Selbstlernende Suchmaschine als zentraler Informationszugang bei heterogener Informationslandschaft

Julian Bahrs, Benedikt Meuthrath, Kirstin Peters

Lehrstuhl für Wirtschaftsinformatik und Electronic Government
Universität Potsdam
August-Bebel-Straße 89
14489 Potsdam
[jbahrs, bmeuthrath, kpeters]@wi.uni-potsdam.de

Abstrakt: Bisherige Ansätze für die Informationssuche im Unternehmen berücksichtigen die Charakteristik der in Unternehmen vorherrschenden dezentralen und heterogenen Informationsquellen nicht ausreichend oder decken nur begrenzte Teile der vorhandenen Information ab. Zur Verbesserung der Suche in Unternehmen wird ein lernendes System zur Suche in verschiedenen Informationsquellen mit profil- und kontextorientiertem Ranking vorgestellt, das einen zentralen Informationszugang im Unternehmen ermöglicht.

1 Einleitung

Für die Wettbewerbsfähigkeit von Unternehmen ist die Wiederverwendung und der schnelle Zugriff auf Informationen aus Effizienzgründen und zur Erhöhung der Reaktionsgeschwindigkeit erforderlich. Im Beitrag wird zunächst anhand einer empirischen Untersuchung der erreichte Stand von Suchinstrumenten in Unternehmen im deutschen Sprachraum erläutert (Abschnitt 2). Dabei werden auch Besonderheiten der Suche in Unternehmen sowie bisherige Ansätze und deren Defizite vorgestellt und so ein bestehendes Problemfeld skizziert. Der zweite Teil des Beitrags beschreibt zunächst die Architektur (Abschnitt 3) und anschließend die Funktionsweise (Abschnitt 4) der im Forschungsprojekt „Selbstlernende Suchmaschine für die profil- und kontextbezogene Suche in unternehmensweiten Informationsbeständen“ (SLS) entwickelten Suchmaschine.

2 Suche im Unternehmen – Status und Defizite

Die Leistungsfähigkeit der Suchmaschinen und der Anwenderkreis haben durch die Entwicklung im Internet stark zugenommen. In Unternehmen ist diese Entwicklung jedoch noch nicht angekommen.

So ist die berufbezogene Nutzungsfrequenz unternehmensexterner Suchinstrumente deutlich höher, als die der unternehmensinternen [BSM07]. Auch gehen nur rund 20% der Befragten (n=140) davon aus, dass im Unternehmen vorhandene Informationen zu einem Thema gefunden werden kann. Rund 60% vermuten dies, es besteht jedoch ein erhebliches Misstrauen gegenüber den Suchergebnissen. Die verbleibenden 20% erwarten gar nicht, dass diese Information aufgefunden werden können [BSM07]. Dies ist auf die vorhandenen Suchinstrumente im Unternehmen zurückzuführen. Zwar setzen über 80% der Unternehmen Suchmaschinen ein. Jedoch sind die Informationen nur teilweise und erst durch die Nutzung mehrerer Suchinstrumente zugänglich [BSM07]. Ein zentraler Informationszugang durch eine unternehmensweite Suche fehlt.

Die Gründe dafür liegen in der für Suchmaschinen schwierigen Umgebung, die in Unternehmen vorherrscht: Zunächst gibt es kein einheitliches Format oder Struktur, in der Informationen gespeichert werden, wie es zum Beispiel bei Webseiten im Internet oder einer Fachdatenbank der Fall ist. In Unternehmen werden vielfältige applikationsspezifische Dokumentenformate, Intranet Webseiten, diverse proprietäre Systeme und Datenbanken verwendet. Dies führt zu einer hinsichtlich Struktur heterogenen Informationslandschaft mit dezentralen Speichersystemen. Weiterhin werden im Unternehmen in der Regel Zugriffsrechte eingeschränkt. Das Pendant im Internet, das sogenannte Deep Web, also Informationen die aus Fachdatenbanken oder nur durch die Nutzung von Formularen oder aus nicht öffentlichen Bereichen stammen, wird jedoch auch im Internet bisher kaum erschlossen [MH08]. In Unternehmen ist jedoch der überwiegende Teil der Informationen nur mit entsprechenden Zugriffsrechten erreichbar. Ein zentraler Index muss daher die Zugriffsrechte abbilden können. Dies ist bei Datenbankstrukturen, wo ggf. einzelne Attribute gesondert geschützt werden, komplex.

