Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Bachelor

Andrei Sokolovski

Erstellen eines Systems zur Ermittlung aktueller Themen aus Nachrichtentexten

Andrei Sokolovski

Erstellen eines Systems zur Ermittlung aktueller Themen aus Nachrichtentexten

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung
im Studiengang Technische Informatik
am Studiendepartment Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer : Prof. Dr. rer. nat. Kai von Luck
Zweitgutachter : Prof. Dr.-Ing. Andreas Meisel

Abgegeben am 4. Dezember 2007

Thema der Diplomarbeit

Erstellen eines Systems zur Ermittlung aktueller Themen aus Nachrichtentexten

Stichworte

Informationswiedergewinnung, Medienbeobachtung, Themenextraktion, Clusterbildung

Kurzzusammenfassung

Effektive Medienbeobachtung ist heute ein wichtiger Erfolgsfaktor für ein Unternehmen. Themenverfolgung, Trendkenntnisse liefern dem Management zeitnahe Information, mit der das Unternehmen aktiv agieren kann. Meldungen unterschiedlicher Quellen müssen kategorisiert werden, um einen strukturierten Zugang für die unstrukturierten Daten zu erhalten. Ziel dieser Arbeit ist ein System zu entwickeln, dass aus einer unübersichtlicher Nachrichtenmenge eine begrenzte Anzahl von besonders wichtigen Nachrichten extrahiert und thematisch gruppiert bereitstellt.

Title of the paper

Creating of a System for the Identification of Current News Topics from Current Affairs Messages

Keywords

information retrieval, media monitoring, theme extraction, clustering

Abstract

Effective media monitoring is an important success factor for a company. Topics persecution, trends knowing - both deliver timely management information, with which the company can operate actively. Reports from different sources must be classified in order to provide a structured access to unstructured data. The goal of this work is to develop a system, that extracts and thematically groups a limited number of especially important messages from a blind mass of information.

Inhaltsverzeichnis

Tabellenverzeichnis

Abbildungsverzeichnis

1. Einführung

Effektive Medienbeobachtung ist heute ein wichtiger Erfolgsfaktor für ein Unternehmen. Wissen was die Presse über das Unternehmen, sein Management, seine Produkte schreibt ist bedeutungsvoll im Hinblick auf das Image von Unternehmen (unternehmensbezogene Personen und Produkten). Themenverfolgung und Trendkenntnisse liefern dem Management zeitnahe Information, mit der das Unternehmen aktiv agieren kann. Außer Firmenleitung können spezialisierte Abteilungen wie Vertrieb, Forschung oder Produktion mittels gezielter Medienbeobachtung ihre Leistung steigern (zum Beispiel um mögliche Konkurrenzprodukte auszumachen oder um neue Entwicklungstrends möglichst früh zu erkennen).

Medienbeobachtung beinhaltet längst nicht mehr nur Beobachtung der Print-Medien. Elektronische Medien bieten heutzutage eine Vielzahl von Informationsquellen.

Einige Beispiele:

- PMG Presse-Monitor Deutschland GmbH & Co. KG
- Presseagenturen (factiva, dpa, Reuters etc.)
- RSS-Feeds und Email Newsletter
- Multimedia (TV/Hörfunk)
- Scan & Clip

Angesichts der Rolle, die Internet inzwischen spielt, gewinnt Online - Pressemonitoring mehr und mehr an Bedeutung.

Firmen mit hoher Medienpräsenz haben hierzu eigene Pressestellen oder Abteilungen für Pressemonitoring, die für einen bestimmten Leserkreis Meldungen unterschiedlicher Quellen (Zeitungsartikeln, Agenturmeldungen, Pressemitteilungen, etc.) aufbereiten. Diese Informationen müssen kategorisiert werden um einen strukturierten Zugang für die große Menge von unstrukturierten Daten zu ermöglichen.

Die Kategorisierung erfolgt durch manuelle Klassifikation und Zuordnung der Meldungen zu zuvor ermittelten Themen. Dies wird gewöhnlich manuell von Lektoren vorgenommen, welche hierzu jede einzelne Meldung sichten müssen. Den Redakteuren (oder auch Lektoren)

obliegt auch die Ermittlung von neuen Themen, die für das Unternehmen bzw. die Behörden in der Außendarstellung Relevanz erlangen kann. Das Auffinden der aktuellen Themen stellt eine Art Frühwarnsystem zur Verfügung, so dass Unternehmen rechtzeitig aktiv agieren können.

Die konventionelle Themenextraktion erfordert, dass der Redakteur sich einen Überblick über eine Vielzahl von Meldungen und Berichten verschafft und diese nach Häufungskriterien (Clusterbildung) gruppiert. Dies erfordert professionelle Analyse, was bei großen Mengen von Daten zu erheblichen Aufwand führen kann (in der Regel ist dann eine volle Medienanalyse-Infrastruktur notwendig).

Diese manuelle Arbeit könnte durch automatisierte Klassifikationsmechanismen unterstutzt werden. Für die Bestimmung von aktuellen Themen (Thema des Tages, des Monats usw.) aus einer Reihe einlaufender Meldungen sollen Möglichkeiten der Clusterbildung und Distanzermittlung von Volltextinhalten untersucht und genutzt werden. Diese können als eine Vorschlagsliste von Themen und in einem weiteren Schritt für eine automatisierte Kategorisierung der Meldung mittels entsprechender Klassifikatoren genutzt werden.

Die vorliegende Arbeit befasst sich mit dem Entwurf und der Implementierung eines Systems, das oben erwähnte Kategorisierungsaufgaben zum Ziel hat.

Im Kapitel Analyse wird zunächst ein bestehendes kommerzielles Nachrichtenportal vorgestellt, in dessen Umfeld das System integriert werden soll. Danach werden Kundenanforderungen untersucht und der Rahmen für den Projektumfang gesetzt. Das Kapitel Entwurf beschreibt kurz die grundliegende Konzepte und Techniken, die für das Projekt wichtig sind. Es gibt eine Vorstellung auf abstrakter Ebene über die Architektur des Systems. Anschließende Abschnitte setzten sich mit einzelnen Java-Klassen und Datenstrukturen auseinander. In dem Kapitel Evaluierung wird die Effizienz der Module und des Gesamtsystems in betrachtet, Implementierungs- und Optimierungsprobleme und deren Lösungen dargestellt.

2. Analyse

2.1. Umfeldbeschreibung

Die uknow GmbH ist europaweiter Anbieter von Wissensmanagementlösungen im Intranet. Angewandte Lösungen existieren für die Bereiche PR und Unternehmenskommunikation sowie abteilungsübergreifend als Kommunikationssoftware für die Team- und Projektarbeit.

Die moderne Vielfalt von Informationsquellen entspringt dem zentralen Problem, dass die gesamte Nachrichtenversorgung einer Organisation weder inhaltlich noch kostenseitig von einem einzelnen Informationsanbieter optimal bedient werden kann. Ein Produkt der Firma - uknow News Manager - versorgt das Unternehmen mit der nötigen Flexibilität, dieses Optimierungsproblem effizient zu lösen. Das Produkt ist dafür geschaffen, alle diese Informationen zu bündeln und zu harmonisieren [uknow (2007)].

Es handelt sich um eine Intranet-Portalanwendung, deren Aufgabe die Aufnahme, die Speicherung und die Verteilung von Informationen aus unterschiedlichen Quellen ist. Das System kann Inhalte aus dem Internet überwachen, Tickermeldungen von Nachrichtenagenturen bereitstellen, Emaildienste integrieren, Inhalte von Contentlieferanten (z.B. PMG) aufnehmen etc. Es stellt Schnittstellen zu den gängigen Clipping- und Scansystemen für Presseanwendungen bereit. Die gesamte Informationsaufnahme erfolgt automatisiert und ohne manuelle Tätigkeit. Mit Hilfe dieser Funktionalität ist ein automatisiertes Monitoring ohne personellen Aufwand möglich.

Für die Informationslieferung stehen grundsätzlich vier Transportverfahren zur Verfügung. Bei einem HTTPInterface werden Daten über HTTP-Requests von einem vom Content-Provider im Internet bereitgestellten System abgeholt. Bei der Nutzung eines Email-Transport werden Daten über SMTP-Mail vom Content-Provider an eine Emailadresse versendet. Hinter der Emailadresse muss sich ein Mail-Postfach verbergen. Auf das Postfach wird mit Hilfe des IMAPS Protokolls zugegriffen. Als drittes Verfahren werden RSS Feeds genutzt. Hierbei wird in einem definierten Zeitraum der Feed auf neue Dokumente überprüft. Die Metainformationen werden aus dem Feed entnommen. Als Dokument wird eine Referenz auf das HTML Dokument genutzt. Diese wird heruntergeladen und lokal im News Manager gespeichert. Schließlich wird als viertes Verfahren das XML-basierte Filesystem-Importverfahren genutzt.

