir uns für die Kollektionsgröße von 100.000 entschieden.

	precision	*recall*	$F_{\beta=1}$	$F_{\beta=1.5}$	*ERR*	*SER*
1.000	0,85561	0,96831	0,90848	0,93059	0,16770	0,19510
10.000	0,86180	0,96187	0,90909	0,92869	0,16667	0,19237
100.000	0,85805	0,96831	0,90986	0,93148	0,16539	0,19188
ca. 3,5 Millionen	0,85677	0,96856	0,90924	0,93117	0,16642	0,19337

Tabelle 3.8: Güte der *ESA* $f_{\text{geom\_esa}}(s_j, a_i)$ für verschiedene Kollektionsgrößen

	ACC_{shift}	ACC_{cont}	ACC_{avg}
1.000	0,96831	0,90871	0,92422
10.000	0,96187	0,91383	0,93785
100.000	0,96831	0,91051	0,92555
ca. 3,5 Millionen	0,96856	0,90954	0,92490

Tabelle 3.9: Genauigkeit der *ESA* $f_{\text{geom\_esa}}(s_j, a_i)$ für verschiedene Kollektionsgrößen

3.4.5 Erweiterte Repräsentation

Wie in Abschnitt 2.2.3 erläutert, ist die erweitere Repräsentation in der Literatur ein häufig verwendetes Merkmal, um zwischen einer Sitzungsfortführung und einer neuen Suchsitzung zu unterscheiden. Auf Basis der Ergebnisse einer Suchmaschine wird dazu eine erweiterte Repräsentation einer Suchanfrage erstellt und auf dessen Grundlage das Ähnlichkeitsmaß $\varphi_{\text{sr}}(s_i, a_i)$ ermittelt. Je nach Verfahren werden hierzu die URLs der Ergebnisse, die durch die Ergebnisliste gegebenen Seitentitel und Text-Auszüge, und die Ergebnisdokumente quantifiziert. Die Ähnlichkeit auf Basis der URLs wird über den Jaccard-Koeffizienten bestimmt. Die Ähnlichkeitsermittlung der beiden anderen Varianten zweier erweiterter Repräsentationen erfolgt über die Kosinusähnlichkeit nach dem standardisierten Vektorraummodel. Dementsprechend erfolgt eine $tf \cdot idf$-Gewichtung der einzelnen Terme.