Im Forschungsfeld Enterprise Search wird die Suche über alle Textinhalte die in digitaler Form im Intranet und auf den Webseiten eines Unternehmens, in Datenbanken, E-Mails, Dokumenten usw. vorzufinden sind, verfolgt [Ha04]. Ansätze hierzu sind zum einen die Integration der Informationsrepositories, z. B. durch übergeordnete Wissensmanagementsysteme oder Enterprise Content Management Systeme. Diese Systeme verfügen i.d.R. über integrierte Suchinstrumente, die heute häufig über eine der größten Erschließungsreichweiten im Unternehmen verfügen. Sie erreichen dies durch die Vereinheitlichung von Zugriffsrechten und Metadaten, sowie oft eines (oder mehreren) quellenübergreifenden Ordnungssystems. Auch aktuelle Ansätze für Enterprise Search verfolgen einen solchen integrierenden Ansatz.

Der zweite Ansatz ist die Schaffung von lokalen Lösungen. Dazu zählen vor allem Fachanwendungen, die zur Deckung von zuvor als relevant identifizierten Informationsbedürfnissen eingesetzt werden. Durch die Konzentration auf eine schmale Domäne werden gezieltere Abfragen mit spezifischen Abfrageparametern, unter Berücksichtigung der Zugriffsrechte, möglich. Auch für die iterative Verfeinerung der Suchanfrage kann das gesamte Methodenspektrum des Information Retrievals, von der Datenbankanfrage bis zur Applikationsentwicklung, angewendet werden [vgl. White 2007]. Es ist für die Anwender jedoch schwierig das „richtige“ Suchinstrument vor der Stellung der Suchanfrage auszuwählen.

Bisherige Ansätze für Enterprise Search verfolgen vor allem die erste, integrierende Richtung und stoßen dabei an Grenzen. Im Folgenden wird ein Ansatz vorgestellt, der die Leistungsfähigkeit dezentraler Suchdienste für das unternehmensweite Information Retrieval nutzt. Das zusammenfassende Ranking der Teilergbnisse erfolgt mittels fallbasiertem Schließen (engl. Case-based Reasoning, kurz CBR) (vgl. Abschnitt 4)

3 Selbstlernende Suchmaschine – Architektur

Die SLS ist eine modular aufgebaute Meta-Suchmaschine, welche sich vorrangig auf Webservices stützt, die von Informationsquellen angeboten werden. Eine Informationsquelle bezeichnet dabei eine Suchmaschine für eine spezifische Domäne, die minimal für jedes Suchresultat folgende Informationen bereitstellen muss: ein uniform Resource Identifier (URI)) [BFM05], ein Ranking-Wert, optional eine kurze Beschreibung des Treffers sowie optional eine Information über die Quelle des Treffers.

Die Architektur der SLS in Abbildung 2 ist mit Hilfe der Fundamental Modeling Concepts (FMC) modelliert [KGT05]. Dargestellt ist die Kompositionsstruktur als Blockdiagramm mit aktiven und passiven Komponenten. Aktive Komponenten sind Agenten, welche als Rechteck modelliert werden. Passive Elemente sind Speicher, dargestellt als abgerundete Form, und Kommunikationskanäle, im Modell markiert mit Kreisen. Passive Komponenten speichern oder transportieren Informationen. Das Blockdiagramm beschreibt somit welche Agenten Zugriff auf welche Daten haben und wie sie untereinander über Kommunikationskanäle oder gemeinsam genutzte Speicher kommunizieren.

Die SLS besteht aus einer Kernanwendung, welche eine webbasierte graphische Benutzungsschnittstelle hat. Dem Nutzer wird ermöglicht, Anmeldedaten einzugeben, die an die Informationsquellen weitergereicht werden. Weiterhin wird ihm die Möglichkeit geboten, zu einer Suchanfrage einen Kontext (vgl. Abschnitt 4) auszuwählen oder einen neuen anzulegen. Die Kernanwendung leitet diese Informationen zu einer Ranking-Komponente, welche wiederum eine Integration Factory nutzt, um Konnektoren zu den einzelnen Webservices zu instanziieren. Die Ranking-Komponente sammelt die Suchergebnisse der einzelnen Webservices, fasst diese in einer Liste zusammen und ordnet sie gemäß der durch die Quell-CBR-Komponente ermittelten Werte. Diese CBR-Komponente nutzt die (standardisierten) originären Ranking-Werte, welche durch die Webservices der zugrundeliegenden Suchmaschinen übermittelt werden, und die erinnerten Fälle um den endgültigen Ranking-Wert zu berechnen [vgl. Abschnitt 4]. Die Ergebnisliste wird dem Nutzer präsentiert, welcher Feedback in Form von Evaluationen einzelner Treffer geben kann. Dieses Feedback wird an die Quell-CBR-Komponente weitergeleitet. Eine weitere Term-CBR-Komponente speichert diejenigen Suchanfragen, die zu einem positiv evaluierten Treffer geführt haben. Diese werden benutzt, um einem weiteren Nutzer des Systems mit ähnlicher Suchanfrage eine Liste mit potenziell erfolgreichen, alternativen Suchanfragen zu präsentieren [GL03].