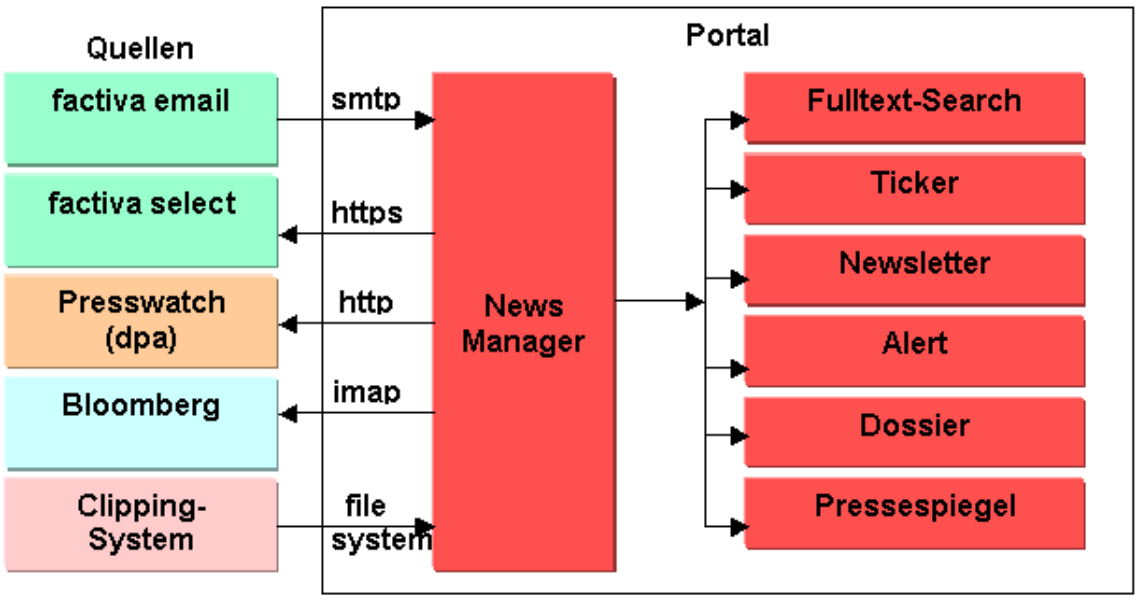

Abbildung 2.1.: News Manager: Quellen und Dienste

Basismodule für die Portalanwendung sind standardisierte Komponente und stammen aus Open-Source Welt.

Das Portal ist eine übergeordnete Struktur, die jeweils benutzerbezogen aus mehreren Einheiten (Workspaces und Portlets) je nach Aufgabe und Rolle des Benutzers unterteilt ist. Anzahl und Inhalt der Workspaces sind entweder durch eine zentrale Administration oder durch den Benutzer aufzubauen (Abbildung 2.3).

Informationsdienste im Portal:

- Das Pressearchiv ist eine Kernfunktion des System, es umfasst alle Artikel und Meldungen des Systems im Volltext inklusive ihrer Metadaten wie z.B. Titel, Quelle, Erscheinungsdatum (Abbildung 2.4).
- Der Ticker ist eine wichtige Komponente des Grundsystems. Er ermöglicht den schnellen Zugang zu den aktuellsten Agenturmeldungen. Benutzer können entweder auf alle Meldungen zugreifen oder auch nach vorgefertigten Rubriken filtern (Abbildung 2.5).
- Der Newsletter ist ein personalisierter Dienst der z.B. eine Zusammenstellung unterschiedlicher Agentur-Meldungen und Presseartikel in einem definierten Zeitraum zusammenfasst.
- Der Alert Dienst dient dazu wichtige Nachrichten sofort an interessierte Empfänger zu senden.

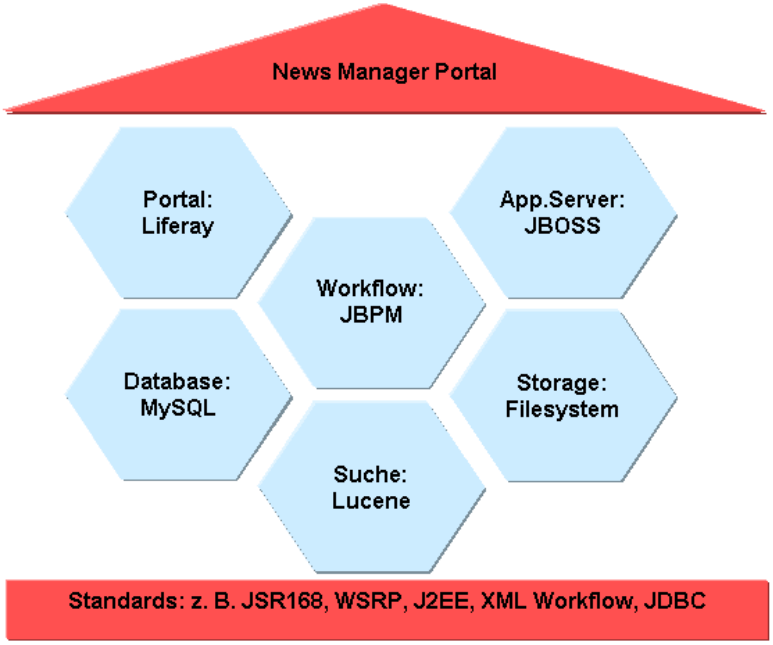

Abbildung 2.2.: Aufbau nach Komponenten-Prinzip

- Bei Dossiers handelt es sich um thematische Dokumentenzusammenstellungen die meistens themenbezogen (z.B. Anfrage) erstellt wird. Sie kann mehrere ausgewählte Artikel, Dokumente, etc. beinhalten (Abbildung 2.6).

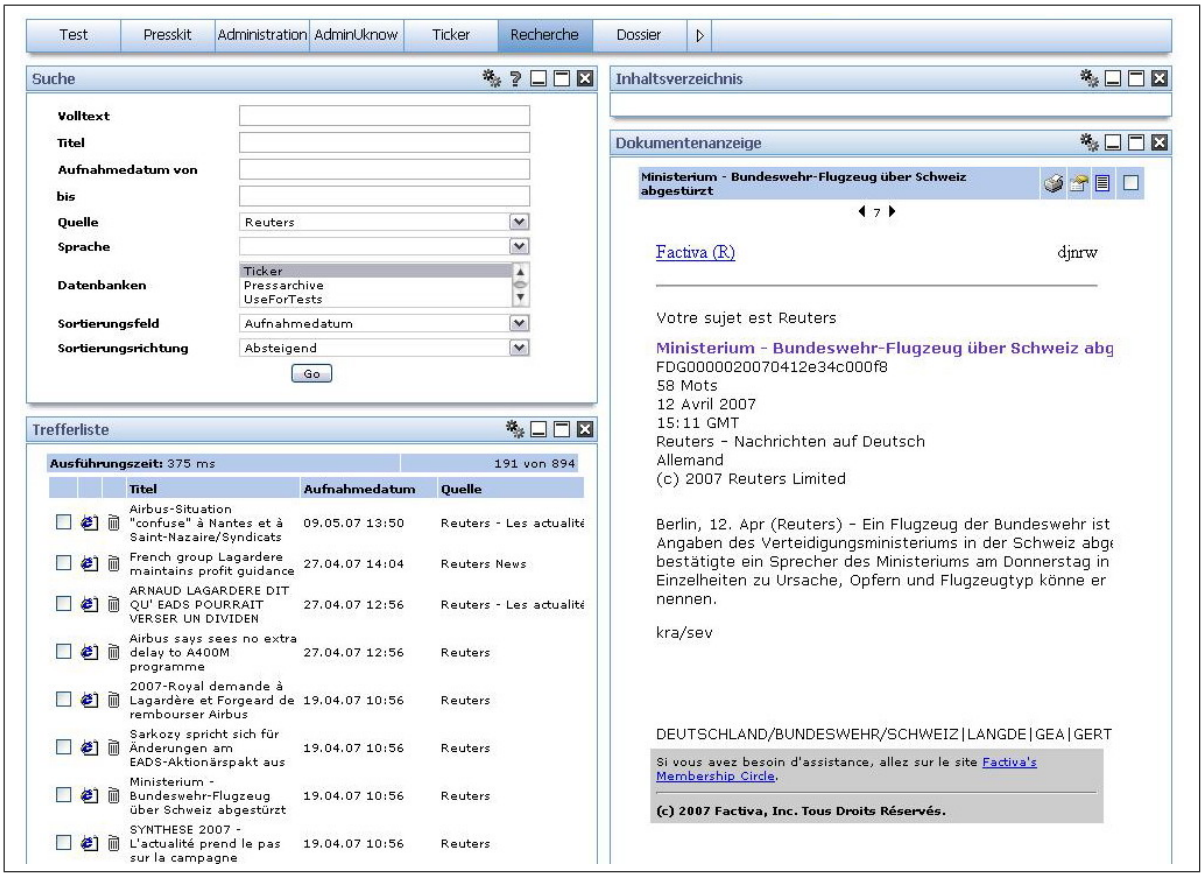

Abbildung 2.3.: Beispielansicht der Portalanwendung mit benutzerdefinierten Portlets

Abbildung 2.4.: Portlet ”Volltextsuche”

2.2. Zielsetzung

In Rahmen dieser Abschlussarbeit soll eine Weiterentwicklung des Pressespiegels (als eigenständiges Modul) zur automatischen Identifikation von aktuellen Themen begonnen werden.