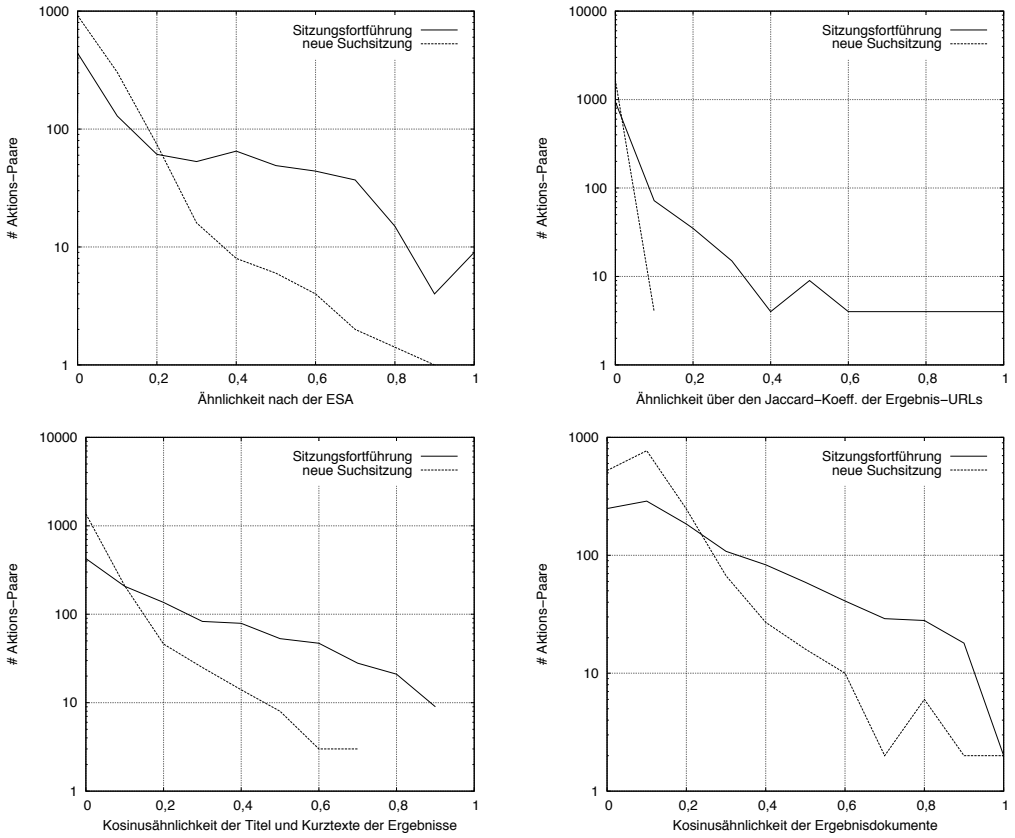

Abbildung 3.13: Ähnlichkeits-Verteilungen der ESA im Vergleich zu den verschiedenen Merkmalen auf Basis der Ergebnisse einer Suchmaschine

Wir haben mit allen genannten Möglichkeiten experimentiert und die einzelnen Resultate für eine Ergebnisanzahl von 10, 20 und 50 Dokumenten evaluiert. Die Verteilungen der Ähnlichkeitsmaße der einzelnen Merkmale im Vergleich zur *ESA* sind in der Abbildung 3.13 aufgetragen.

Analog zu dem Merkmal $f_{\text{geom\_esa}}(s_j, a_i)$ ermitteln wir das Merkmal der erweiterten Repräsentation $f_{\text{casc\_sr}}(s_j, a_i)$ (*search results*) nur dann, wenn die Bedingungen

$$f_{\text{time}}(a_i) > 0{,}8 \ ,$$

$$f_{\text{cosSim}}(s_j, a_i) < 0{,}4 \text{ und}$$

$$\varphi_{\text{esa}}(s_i, a_i) \leq T_{\text{esa}}$$

erfüllt sind, in der Form

$$f_{\text{casc\_sr}}(s_j, a_i) = \begin{cases} 1 & , \varphi_{\text{sr}}(s_i, a_i) > T_{\text{sr}} \\ f_{\text{geom\_cut}}(s_j, a_i) & , \varphi_{\text{sr}}(s_i, a_i) \leq T_{\text{sr}} \, . \end{cases}$$