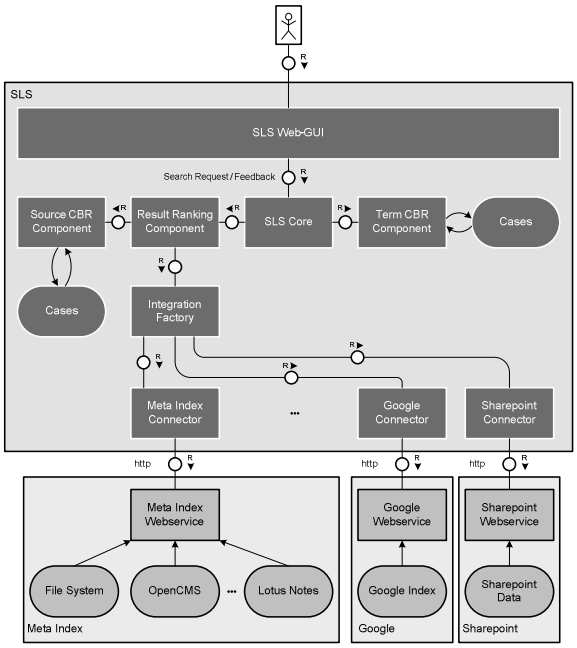

Abbildung 1: Architektur des SLS Prototyps

Die Konnektoren sind variabel in der Implementierung. Beispielsweise stellt der „Meta index connector“ eine Verbindung zu einem Webservice her, der mehr als eine Quelle durchsucht. Dies erlaubt es große Informationssammlungen, wie z.B. MS SharePoint, weiter zu zerteilen, um eine genauere Bewertung durch die CBR-Komponente zu ermöglichen.

4 Fallbasiertes Schließen

Die CBR-Komponente beeinflusst die Zusammenstellung der Suchergebnisse aus den einzelnen Informationsquellen, indem sie deren Ranking-Werte modifiziert. Da Suchende i.d.R. nur die obersten Ergebnisse genauer betrachten, spielt das Ranking eine große Rolle für die wahrgenommene Qualität der Ergebnisliste [JR07]. Wir gehen davon aus, dass die einzelnen Informationsquellen spezialisierte Suchlösungen nutzen, die bereits qualitativ hochwertige Ergebnislisten für die Einträge innerhalb der Suchquelle liefern. Zur Steigerung der Qualität der Suchergebnisse innerhalb einer Quelle können bekannte Methoden [vgl. z.B. GL03] einschließlich einer erneuten Personalisierung und fallbasiertem Schließen verwendet werden. Außerdem können auch innerhalb der beschriebenen Meta-Suchmaschine bekannte Methoden zur Verbesserung der Ergebnisse genutzt werden. In diesem Beitrag wird das Zusammenfügen der einzelnen Ergebnislisten zu einer qualitativ hochwertigen Gesamtliste fokussiert. Wir gehen weiterhin davon aus, dass die Inhalte der verschiedenen Informationsquellen zwar nicht komplett disjunkt sind, aber dennoch unterschiedliche Aufgaben erfüllen und Inhalte zu unterschiedlichen Zwecken enthalten. Unser Ansatz sieht nun vor, für jede Anfrage die

relevanten Informationsquellen zu identifizieren, d.h. die Informationsquellen mit der Intention hinter der Anfrage entsprechenden Inhalten, und die entsprechenden Ergebnisse im Ranking zu bevorzugen.

Aufgrund der Dynamiken in Unternehmen können sich die Informationsquellen und die Aufgaben im Unternehmen ändern. Somit müssen auch die ermittelten Relevanzen ständig überdacht werden. Deshalb wird das maschinelle Lernverfahren des fallbasierten Schließens für die Ermittlung der relevanten Informationsquellen herangezogen.