Das Projekt wird in mehrere Phasen unterteilt:

1. Konzipierung und Implementierung eines Basissystems zur sequentieller Datenverarbeitung (Workflow);
2. Analyse der verdichteten Indexdaten;
3. Entwurf von Extraktionsstrategien;
4. Beurteilung der Ergebnisse;
5. Erweiterung des Funktionsumfangs (allgemeine morphologische Filter, Klassifikationsmechanismen, usw.);
6. Integration in den bestehende News Manager System;

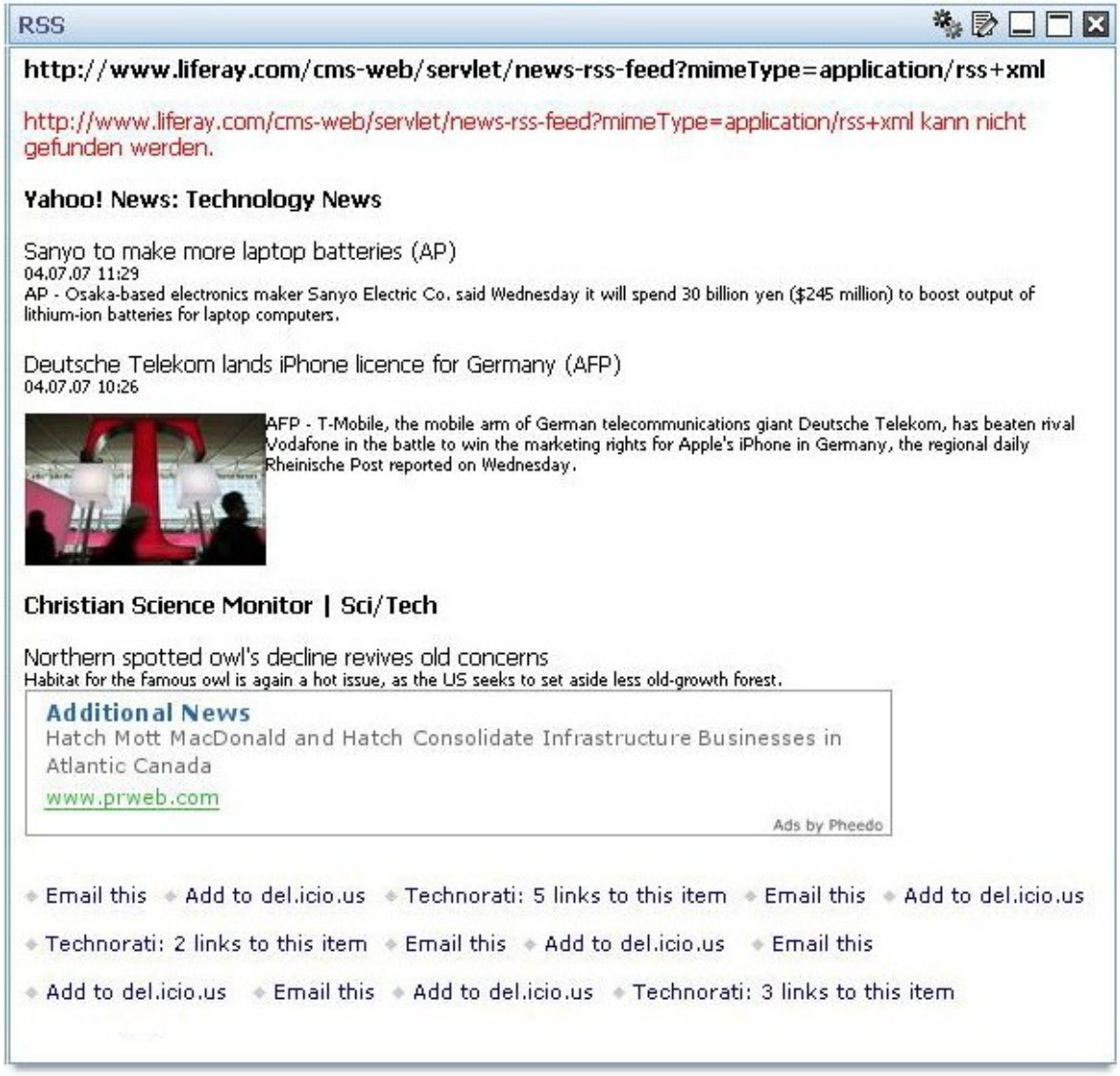

Abbildung 2.5.: Portlet ”RSS-Feed”

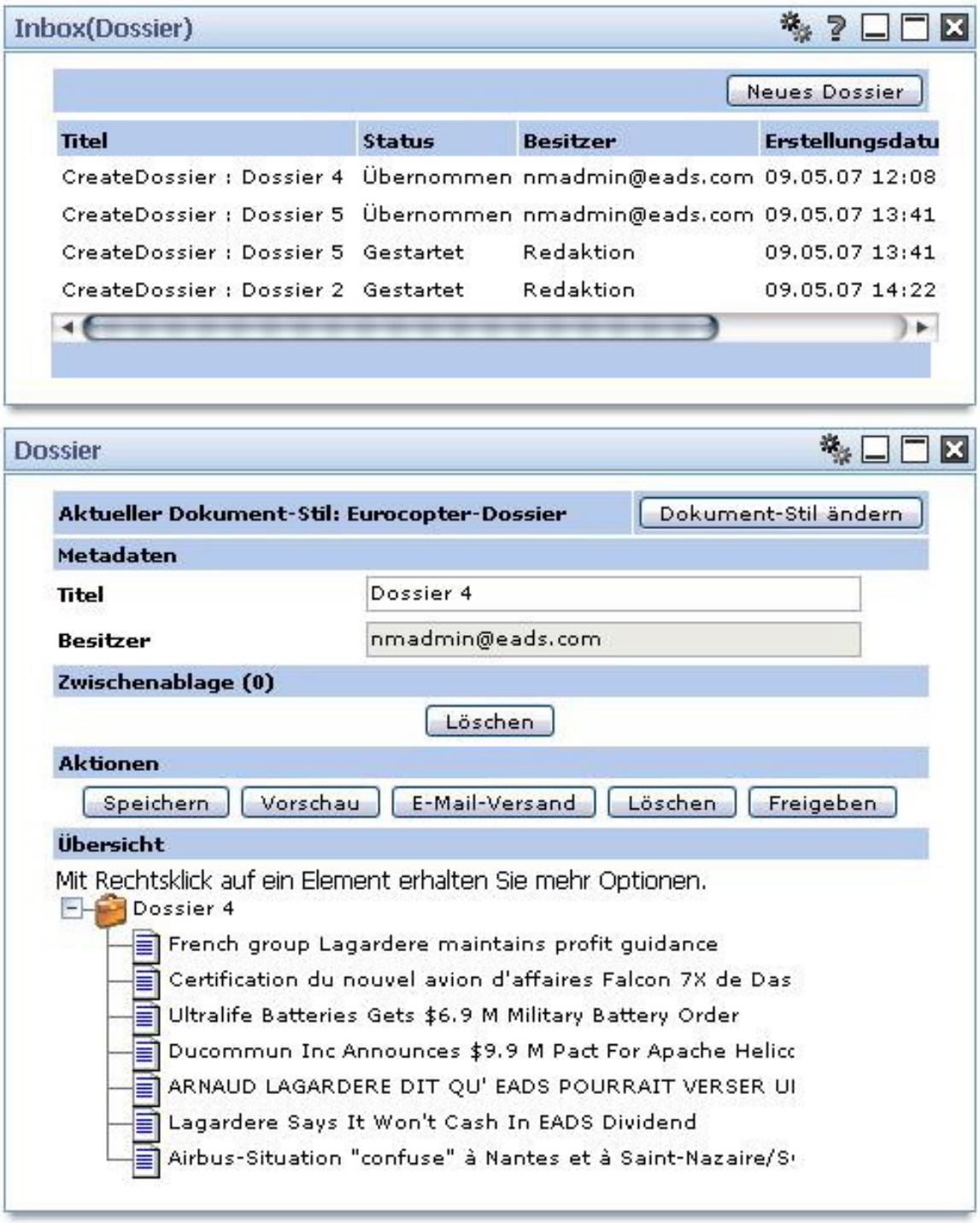

Abbildung 2.6.: Portlet ”Dossier”

7. Kundenspezifische Anpassung (Klassifikationsmerkmale, weitere Filter und morphologische Module).

Erfüllung von Phasen 1 bis 4 ist Ziel dieser Abschlussarbeit.

2.3. Anforderungsanalyse

2.3.1. Typischer Kunde

Insgesamt orientiert sich die Anwendung an Unternehmen mit hoher Medienpräsenz, mit fortgeschrittener Infrastruktur und eigener Pressestelle bzw. Abteilung für Medienbeobachtung. Meistens sind es europa-/weltweite korporative Unternehmen aus verschiedenen Branchen. Sie haben gesetzliche Kommunikationsverpflichtungen, die sie erfüllen müssen (d.h. sie sind Börsennotiert, beispielsweise Volkswagen, EADS) oder ihr Profil ist stark mit Medienbeobachtung verbunden (Bsp.: RTL-Gruppe, Bundespresseamt). Dem entsprechend sind auch individuelle Anpassungen der Anwendung nötig: einige interessieren sich an gesamtes Spektrum, andere - nur an Branchen-/Unternehmensbezogene Mitteilungen.

Als Projektbeispiel wird eine weltweitoperierende Maschinenbau-Firma XY mit eigener Abteilung für Medienbeobachtung definiert.