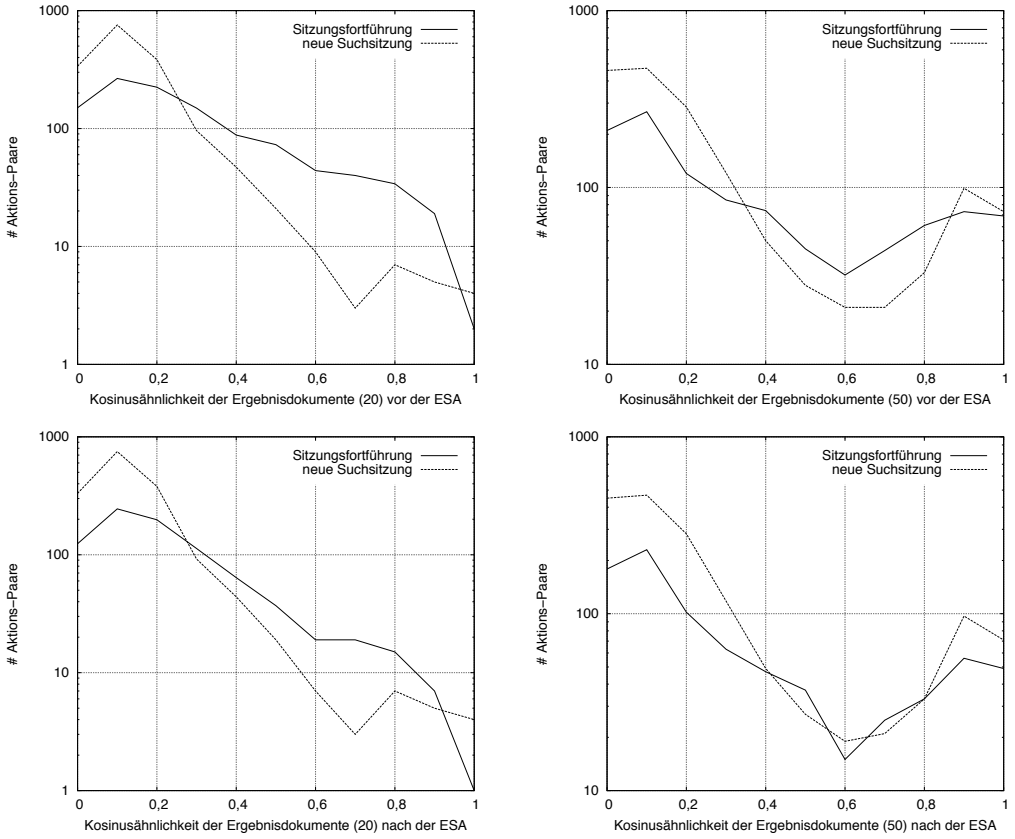

Abbildung 3.14: Ähnlichkeitsverteilungen der ESA im Vergleich zu den verschiedenen Merkmalen auf Basis der Ergebnisse einer Suchmaschine

Die Diagramme in Abbildung 3.14 zeigen den Einfluss der berücksichtigten Ergebnisanzahl, sowie die Redundanz zwischen der *ESA* und dem Merkmal der erweiterten Repräsentation für die Variante der Ergebnisdokumente. Ersteres lässt sich im Vergleich zwischen der linken (20 Dokumente) und der rechten Seite (50 Dokumente) ablesen. Die Redundanz der Diskriminanz beider Merkmale ist dadurch erkennbar, dass oben die Verteilung von $f_{\text{casc\_sr}}(s_j, a_i)$ ohne das vorherige Merkmal $f_{\text{geom\_esa}}(s_j, a_i)$ und unten die Verteilung die Ähnlichkeit der $f_{\text{casc\_sr}}(s_j, a_i)$ mit dem vorgeschalteten $f_{\text{geom\_esa}}(s_j, a_i)$ abgetragen ist. Die Experimente bestätigten das Erwartbare, sodass mehr als 10 bis 20 berücksichtigte Ergebnisse die Güte negativ beeinflussen. Des Weiteren ist die *ESA* in der

Aussagesicherheit mit dem Merkmal der erweiterten Repräsentation deckungsgleich, sodass nur eine von neun Varianten eine marginale Verbesserung der Güte ermöglicht. Interessant ist, dass es sich hierbei um die Quantifizierung des Jaccard-Koeffizienten der übereinstimmende URLs der ersten 10 Ergebnisse handelt. Das entsprechende Ergebnis ist in den Tabellen 3.10 und 3.11 vermerkt.

	precision	*recall*	$F_{\beta=1}$	$F_{\beta=1.5}$	*ERR*	*SER*
$f_{\text{casc\_sr}}(s_j, a_i)$	0,86174	0,96757	0,91159	0,93234	0,16245	0,18767

Tabelle 3.10: Güte der $f_{\text{casc\_sr}}(s_j, a_i)$ für die ersten 10 URLs

	ACC_{shift}	ACC_{cont}	ACC_{avg}
$f_{\text{casc\_sr}}(s_j, a_i)$	0,96757	0,91328	0,92740

Tabelle 3.11: Genauigkeit der $f_{\text{casc\_sr}}(s_j, a_i)$ für die ersten 10 URLs