Als fallbasiertes Schließen bzw. CBR wird das Lösen von Problem mit Hilfe von Erfahrungen aus ähnlichen Situationen bezeichnet. Roger Schank beschreibt die zentrale Rolle von Erfahrungen mit früheren Situationen für die Fähigkeit des Menschen zu Denken und zu Lernen [Sc82]. Erfahrungen werden in Form von Fällen bestehend aus einer Problembeschreibung, seiner Lösung und einer Bewertung der Lösung gespeichert. Die Grundidee von CBR ist, dass für ähnliche Probleme ähnliche Lösungen existieren [Ko92, Le96].

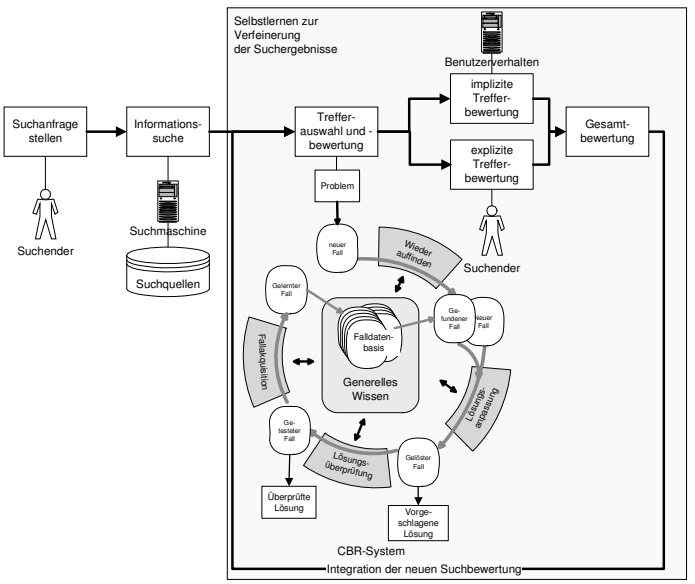

Abbildung 2: Selbst lernender Suchprozess

Das Lösen von Problem mit Hilfe von CBR kann in vier Schritte unterteilt werden (illustriert durch den CBR-Zyklus [AP94] innerhalb der Abbildung 3): *Wieder auffinden* ähnlicher Fälle in der Falldatenbasis. *Anpassung der Lösungen* an den neuen Fall. *Überprüfung* der vorgeschlagenen Lösung und *Akquisition* des gelernten Falles.

Die CBR-Komponente lernt aus den gesammelten Erfahrungen, welche Informationsquellen für welche Anfragen relevant sind, durch die Erinnerung an einen ähnlichen Fall und die dort relevanten Quellen. Die Anfrage wird durch eine Kombination aus Profil und Kontext beschrieben. Das Profil umfasst die generellen Aufgaben eines Mitarbeiters und die generelle Intention hinter seinen Anfragen. Profile werden im Vorfeld angelegt. Sie können zum Beispiel aus der Aufbauorganisation des Unternehmens abgeleitet werden. Das Profil ist rollenorientiert, daher kann mehreren Mitarbeitern das gleiche Profil zugeordnet sein. Der Kontext spezifiziert die Intention einer Anfrage und verfeinert damit das Profil. Kontexte können von allen Mitarbeitern mit dem gleichen Profil angelegt und getauscht werden. Für jede Anfrage wählt der Suchende einen angelegten Kontext aus. Profil und Kontext bilden zusammen die Beschreibung des Problems, d. h. die Intention des Suchenden, und identifizieren damit einen Fall in der Falldatenbasis.

Um zu erlernen welche Informationsquellen bevorzugt werden sollten, benötigt die CBR-Komponente Feedback über die Qualität der präsentierten Suchergebnisse. In der Literatur werden verschiedene Ansätze zur Gewinnung von Feedback beschrieben [KT07, JR07, SH05]. In jedem Fall wird jeder Informationsquelle ein Koeffizient zur Beeinflussung des Rankings der Einträge der entsprechenden Quelle zugeordnet. Wird für ein Suchergebnis positives Feedback gegeben, so wird der Koeffizient der entsprechenden Quelle erhöht. In der nächsten Anfrage im gleichen Fall, d. h. unter der gleichen Kombination aus Profil und Kontext, wird das Ranking für Ergebnisse dieser Quelle entsprechend des Koeffizienten erhöht. Bei negativem Feedback wird entsprechend der zugehörige Koeffizient verkleinert.