Firmenstruktur ist definiert wie in Abbildung 2.7

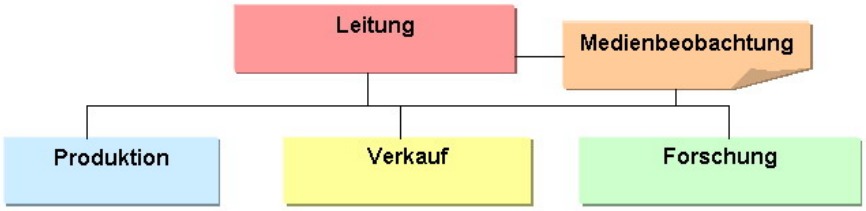

Abbildung 2.7.: Verbindungen in Unternehmensstruktur bezüglich Medienbeobachtung

Die Zahl der einlaufenden Nachrichten mit Hunderte pro Tag abgeschätzt.

Definierte Benutzerrollen: **Administrator, Redakteur, Leser**.

2.3.2. Benutzeranforderungen

Die Software soll Tagesthemen automatisch erzeugen und darstellen können:

- Liste aktueller Themen muss übersichtlich sein um eine gute Benutzerakzeptanz zu erreichen.
- Aktuelle Themen müssen klar und verständlich formuliert werden.

- Die Software muss eine Mittel zum Aufruf themenbezogenen Nachrichten bereitstellen um schneller Zugang zu den Nachrichtentexten zu gewährleisten.

Für spezielle Benutzer (Redakteure) muss es eine Möglichkeit geben Themenerzeugung und Ausgaben zu konfigurieren:

- Für ein Redakteur muss es möglich sein, Zeitintervall, für den die Themen erzeugt werden, zu ändern (Stunden, Tagen, Monate).
- Die Software muss über einen Mittel verfügen, das es einem Redakteur erlaubt eine neue Themensuche manuell anzustoßen.
- Für den Redakteur soll es eine Möglichkeit geben den Erscheinungstext der Themen anders zu formulieren.

2.3.3. Systemanforderungen

Nichtfunktionale Anforderungen:

- Das System muss in der Lage sein eine Nachrichtenmenge in Großenordnung ungefähr 1000 Nachrichten pro Tag abarbeiten zu können
- Die Software muss sich leicht in die bestehende Portalanwendung integrieren lassen und bestehende Datenformate, Datenbankkonfigurationen und Standarte der Darstellung unterstutzen.
- Bei der Verwendung von externer Bibliotheken sollen vorzugsweise Open-Source- oder Freeware-Komponente eingesetzt werden.
- Designarchitektur, sowie Speicherungsformate müssen an entscheidenden Stellen freikonfigurierbar, nachvollziehbar und erweiterbar sein.

Funktionale Anforderungen:

- Aktuelle Tagesthemen sollen aufgrund eingehenden Nachrichten in einem regelmäßigen Abstand neu ermittelt und in der Darstellung aktualisiert werden.
- Das System soll zwischen 2 Benutzertypen unterscheiden: Leser und Redakteur.
- Für verschiedene Benutzertypen müssen aktuelle Themen anders dargestellt werden:
  - Ein Leser hat eine Listenansicht, in der die aktuellen Themen in Form anklickbaren Verknüpfungen dargestellt werden. Beim Anklicken werden themenbezogene Nachrichten angezeigt. Einstellungsmöglichkeit: Listengröße.
  - Ein Redakteur hat zusätzlich zur Lesersicht folgende Einstellmöglichkeiten:

  - Zeitintervall zur Themenerfassung,
  - Zeitintervall zur Themenerzeugung,
  - Feld zur Veränderung automatisch erzeugter Thementitel,
  - Schaltfläche für direktes Anstoßen des Themenerzeugungsprozesses,
  - Reihenfolge von Thementitel in der Liste.
- Es müssen mehrere Instanzen den Themenlisten mit verschiedenen Einstellungen unterstutzt werden (z.B. Tagesthemen von heute, Tagesthemen von gestern, Wochenthemen...)

2.3.4. Anwendungsfälle

Name	Aktuelle Nachrichten
Kurzbeschreibung	Ein Leser sucht sich aktuelle Nachrichten aus
Auslöser	Ein Leser möchte aktuelle Informationen bekommen
Ergebnis	Der Leser bekommt nötige Informationen
Akteure	Leser
Eingehende Information	keine
Vorbedingungen	Themenliste von Redakteur konfiguriert und von System erzeugt
Nachbedingungen	keine
Wesentliche Schritte	Benutzer als Leser identifizieren Listengröße für diesen Bestimmten Leser prüfen Themenliste gemäß Listengröße anzeigen Konkretes Thema bestimmen Themenbezogene Nachrichten anzeigen Konkrete Nachricht bestimmen Nachricht anzeigen

Tabelle 2.1.: Anwendungsfall ”Aktuelle Nachrichten”

Name	Themenerzeugung
Kurzbeschreibung	Ein Redakteur startet neue Themenerzeugung
Auslöser	Ein Redakteur möchte Themenliste aktualisieren
Ergebnis	Aktualisierte Themenliste steht zur Verfügung
Akteure	Redakteur
Eingehende Information	Themenerzeugungsintervall
Vorbedingungen	keine
Nachbedingungen	Themenliste erzeugt
Wesentliche Schritte	Benutzer als Redakteur identifizieren Redaktoreinstellungen prüfen Themenliste anzeigen Konkretes Thema bestimmen Starten der Themenerzeugung bestimmen Themenerzeugung durchführen Neue Themenliste anzeigen

Tabelle 2.2.: Anwendungsfall ”Themenerzeugung”

2.4. Kundensicht

Die Kundenanforderung wird nachfolgend in einem allgemeinem Prozessablauf skizziert und beschrieben.

In Pool ”Rohdaten” werden kontinuierlich neue Nachrichten eingestellt. Diese sollen zur Bearbeitung an einen Textprozessor gesendet werden. Die Prozessschritte gliedern sich wie folgt:

- Jede Nachricht wird mit einem Parser in eine strukturierte Form mit charakteristischen Abschnitten (typischerweise Titel, Quelle, Datum, Text) gebracht. Als strukturierte Form soll hier XML verwendet werden;
- Workflow ”Filter” dient der Verdichtung der Daten. Hier sollen:
  - Stopwörter (wie ”der”, ”und”, ”aber”, etc.) entfernt werden;
  - ein Stammwort-Mapping durchgeführt werden;
  - die Relevanz bzw. Zuverlässigkeit der Nachricht (z.B. anhand der Quelle) beurteilt werden (nicht in Rahmen dieser Arbeit);
  - Synonyme ausgewertet werden;
  - weitere Filter einsetzbar sein können;

Workflows sollen frei konfigurierbar sein. Basis der Verarbeitung stellt die JBPM-Workflow-Engine dar;

- Der Inhalt der Nachricht wird indexiert, alle vorhandene Terme werden in der Datenstruktur ”Verdichtete Indexdaten” gespeichert, Indexierungsmechanismen wird Apache Lucene bereitstellen;
- Die Datenstruktur ”Verdichtete Indexdaten” soll indexierte Daten mit Zeitstempel und Häufigkeit enthalten;
- Die Datenextraktion erfolgt in regelmäßigen Abständen (sollte aber auch direkt angestoßen werden können);
- Der Extraktor erstellt abhängig vom Zeitintervall (Tages/Wochen/Monatsthema) eine Liste von Wörtern mit entsprechender Gewichtung (in dieser Arbeit - nur aufgrund der Häufigkeit in betrachtetem Zeitintervall);
- Eine bestimmte Anzahl von gefundenen Wörtern (List ”Top Ten”) wird an den ”Searcher” übergeben. Dieser Sucht im Pool ”Rohdaten” nach einzelne Wörtern und deren Kombinationen und liefert eine Liste der Nachrichten nach Relevanz sortiert;
- Ein Klassifikator erzeugt aus der Liste ”Relevante Nachrichten” eine Liste von relevanten Themen. In Rahmen der Arbeit soll die Themenformulierung durch Übernahme des Titels eines Dokumentes aus der Gruppe erfolgen. Andere Dokumente sollen als Verweis unterhalb des jeweiligen Themas dargestellt werden.