Da die erweiterte Repräsentation der Suchanfragen mit den Ergebnissen einer Suchmaschine der letzte Schritt unsers kaskadierenden Verfahren ist, ist diese diesem gleichzusetzen:

$$f_{\text{cascade}}(s_j, a_i) = f_{\text{casc\_sr}}(s_j, a_i)$$

Wie bereits weiter oben erwähnt, möchten wir in der Anwendung unseres Verfahrens für die gesamten AOL Log-Datei nur die Suchsitzungen verwenden, für die wir eine hohe Aussagesicherheit gewährleisten können. Somit sollen alle Suchsitzungen entfernt werden die als „unsicher“ gelten. Wenn die Bedingungen

$$f_{\text{time}}(a_i) > 0{,}8 \ ,$$

$$f_{\text{cosSim}}(s_j, a_i) < 0{,}4 \ ,$$

$$\varphi_{\text{esa}}(s_i, a_i) \leq T_{\text{esa}} \text{ und}$$

$$\varphi_{\text{sr}}(s_i, a_i) \leq T_{\text{sr}}$$

erfüllt sind, ermittelt dieses Verfahren

$$f_{\text{casc\_sample}}(s_j, a_i) = \begin{cases} 1 & , f_{\text{casc\_sr}}(s_i, a_i) > T_{\text{strict\_max}} \\ 0 & , f_{\text{casc\_sr}}(s_i, a_i) < T_{\text{strict\_min}} \\ -1 & , T_{\text{strict\_max}} \geq f_{\text{casc\_sr}}(s_i, a_i) \geq T_{\text{strict\_min}} \end{cases}$$

drei verschiedene Klassen. Eine Quantifizierung $f_{\text{casc\_sample}}(s_j, a_i) = 1$ entspricht einer Sitzungsfortführung, $f_{\text{casc\_sample}}(s_j, a_i) = 0$ einer neuen Suchsitzung und $f_{\text{casc\_sample}}(s_j, a_i) = -1$ der neu eingeführten Klasse „unsicher“.

Kapitel 4

Zusammenfassung und Ausblick

Wir haben auf Grundlage der geometrischen Methode [Gay09] und verschiedener weiterer Merkmale ein kaskadierendes Verfahren $f_{\text{cascade}}(s_j, a_i)$ zur Detektierung von Suchsitzungen entwickelt, wobei eine höhere Güte bei einem kleinstmöglichen Effizienzverlust erreicht wird. Darauf aufbauend konnten wir zudem für einen Anteil der Aktionen (74,6%) über eine zusätzliche Filterimplementierung $f_{\text{casc\_sample}}(s_j, a_i)$ eine noch sehr viel höhere Aussagesicherheit erreichen. Ein Vorteil ergibt sich hierbei für die Abtastung einer Untermenge aus einer Anfrage-Log-Datei und deren Segmentierung in Suchsitzungen, sofern nicht alle Anfragen zwingend berücksichtigt werden müssen.

	precision	*recall*	$F_{\beta=1}$	$F_{\beta=1.5}$	*ERR*	*SER*
$f_{\text{geom\_org}}(s_j, a_i)$	0,86726	0,94306	0,90357	**0,91836**	0,17590	0,20129
$f_{\text{cascade}}(s_j, a_i)$	0,86174	0,96757	0,91159	**0,93234**	0,16245	0,18767
$f_{\text{casc\_sample}}(s_j, a_i)$	0,96799	0,97884	0,97339	**0,97548**	0,05185	0,05353

Tabelle 4.1: Güte der kaskadierenden Verfahren $f_{\text{cascade}}(s_j, a_i)$ und $f_{\text{casc\_sample}}(s_j, a_i)$ im Vergleich zur geometrischen Methode $f_{\text{geom\_org}}(s_j, a_i)$

	ACC_{shift}	ACC_{cont}	ACC_{avg}
$f_{\text{geom\_org}}(s_j, a_i)$	0,94306	0,91936	0,92785
$f_{\text{cascade}}(s_j, a_i)$	0,96757	0,91328	0,92740
$f_{\text{casc\_sample}}(s_j, a_i)$	0,97884	0,98826	0,98628

Tabelle 4.2: Genauigkeit der kaskadierenden Verfahren $f_{\text{cascade}}(s_j, a_i)$ und $f_{\text{casc\_sample}}(s_j, a_i)$ im Vergleich zur geometrischen Methode $f_{\text{geom\_org}}(s_j, a_i)$

Die Ergebnisse der beiden Verfahren sind in den beiden Tabellen 4.1 und 4.2 den Ergebnissen der geometrischen Methode $f_{\text{geom\_org}}(s_j, a_i)$ von Gayo-Avello gegenübergestellt [Gay09]. Wir orientieren uns in erster Linie am Gütemaß F-Measure mit $\beta = 1{,}5$, da wir ein fehlerhaftes Zusammenfallen von zwei unterschiedlichen Suchsitzungen als das schwerwiegendere Problem definieren.