5 Ausblick und Fazit

Der vorgestellte Ansatz passt sich an Umgebungen mit heterogenen Informationsquellen an und eignet sich um einen zentralen Informationszugang bereitzustellen. Die Personalisierung und Kontextualisierung mit anschließender Bewertung der Ergebnisqualität führt zu einer selbstlernenden Verbesserung des Rankings, auch ohne vorherige Analyse von Informationsbedürfnissen oder Definition von Interessensprofilen.

Eine prototypische Implementierung der SLS auf Basis einer existierenden, quelloffenen Metasuchmaschine befindet sich derzeit in der Umsetzung. In einem Piloteinsatz wird der Prototyp in verschiedenen Stufen der Entwicklung getestet. Gleichzeitig wird ein Protokoll über sämtliche Aktionen der Benutzer erhoben. Die Entwicklungsstufen umfassen den normalen Einsatz als 1. Metasuchmaschine, 2. mit Bewertung der geöffneten Treffer, 3. mit Personalisierung über eine Nutzerauthentifikation und 4. mit Auswahl des Suchkontextes und über die CBR-Komponente geändertem Ranking. Mittels der gesammelten Daten soll eine Effizienzsteigerung bei der Suche nachgewiesen werden.

Zu einem späteren Zeitpunkt kann die entstehende Fallbasis für weitere Untersuchungen, wie Beispielsweise die Identifikation ungedeckter Informationsbedürfnisse bei anhaltend negativer Ergebnisbewertung, genutzt werden.

Literaturverzeichnis

[AP94] Aamodt, A., Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. 1994.

[BFM05] Berners-Lee, T., Fielding, R., Masinter, L.: "RFC 3986 - Uniform Resource Identifier (URI): Generic Syntax"; The Internet Society, IETF, January 2005.

[BSM07] Bahrs, J.; Schmid, S.; Müller, C.; Fröming, J.: Wissensmanagement in der Praxis - Empirische Untersuchung. Gito (Berlin), 2007.

[Fe04] Feldmann, S.: The high cost of not finding information. http://www.kmworld.com/Articles/ReadArticle.aspx?ArticleID=9534 (Abruf am: 21.11.2007).

[GL03] Gronau, N., Laskowski, F.: „Using Case-Based Reasoning to Improve Information Retrieval in Knowledge Management Systems"; In: E. Menasalvas, J. Segovia, P. Szczepaniak (Hrsg.): Advances in Web Intelligence. Proc. AWIC 2003, Madrid, May 2003; S. 94-102.

[Ha04] Hawking, D.: Challenges in enterprise search. In: Klaus-Dieter Schewe, Hugh Williams (Hrsg.): Proceedings Fifteenth Australasian Database Conference, Volume 27. Australian Computer Society, Inc. (Dunedin, New Zealand), 2004; S. 25-24.

[JR07] Joachims, T., Radlinski, F.: Search Engines that Learn from Implicit Feedback. In Computer, 40(8), 2007; S. 34–40.

[KGT05] Knöpfel, A., Gröne, B., Tabeling, P.: "Fundamental Modeling Concepts. Effective Communication of IT Systems"; John Wiley & Sons Ltd., Chichester (2005).

[Ko92] Kolodner, J.: An introduction to case-based reasoning. In Artificial Intelligence Review, 6(1), 1992; S. 3–34.

[KT07] Kubat, M., Tapia, M.: Time spent on a web page is sufficient to infer a user's interest. In Proceedings of the IASTED European Conference: internet and multimedia systems and applications table of contents, 2007; S. 41–46.

[Le96] Leake, D.: CBR in Context: The Present and Future. Case-Based Reasoning: Experiences, Lessons, and Future Directions, 1996; S. 3–30.

[MH08] Madhavan, J.; Halevy, A.: Crawling through HTML Forms. http://googlewebmastercentral.blogspot.com/2008/04/crawling-through-html-forms.html (Abruf am: 23.04.2008).

[Sc82] Schank, R.: Dynamic Memory: A Theory of Reminding and Learning in Computers and People. Cambridge University Press New York, NY, USA, 1982.

[SH05] Sharma, H., Jansen, B.: Automated evaluation of search engine performance via implicit user feedback. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, 2005; S. 649–650.

[Wh07] White, C.: What is the Difference Between Querying and Browsing Data? http://www.b-eye-network.com/blogs/business_integration/archives/2007/09/what_is_the_dif.php (Abruf am: 13.09.2007).