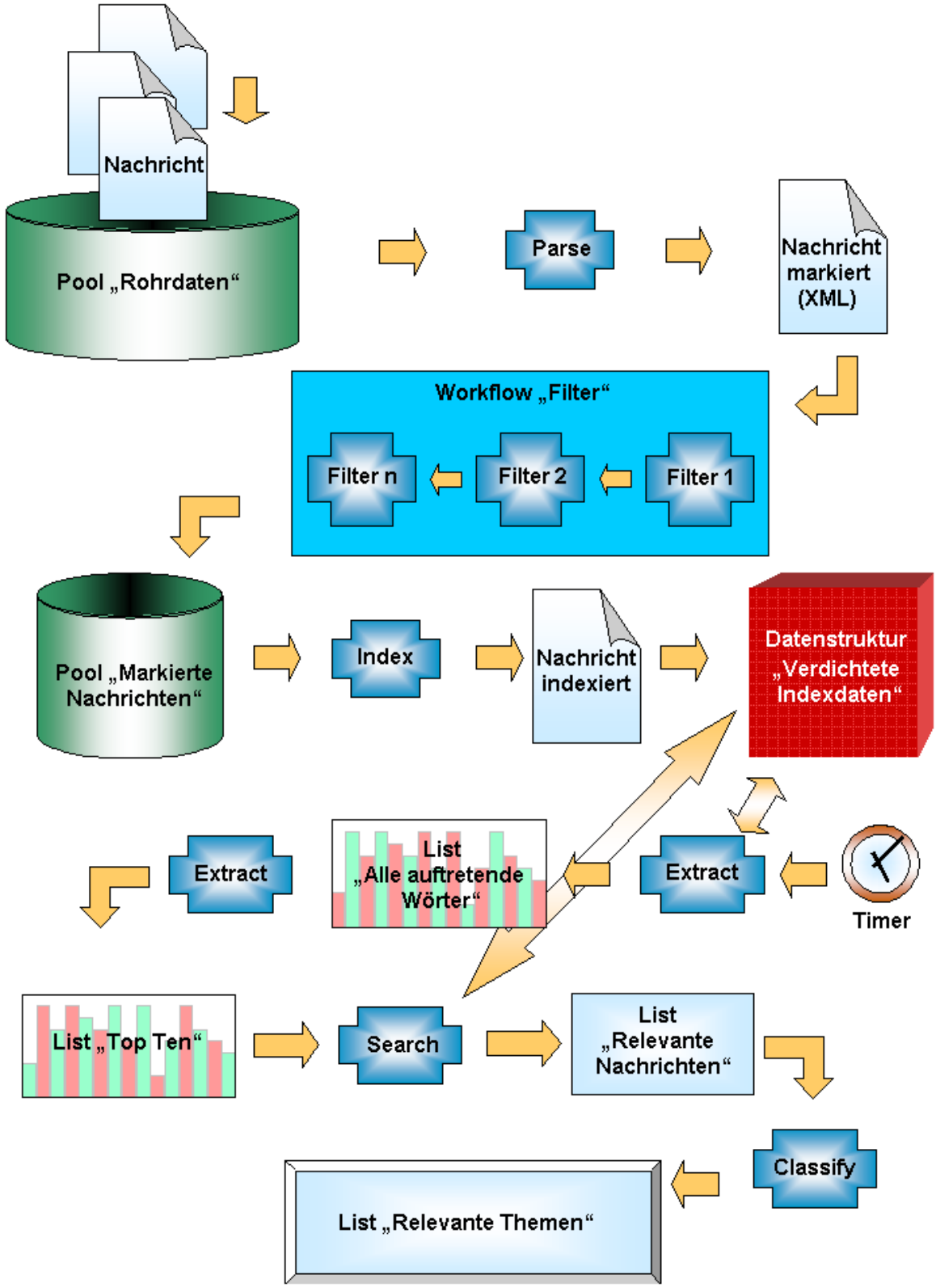

Abbildung 2.8.: Prozessablauf-Skizze

3. Entwurf

3.1. Grundlagen

3.1.1. Information Retrieval

Die Aufgabe des Projekts gehört zum Teilgebiet der Informationswissenschaft ”Information Retrieval” (Informationswiedergewinnung). Es ist ein Fachgebiet, das sich mit computergestütztem inhaltsorientiertem Suchen beschäftigt. Die klassische Definition suggeriert, dass Informationen in großen Datenbeständen verloren sind. Sie müssen wieder gewonnen bzw. wieder gefunden werden. Dabei ist die Hauptaufgabe nicht nur einen genau passenden Text zum Begriff zu finden, sondern eine Menge ähnlichen Dokumente zum Thema, aus den dann die am besten passende ausgewählt werden sollen. [Rijsbergen (1979)].

3.1.2. Apache Lucene

Als Grundsystem für Information Retrieval im Projekt wurde einen Suchmaschinen-Framework ”Apache Lucene” eingesetzt. Gründe dafür sind:

- Lucene ist ein Projekt der Apache Jakarta Gruppe - also Open Source.
- Es ist modular, leicht zu integrieren und hat wohldefinierte Schnittstellen. Mit der offiziellen online Dokumentation und grundlegenden Java Kenntnissen, ist es ein leichtes Lucene in bereits bestehende Systeme und Anwendungen zu integrieren und zu nutzen.
- Lucene benötigt extrem wenig Systemressourcen. Das betrifft sowohl Plattenplatz, als auch Prozessor und Arbeitsspeicher.
- Es bietet einen Suchindex, der an Performance und Funktionalität so manches kommerzielle Produkt in den Schatten stellt.
- Da es ein Open Source ist, bietet es Möglichkeit die Indexierungs- und Suchfunktionen an die eigenen Bedürfnisse anzupassen (Lucene ist komplett in Java geschrieben).

Doug Cutting, der Erfinder von Lucene, ist in der Welt des Information Retrieval kein Unbekannter. So zeichnet Cutting verantwortlich für den Index-Kern von Apples Suchhilfe ”Sherlock”. Beim Suchmaschinenbetreiber eXite war er maßgeblicher Architekt der hauseigenen Technik. Die erste Idee zu Lucene kam ihm 1997. Erste Prototypen waren 1998 verfügbar. Mit Lucenes Umzug 2001 von Sourceforge unter das Dach der Apache Foundation war vor allem die Hoffnung verknüpft, dass Lucene eine breitere Öffentlichkeit erreichen würde und die Weiterentwicklung des Projekts gesichert sei. Seit Mitte Februar 2005 ist Lucene ein Top-Level-Apache-Projekt. Mehr zu Lucene - in [Hatcher und Gospodnetic (2004)].

Indexierung

Sie hat das primäre Ziel möglichst schnell jene Stellen zu identifizieren an denen ein bestimmtes Wort steht. Lucene nutzt einen umgekehrten Wortindex, das für kompakte Index-Größe sorgt. Denn eine Suchmaschine definitionsgemäß für große Datenmengen eingesetzt wird. Beim umgekehrten Wortindex wird jede Fundstelle eines Wortes mit dem Wort als Schlüssel gespeichert. Für die Indexierung stellt Lucene Klassen und Methoden bereit, um Dokumente zu analysieren, zu indexieren und zu speichern. Im Grunde kann alles verarbeitet werden was in Textform vorliegt.

Wenn die eigentlichen Textinhalte vorliegen, werden sie einem so genannten Analyzer übergeben. Die Daten werden nach fest definierten Regeln zerlegt und in einer einheitlichen Weise für den Index aufbereitet. Dabei können Stoppwörter eliminiert oder Zeichenketten in Kleinschreibung normalisiert werden. Eine Stopliste enthält Worte wie ”und”,”dass” oder ”der”, die einerseits sehr häufig vorkommen, andererseits keine Relevanz für die Suche haben. Sie blähen den Index nur unnötig auf. Für einfache Aufgaben bringt Lucene einige Simple Analyzer mit.

Nach der Analyse erfolgen die eigentliche Indexierung und das Abspeichern des aufbereiteten Suchraums. Lucene unterscheidet die Indexfelder ”indexed”, ”tokenized” und ”stored”. Diese Attribute, die sich auch kombinieren lassen, geben an ob ein Feld durchsuchbar (indexed) sein soll, ob der Eingabestream durch einen Analyzer normalisiert (tokenized) wird und schlussendlich, ob das Feld in seiner Originalform im Index abgespeichert (stored) werden soll.

Suche

Dank des Suchindex ist man in der Lage sehr schnell jene Dokumente zu finden, die ein bestimmtes Wort enthalten. Prinzipiell könnte man also den Benutzer auffordern, ein oder mehrere Wörter zu nennen, um ihm daraufhin die Fundstellen zu präsentieren.

Jedes atomare Wort wird zunächst wie auch bei der Erstellung des Suchindex vom gleichen Analyzer bearbeitet. Für jedes Wort wird eine Ergebnisliste aus dem Index ermittelt. Diese Ergebnislisten werden nun entsprechend der Regeln des Syntaxbaums miteinander in Verbindung gebracht. Bei einer UND-Operation wird die Schnittmenge gebildet, bei ODER werden beide Listen vereinigt.

Zusätzlich zur Fundstelle können im Index noch weitere Informationen abgelegt werden, die eine Bewertung der Fundstelle erlauben (z.B. Datum oder Informationsquelle). So können Treffer, die relevanter für ein bestimmtes Wort sind, im Suchergebnis weiter oben angezeigt werden.

Bei Lucene wird für die Bewertung eines Treffers die Häufigkeit des Wortes im betreffenden Dokument herangezogen. Ein Indexeintrag besteht bei Lucene daher neben Wort und Fundstelle auch aus der Häufigkeit des Wortes im betreffenden Dokument. Um zu verhindern, dass sehr große Dokumente, die naturgemäß viel mehr Worte enthalten, kleinen Dokumenten gegenüber bevorzugt werden, wird dabei die relative Häufigkeit genutzt. Die relative Häufigkeit ("Score") berechnet sich aus der absoluten Häufigkeit geteilt durch die Gesamtanzahl an Worten im Dokument. Für die ganze Abfrage kann der Score-Mechanismus durch folgende Formel beschrieben werden:

$$score_d(q, d) = \sum_{t \in q} x(t, d) * b_t$$

Erklarung: Die Score eines Treffers d aus der Summe aller Bewertung x fur die Terme und Phrasen t der Anfrage q unter Beachtung der Boost-Faktoren b_t (Boost-Faktor, die Gewichtung, kann für jeden Term in der Anfrage vordefiniert werden).

3.1.3. Clustering

Die Erstellung von Thementitel basierend auf gefundener Dokumentenmenge. Sie erfolgt im Projekt durch gruppierung ähnlicher Dokumente mittels Clustering-Verfahren.

Ein Clustering (Clusterbildung) in allgemeinem ist die Klassifizierung von Objekten in Gruppen (Cluster), so dass innerhalb einer Gruppe die Objekte möglichst homogen, die Gruppen untereinander aber möglichst heterogen sind. Ausreißer sind Objekte, die keinem der gefundenen Cluster angehören.