Die Abbildung 4.1 veranschaulicht den Ablauf des kaskadierenden Verfahrens $f_{\text{cascade}}(s_j, a_i)$ im Detail. Um die Aktionen der Benutzer in Suchsitzungen zu gruppieren, werden iterativ je nach Fall verschiedene Merkmale der aufeinanderfolgenden Aktionspaare quantifiziert.

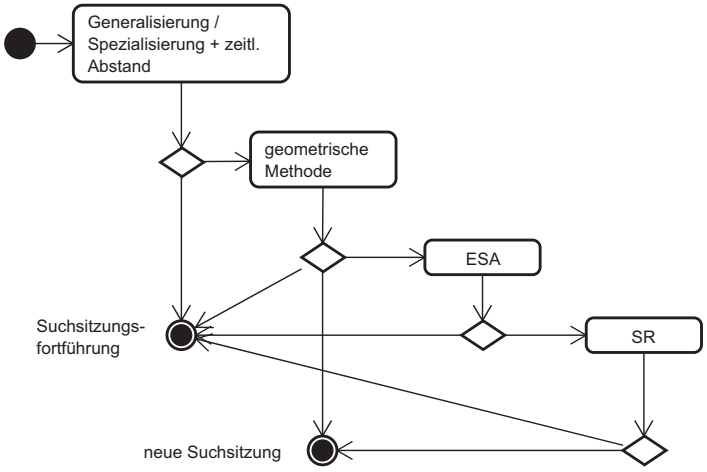

Abbildung 4.1: Funktionsweise des kaskadierenden Verfahrens $f_{\text{cascade}}(s_j, a_i)$

In einem ersten Schritt versuchen wir zunächst, einfache Fälle wie Wiederholung oder Generalisierung und Spezialisierung einer Suchanfrage zu detektieren, indem wir prüfen, ob eine der Suchanfragen die andere enthält. Die weiteren Merkmale werden anhand ihrer „Kosten“, also ihrer Laufzeit, schrittweise quantifiziert. Falls ein Merkmal mit genügender Sicherheit die beiden zu vergleichenden Aktionen klassifizieren kann, bricht der Schritt ab, ohne die überflüssigen Merkmale zu ermitteln. Als weiteres Merkmal quantifizieren wir eine optimierte Variante der geometrischen Methode. Falls bestimmte Bedingungen erfüllt sind, ermitteln wir ein Ähnlichkeitsmaß das auf der expliziten semantischen Analyse basiert. Zuletzt ermitteln wir als das „teuerste“ Merkmal ein Ähnlichkeitsmaß auf Basis der Suchergebnisse beider Suchanfragen. Wenn auch die weiteren Merkmale keine genaue Entscheidung ermöglichen, wird die geometrische Methode als das noch am stärksten diskriminierende Merkmal verwendet.
In Abbildung 4.2 ist die erweitere Filterimplementierung $f_{\text{casc\_sample}}(s_j, a_i)$ des kaskadierenden Verfahrens $f_{\text{cascade}}(s_j, a_i)$ schematisch erläutert. Das Verfahren