Die Methoden des Clustering's stammen ursprünglich aus dem Bereich Statistik, wo sie auf numerische Daten angewendet werden. Aber mit gewaltigem Zuwachs der Menge von Textdaten (besonders durch das Internet) werden diese Methoden auch in Computerlinguistik eingesetzt. Eine Übersicht von Text-Clustering Algorithmen kann man sich im [Osinski u. a. (2004)] verschaffen.

Zwei wesentliche Clustering-Typen sind hierarchisches und partitionelles (K-means) Clustering. Des Weiteren werden für Clustering-Zwecken neuronale Netze benutzt (Self Organizing Maps). Für das Projekt wurde hierarchisches Clustering eingesetzt. Es ist zwar von qudratischer Aufwand $O(n^2)$ gegen linearen bei partitionellen Verfahren $O(n * (logn))$, das ist aber bei der übersichtlichen Anzahl von Dokumenten nicht ausschlaggebend, dafür ist die Qualität besser und es ist einfach zu implementieren.

Gewöhnlich erfolgt hierarchisches Clustering (”Single Link”) durch konsequente Zusammenlegung ähnlicher kleiner Cluster in größeren. Das Verfahren wiederholt sich zyklisch und kann dann beendet werden, wenn alle Cluster eine bestimmte Distanz zueinander überschreiten (oder wenn eine bestimmte Anzahl von Clustern erreicht ist). Das Endergebnis des Algorithmus ist ein Baum von Clustern (genannt Dendrogramm), das die Verwandtschaft von Clustern darstellt. Durch das Abschneiden der Dendrogramm auf einer gewünschten Ebene erfolgt Clustering der Daten in disjunkte Gruppen. In dem Beispiel der Abbildung 3.1 werden nach dem Abschneiden auf der Distanz h_1 6 und nach dem Abschneiden auf Distanz h_2 4 Cluster erzeugt.

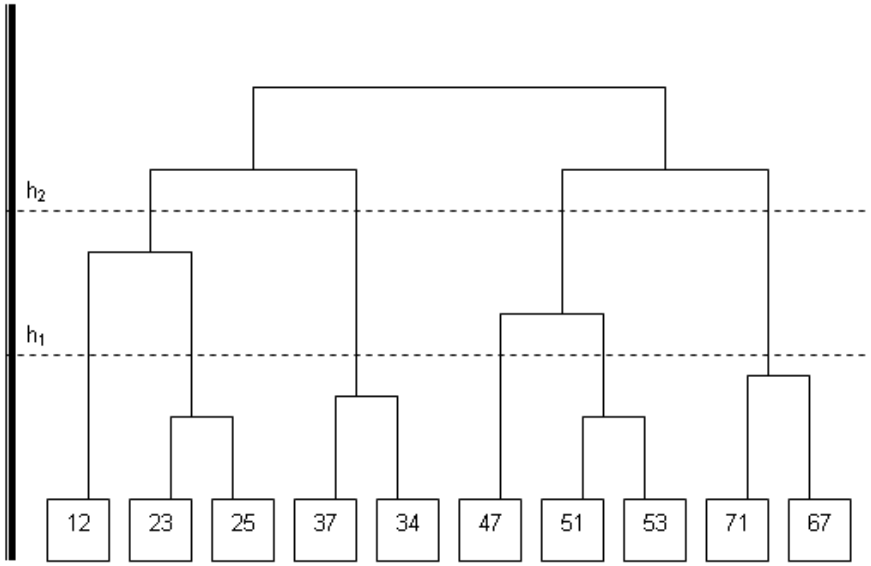

Abbildung 3.1.: Hierarchisches Clustering: Beispiel

3.2. Architektur

Da das Gesamtsystem modular aufgebaut ist wird die neue Anwendung im Newsmanager-Portal als ein Modul innerhalb des Paketes ”Informationsdienste” integriert (Abbildung 3.2).

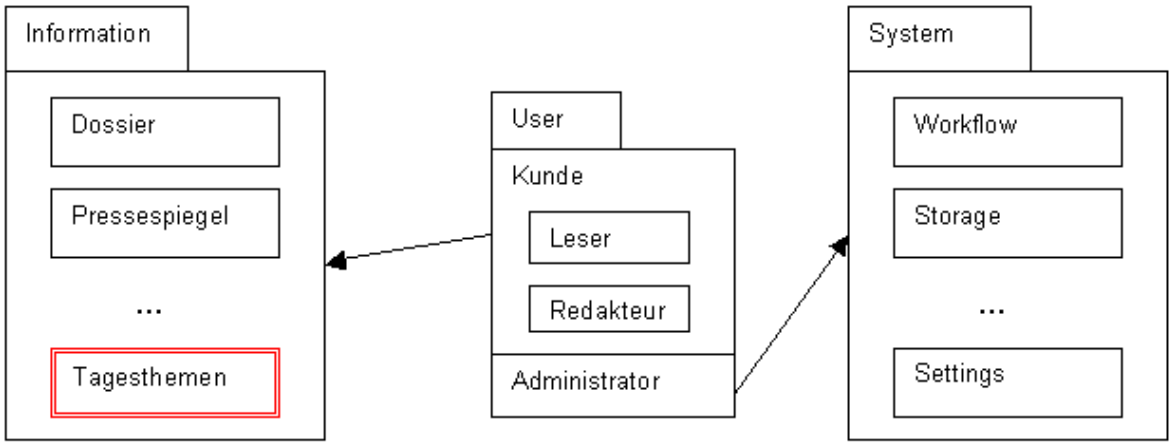

Abbildung 3.2.: Anwendung ”Tagesthemen”: Platzierung im Gesamtsystem

Das Anwendungsdesign stützt sich auf die Prozessbeschreibung aus der Kapitel ”Analyse”. Im Wesentlichen besteht das Gesamtsystem aus zwei Teilen - dem Präprozessor und dem Themen-Extraktor - die gemeinsam als ”Producer-Consumer” Schema agieren. Für den Endkunden wirkt der Themen-Extraktor als Fassade des Gesamtsystems (Abbildung 3.3).

3.2.1. Präprozessor

Der Präprozessor stellt eine einfache Pipe-Struktur dar, die aus mehreren Modulen mit gleichem Interface besteht. Prinzipiell sind nur drei Datenverarbeitungsstufen notwendig: Import, Vorbereitung, Verdichtung (Abbildung 3.4). Die Aufgabe des Import-Moduls ist es die eingehende Daten in anwendungsspezifischer Format zu konvertieren. Die Vorbereitungsstufe bestehet aus verschiedener Filter (z.B. morphologische). In der Verdichtungsstufe werden redundante Daten zu einem indexierten Vokabular geschrumpft.

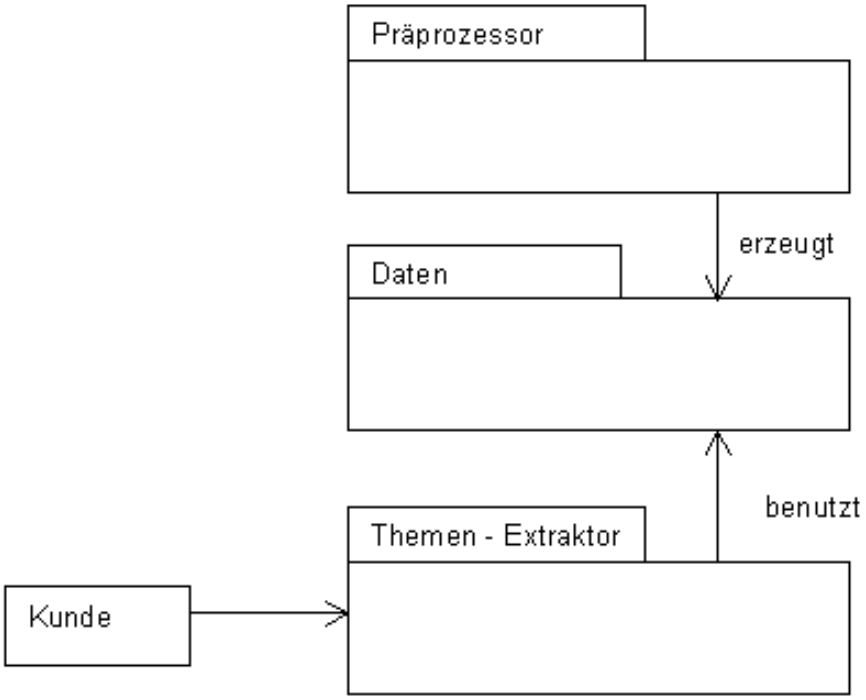

Abbildung 3.3.: Tagesthemen: Allgemeine Struktur

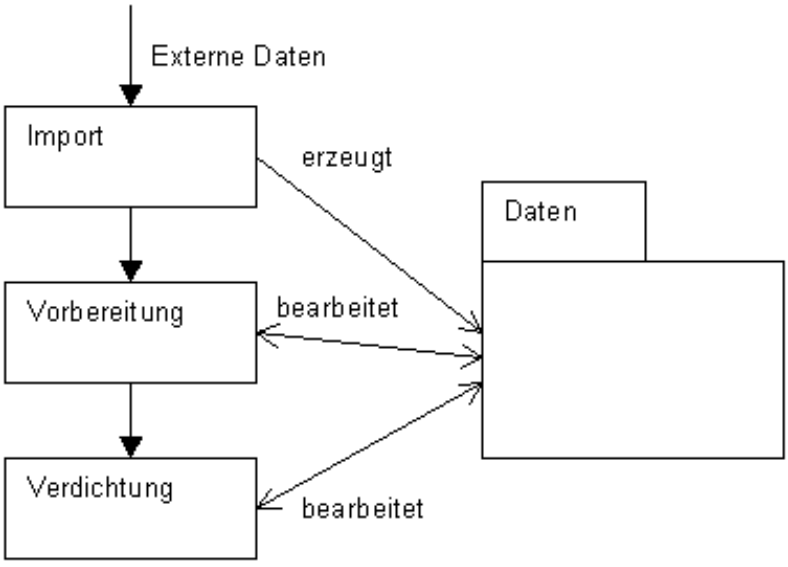

Abbildung 3.4.: Präprozessor: Verarbeitungs-Pipe

3.2.2. Themen-Extraktor

Im Themen-Extraktor ist der Modul ”Klassifikator” die Fassade, die andere Module implizit benutzt. Wesentliche Bestandteile des Themen-Extraktors sind auch der ”Term-Extraktor”, der direkt mit Daten arbeitet und die am häufigsten auftretende Wörter sucht, und der ”Dokumentenfinder”, der aufgrund gefundener Wörter eine Liste entsprechender Dokumente erzeugt. der Klassifikator selbst übernimmt die Gruppierung der Dokumente (Clustering) nach Themen und die Erzeugung von Thementitel (Abbildung 3.5).

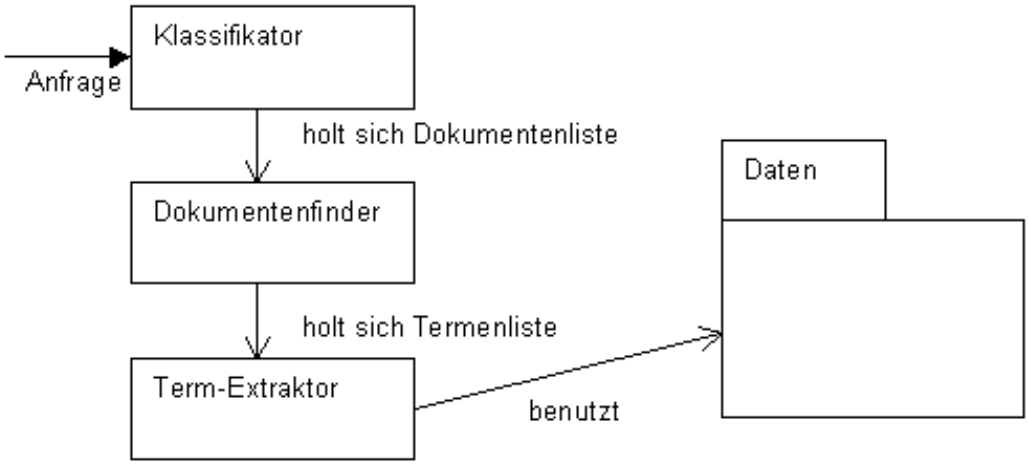

Abbildung 3.5.: Themen-Extraktor: Fassaden-Struktur

3.3. Realisierung

3.3.1. Allgemeines

Da für die Speicherung der Daten XML-Dateien benutzt werden, wird eine Adapter-Klasse XMLProcessor erstellt:

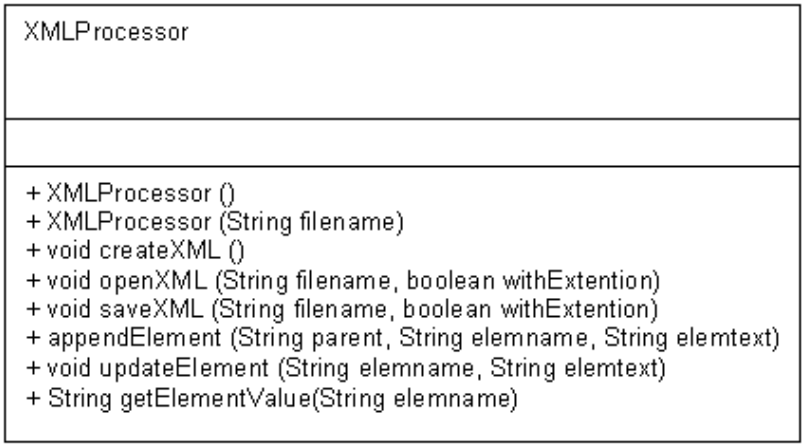

Abbildung 3.6.: Class ”XMLProcessor”

3.3.2. Präprozessor

Alle aktiven Prozessbestandteile des Präprozessors werden zu Klassen, die vom `AbstractDocumentProcessor` erben:

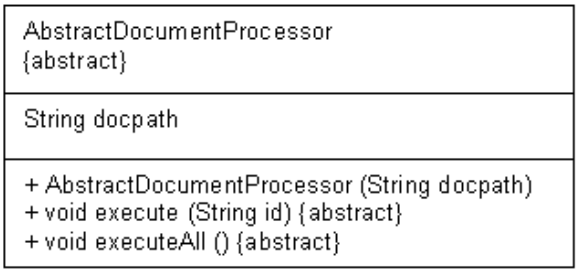

Abbildung 3.7.: Class ”AbstractDocumentProcessor”

Das vereinfacht spätere Ergänzung um weitere Bearbeitungsmodule.

Ein wesentlicher Unterschied zwischen Produktions- und Prototypemodell bestehet darin, dass für die Entwicklung eine schon vorhandene Menge an Dokumenten verwendet wird. Im Produktivsystem (siehe Prozessablauf auf der Abbildung 2.8) werden einzelne Nachrichtentexte ”live” in das System eingestellt. Die Methode ”execute” soll dabei mit einzelnen Dokumenten (nach Dokumenten-id) arbeiten und ”executeAll” mit der ganzen vorhandenen Dokumentenmenge. Die zweite Methode wird für Entwicklungs-/Testzwecke genutzt (bzw. für das Initialisierungs-Stadium bei der Integration in ein bestehendes System). Die Variable ”docpath” enthält Pfad zum Dokumentenverzeichnis.

Importeur

Die Aufgabe des Importeurs ist es Nachrichten verschiedener Quellen in allgemeingültiges XML-Format mit folgender Struktur zu konvertieren:

```
<?xml version="1.0" encoding="UTF-8"?>
<DOCUMENT>
    <DOCID></DOCID>
    <TITLE></TITLE>
    <SOURCE></SOURCE>
    <LANGUAGE></LANGUAGE>
    <DATE></DATE>
    <BODY></BODY>
    <METADATA></METADATA>
</DOCUMENT>
```

Bedeutung der Felder:

DOCID eine eindeutige Identifikation des Dokuments in der Datenbank.

TITLE Nachrichtentitel

SOURCE Quelle (Presseagentur)

LANGUAGE Sprache der Nachricht (Sprachenkürzel nach ISO 639-1 wie ”en”, ”de”)

DATE Erstellungsdatum (wann wurde die Nachricht verfasst)

BODY Text der Nachricht (nicht formatiert)

METADATA Spezialfeld, wird für die Filterung und Indexierung genutzt.

Die Importeur-Klasse in dieser Arbeit (DocumentImporter) ist denkbar einfach, da die Dokumente schon in einer von Firma uknow vorgefertigter Datenbank abgelegt wurden. Als Initialwert für die METADATA - Feld werden TITLE und BODY zusammen verknüpft. Somit wird eine zusätzliche Gewichtung von den Termen erreicht, die im Titel und im Nachrichtentext übereinstimmen.

Filter

Der Filter bereitet das Feld ”METADATA” für die Indexierung vor, das bedeutet, dass nicht relevante Texteinheiten entfernt (Stopwörter, Sonderzeichen usw.) bzw. angepasst werden (z.B. durch ihre Grundform ersetzt).

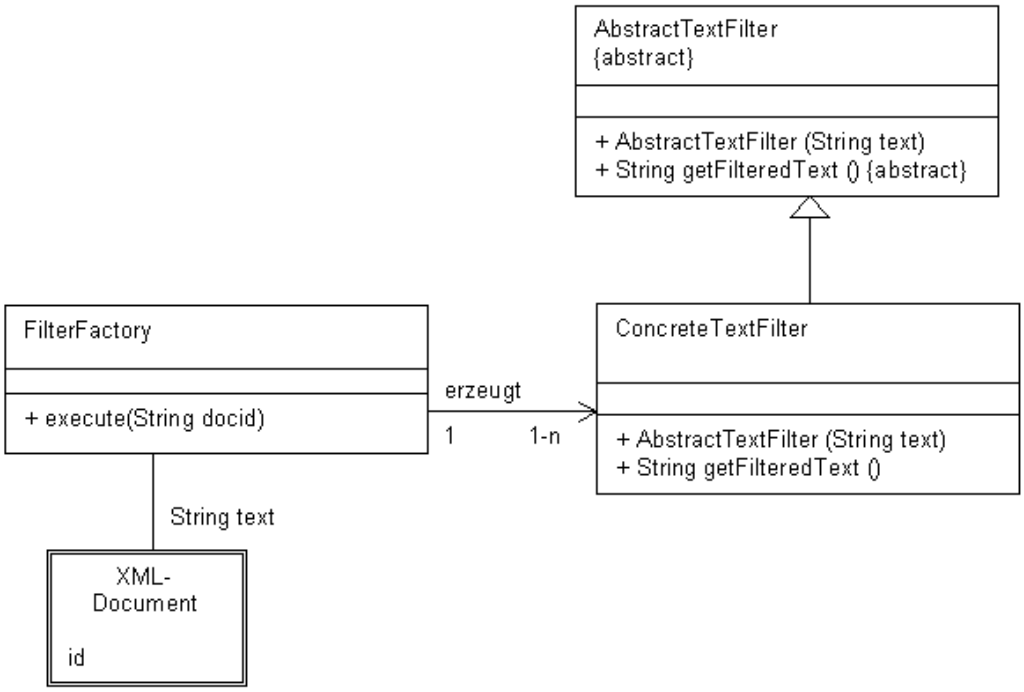

Abbildung 3.8.: ”Filter Factory”