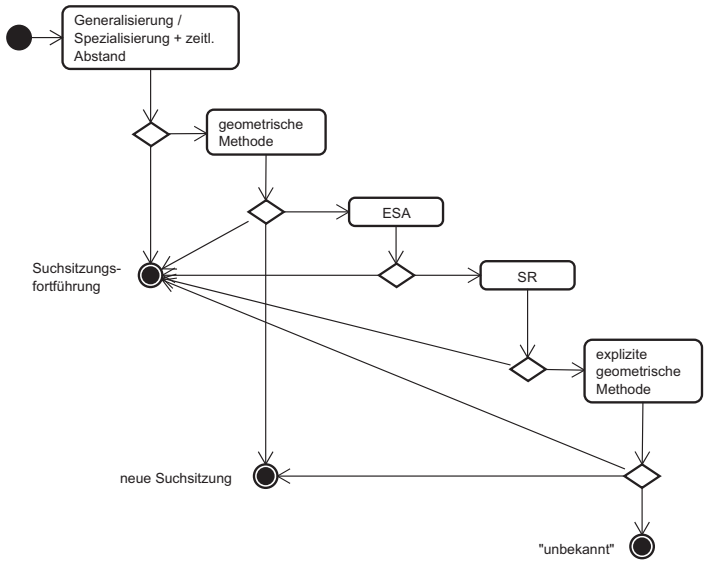

Abbildung 4.2: Funktionsweise des kaskadierenden Verfahrens $f_{\text{casc\_sample}}(s_j, a_i)$

ändert sich nur in dem Detail, wobei eine striktere Variante der geometrischen Methode angewendet wird. Diese klassifiziert neben den Fällen einer Sitzungsfortführung und einer neuen Suchsitzung einen weiteren Fall: „unsicher“.

Neben der Entwicklung dieser beiden optimierten Verfahren konnten wir zudem feststellen, dass für diese Problemstellung bei einem zugrundeliegendem Ähnlichkeitsmaß auf Basis einer *ESA* die Quantifizierung einer erweiterten Repräsentation nur eine sehr marginale Verbesserung der Güte bewirken kann. Diese ist so gering, das in Anbetracht der Kosten zu erwägen wäre, diese nicht zu berücksichtigen. Das größte Problem ist hierbei die mangelnde Transparenz bezüglich der Rangbestimmung, sodass voraussichtliche viele unbekannte Variablen das Ergebnis beeinflussen.
Eine offene Fragestellung ist nach wie vor der Aspekt des Multitaskings. Des Weiteren ist es interessant, ob sich ein Verfahren mit vergleichbaren oder vielleicht sogar besseren Stärken, durch ein maschinelles Lernverfahren realisieren lässt.

Abschließend betrachtet konnten wir zeigen, dass die Kombination mehrere Merkmale zur Suchsitzungsdetektierung die erhoffte Steigerung der Güte ermöglicht.

Abbildungsverzeichnis

Tabellenverzeichnis

Literaturverzeichnis

[AS09] Maik Anderka und Benno Stein. The ESA Retrieval Model Revisited. *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval - SIGIR '09*, Seiten 670–671, 2009.

[Bro02] Andrei Broder. A taxonomy of web search. *ACM SIGIR Forum*, 36(2):3–10, August 2002.

[Buz06a] Nikolai Buzikashvili. An exploratory web log study of multitasking. In *Proceedings of the 29th annual International ACM SIGIR Conference on Research and development in Information Retrieval - SIGIR '06*, Seiten 623–624, 2006.

[Buz06b] Nikolai Buzikashvili. Automatic task detection in the web logs and analysis of multitasking. In *Proceedings of the 8th International Conference on Asian Digital Libraries - ICADL '06*, Seiten 131 – 140, 2006.

[Buz07] Nikolai Buzikashvili. Sliding window technique for the web log analysis. In *Proceedings of the 16th International Conference on World Wide Web - WWW '07*, Seiten 1213–1214, 2007.

[CC04] Shui-Lung Chuang und Lee-Feng Chien. A practical web-based approach to generating topic hierarchy for text segments. In *Proceedings of the 13th ACM Conference on Information and Knowledge Management - CIKM '04*, Seiten 127–136, 2004.

[cI10] comScore Inc. Press release: comScore reports global search market growth of 46 Percent in 2009. `http://www.comscore.com/Press_Events/Press_Releases/2010/1/Global_Search_Market_Grows_46_Percent_in_2009`, Januar 2010.

[CP95] Lara D. Catledge und James Pitkow. Characterizing browsing strategies in the World-Wide web. *Computer Networks and ISDN Systems*, 27(6):1065–1073, April 1995.

[DDH07] Doug Downey, Susan Dumais und Eric Horvitz. Models of searching and browsing: Languages, studies, and applications. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence - IJCAI '07*, Seiten 2740–2747, 2007.

[DF09] Omer Duskin und Dror G. Feitelson. Distinguishing humans from robots in web search logs. In *Proceedings of the 2009 workshop on Web Search Click Data - WSCD '09*, Seiten 15–19, 2009.

[Gay09] Daniel Gayo-Avello. A survey on session detection methods in query logs and a proposal for future evaluation. *Information Sciences*, 179(12):1822–1843, Mai 2009.

[GM07] Evgeniy Gabrilovich und Shaul Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence - IJCAI '07*, Seiten 1606–1611, 2007.

[HE09] Jeff Huang und Efthimis N. Efthimiadis. Analyzing and evaluating query reformulation strategies in web search logs. In *Proceeding of the 18th ACM Conference on Information and Knowledge Management - CIKM '09*, Seiten 77–86, 2009.

[HG00] Daqing He und Ayşe Göker. Detecting session boundaries from web user logs. In *Proceedings of the 22nd BCS annual Colloquium on Information Retrieval Research - IRSG '00*, Seiten 57–66, 2000.

[HGH02] Daqing He, Ayşe Göker und David J. Harper. Combining evidence for automatic web session identification. *Information Processing and Management*, 38(5):727–742, September 2002.

[JBS09] Bernard J. Jansen, Danielle L. Booth und Amanda Spink. Patterns of query reformulation during web searching. *Journal of the American Society for Information Science and Technology*, 60(7):1358–1371, 2009.

[JK08] Rosie Jones und Kristina L. Klinkner. Beyond the session timeout. In *Proceeding of the 17th ACM Conference on Information and Knowledge Mining - CIKM '08*, Seiten 699–708, 2008.

[JMSP06] Bernard J. Jansen, Tracy Mullen, Amanda Spink und Jan Pedersen. Automated gathering of web information: An in-depth examination of agents interacting with search engines. *ACM Transactions on Internet Technology*, 6(4):442–464, November 2006.

[JS06] Bernard J. Jansen und Amanda Spink. How are we searching the World Wide Web? A comparison of nine search engine transaction logs. *Information Processing and Management*, 42(1):248–263, Januar 2006.

[JSB07] Bernard J. Jansen, Amanda Spink und Danielle L. Booth. Defining a session on web Search Engines. *Journal of the American Society for Information Science*, 58(6):862–871, 2007.

[LH99] Tessa Lau und Eric Horvitz. Patterns of search: Analyzing and modeling web query. In *Proceedings of the 7th International Conference on User Modeling*, Seiten 119–128, 1999.

[LOP+10] Claudio Lucchese, Salvatore Orlando, Raffaele Perego, Fabrizio Silvestri und Gabriele Tolomei. Detecting task-based query sessions using collaborative knowledge. In *International Workshop on Intelligent Web Interaction - IWI '10*, 2010.

[MDM07] Donald Metzler, Susan Dumais und Christopher Meek. Similarity measures for short segments of text. *Advances in Information Retrieval*, Seiten 16–27, 2007.

[MKSW99] John Makhoul, Francis Kubala, Richard Schwartz und Ralph Weischedel. Performance measures for information extraction. In *Proceedings of the Broadcast News Workshop*, Seiten 249–252, 1999.

[MLC06] G. Craig Murray, Jimmy Lin und Abdur Chowdhury. Identification of user sessions with hierarchical agglomerative clustering. In *Proceedings of the American Society for Information Science and Technology*, Band 43, Seiten 1–9, 2006.

[ÖÇ05a] H. Cenk Özmutlu und Faith Çavdur. Application of automatic topic identification on Excite web search engine data logs. *Information Processing and Management*, 41(5):1243–1262, September 2005.

[ÖÇ05b] Seda Özmutlu und Fatih Çavdur. Neural network applications for automatic new topic identification. *Online Information Review*, 29(1):34–53, 2005.

[ÖÖB07] Seda Özmutlu, Huseyin C. Özmutlu und Buket Buyuk. Using monte-carlo simulation for automatic new topic identification of search engine transaction logs. *2007 Winter Simulation Conference - WSC '07*, Seiten 2306–2314, Dezember 2007.

[ÖÖS08] Seda Özmutlu, Huseyin C. Özmutlu und Amanda Spink. Automatic new topic identification in search engine transaction logs using multiple linear regression. In *Proceedings of the 41st Annual Hawaii International Conference on System Sciences - HICSS '08*, Seite 140, Januar 2008.

[RJ05] Filip Radlinski und Thorsten Joachims. Query chains: Learning to rank from Implicit feedback. In *Proceeding of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining - KDD '05*, Seiten 239–248, 2005.

[RL03] Ian Ruthven und Mounia Lalmas. A survey on the use of relevance feedback for information access systems. *The Knowledge Engineering Review*, 18(2):95–145, Juni 2003.

[RL04] Daniel E. Rose und Danny Levinson. Understanding user goals in web search. In *Proceedings of the 13th International Conference on World Wide Web - WWW '04*, Seiten 13–19, 2004.

[SA09] Benno Stein und Maik Anderka. Collection-Relative Representations: A Unifying View to Retrieval Models. In *Proceedings of the TIR '09 at the 20th International Conference on Database and Expert Systems Applications - DEXA '09*, Seiten 383–387. IEEE Computer Society, 2009.

[SC06] Nuno Seco und Nuno Cardoso. Detecting user sessions in the Tumba! query log. 2006.

[SH06] Mehran Sahami und Timothy D. Heilman. A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th International Conference on World Wide Web - WWW '06*, Seiten 377–386, 2006.

[SH10] Benno Stein und Matthias Hagen. Making the Most of a Web Search Session. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management - CIKM '10*, August 2010.

[SJÖ00] Amanda Spink, Bernard J. Jansen und H. Cenk Özmultu. Use of query reformulation and relevance feedback by Excite users. *Internet Research*, 10(4):317–328, 2000.

[SLY+10] Shuqi Sun, Sheng Li, Muyun Yang, Haoliang Qi und Tiejun Zhao. Utilizing Variability of Time and Term Content, within and across Users in Session Detection. *Coling 2010: Posters*, Seiten 1203–1210, August 2010.

[SMHM99] Craig Silverstein, Hannes Marais, Monika Henzinger und Michael Moricz. Analysis of a very large web search engine query log. *ACM SIGIR Forum*, 33(1):6–12, September 1999.

[SÖÖ02] Amanda Spink, H. Cenk Özmutlu und Seda Özmutlu. Multitasking information seeking and searching processes. *Journal of the American Society for Information Science and Technology*, 53(8):639–652, 2002.

[Spi07] Amanda Spink. Query Modifications Patterns During Web Searching. In *Proceedings of the International Conference on Information Technology - ITNG '07*, Seiten 439–444, IEEE Computer Society, 2007.

[SPJP06] Amanda Spink, Minsoo Park, Bernard J. Jansen und Jan Pedersen. Multitasking during Web search sessions. *Information Processing and Management*, 42(1):264–275, Januar 2006.

[STZ05] Xuehua Shen, Bin Tan und ChengXiang Zhai. Implicit user modeling for personalized search. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management - CIKM '05*, Seiten 824–831, 2005.

[Swa77] Don R. Swanson. Information retrival as a trial-and-error process. *Library Quarterly*, 47:128–148, 1977.

[SY06] Xiaodong Shi und Christopher C. Yang. Mining related queries from search engine query logs. In *Proceedings of the 15th International Conference on World Wide Web - WWW '06*, Seiten 943–944, 2006.

[SYL+09] Shuqi Sun, Muyun Yang, Sheng Li, Haoliang Qi und Tiejun Zhao. Geometric session detection method for sogou log. *Computational Information Systems*, 5(6):1687–1696, 2009.

[Yu04] Holly Yu. The Impact of Web Search Engines on Subject Searching in OPAC. *Information Technology and Libraries*, Seiten 168–180, 2004.

[ZCBY10] H. Zaragoza, B.B. Cambazoglu und R. Baeza-Yates. Web Search Solved? All Result Rankings the Same? *19th ACM International Conference on Information and Knowledge Management - CIKM '10*, 2010.

[ZM06] Yuye Zhang und Alistair Moffat. Some observations on user search behavior. In *Proceedings of the 11th Australasian Document Computing Symposium*, Seiten 1–8, 2006.