Die Reihenfolge des Filterungsprozesses und der entsprechende Filterungsklassen werden mit einer einfacher Workflow - XML sprachenabhängig konfiguriert:

```
<?xml version="1.0" encoding="UTF-8"?>
<WORKFLOW>
```

```
    <FILTER>
        com.uknow.filters.CustomPhraseFilter
    </FILTER>
    <FILTER>
        com.uknow.filters.SpecialSignFilter
    </FILTER>
     <FILTER>
        com.uknow.filters.NumberFilter
    </FILTER>
    <FILTER>
        com.uknow.filters.MultipleWhiteSpaceFilter
    </FILTER>
</WORKFLOW>
```

Die Filterungsklassen müssen dabei von einer abstrakter Klasse `AbstractTextFilter` erben:

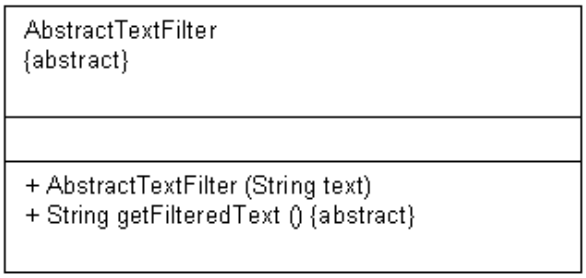

Abbildung 3.9.: Class ”AbstractTextFilter”

Die Filterungsklassen werden im `DocumentFilter` mit Hilfe von `java.lang.Class` und `java.lang.reflect` instanziert.

Indexer

Der Indexer dient der effizienter Verschlagwortung (Indizierung) des ”METADATA” - Feldes eines Dokumentes für die Zwecke einer späterer statistischen Bearbeitung, Abschätzung und Wiederfindung von Begriffen und Texten. In dieser Arbeit wird eine sogenannte freie Indexierung benutzt (Volltextindexierung), so wie sie auch in Suchmaschinen verwendet wird, so dass fertige Mechanismen eines Open-Source-Systems (Apache Lucene) können eingesetzt werden. Im Index werden auch die Dokumentenfelder ”DATE” und ”LANGUAGE” gespeichert.

3.3.3. Themen - Extraktor

Termextraktor

Er erzeugt eine sortierte ”Top Ten” - Liste (`java.util.ArrayList`) der am häufigsten auftretender Schlagwörter (Terme) anhand der indexierten Daten für die vorgegebene Sprache und das gewählte Zeitintervall.

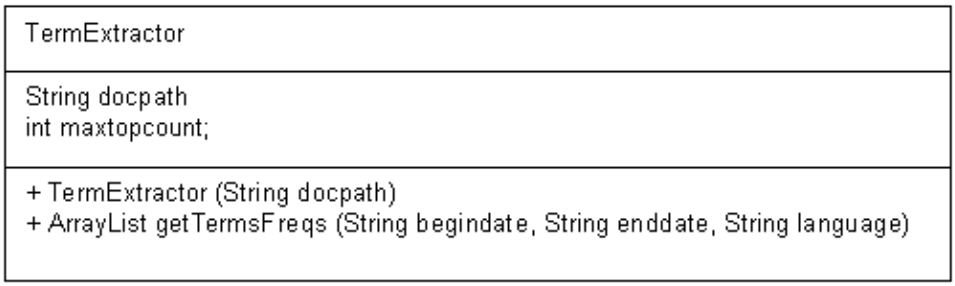

Abbildung 3.10.: Class ”TermExtractor”

Für den Vergleich der Terme - Gewichtung wird eine Objektklasse angelegt, die Interface `java.lang.Comparable` implementiert und somit eine Sortierung von Terme nach Häufigkeit des Auftretens ermöglicht:

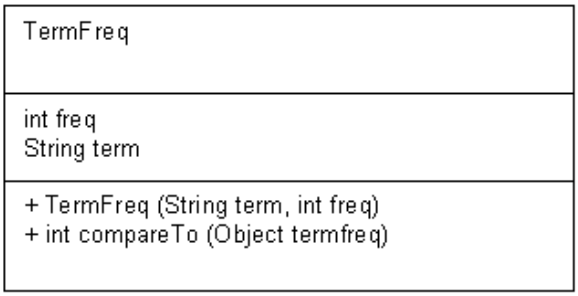

Abbildung 3.11.: Class ”TermFreq”

Dokumentenfinder

Er sucht anhand der Term - Liste die zutreffende Dokumente. Dabei wird die gleiche Suchmaschine benutzt, die für die Erstellung des Indexes verantwortlich war. Die Suchanfrage

ergibt sich aus den Termen der ”Top Ten” - Liste, die durch logisches ODER verknüpft sind.

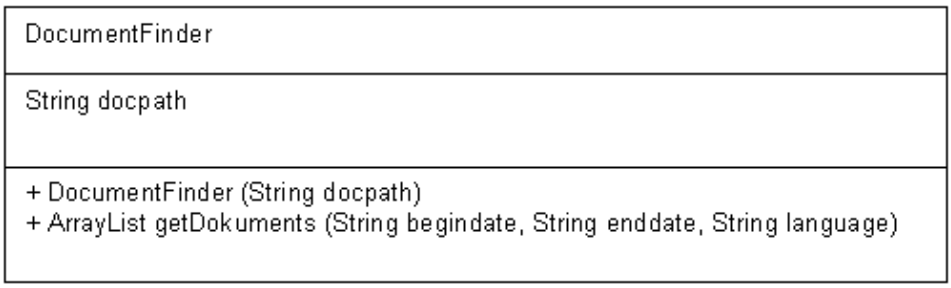

Abbildung 3.12.: Class ”DocumentFinder”

Klassifikator

Der Klassifikator versucht alle gefundene Dokumente in Cluster aufzuteilen um den Überblick zu erleichtern. Entsprechende Dokumententitel werden zu den Clusternamen und dementsprechend zu den Themennamen verarbeitet.

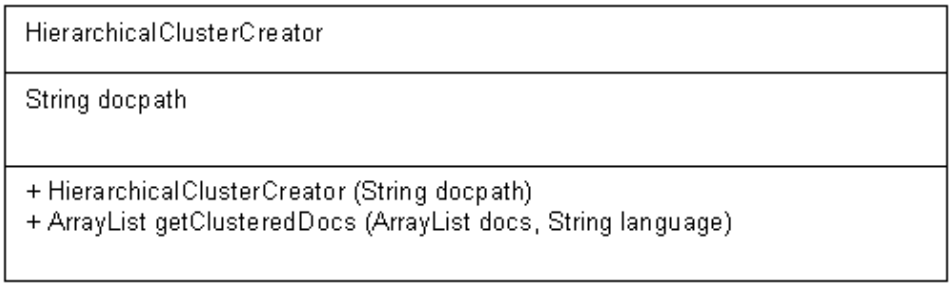

Abbildung 3.13.: Class ”HierarchicalClusterCreator”

Für das Clustering wird ein modifiziertes hierarchisches Clustering verwendet.

Zur Implementierung des hierarchischen Clusterings wurden folgende Annahmen vorgenommen:

- jedes Dokument wird als Anfangs-Cluster Betrachtet;
- erste Dendrogrammebene ist die Schnittebene;

- Abbruch-Bedingung ist nicht die Anzahl von Clustern, sondern das Überschreiten einer Ähnlichkeit (Distanz).

Die Ähnlichkeit von Dokumenten (oben erwähnte Distanz) wird mit Hilfe von Lucene Scoring-Mechanismen festgestellt.

Wegen der spezifischer Anfangsmenge (die Dokumente sind schon nach Gewichtung sortiert), ist dieser Verfahren etwas vereinfacht. Für jedes Top-Dokument aus der Liste wird einmal Distanz zu jedem unten stehendem Dokument berechnet. Überschreitet es eine gewisse Grenze, wird das verglichene Dokument zum Cluster des Top-Dokumentes hinzugefügt und aus der Anfangsliste entfernt. Anschließend bekommt der gebildete Cluster ein Name, kreiert aus dem Titel des Top-Dokuments. Ist die Liste durch, wird das Ganze mit dem nächststehenden Top-Dokument wiederholt. Falls kein Cluster für Top-Dokument gebildet werden kann, wird dieser zunächst am Ende der Liste platziert. Das Verfahren wiederholt sich solange wie sich noch weitere Cluster bilden und solange die Anfangsliste sich verändert. Alle zurückgeblibene Dokumente werden zu einem ”Rest”-Cluster zusammengefasst. Als Ergebnis wird ein `java.util.ArrayList` mit Clusternamen und Id's von zugehörigen Dokumenten zurückgegeben.

Als Abstandmaß zwischen Dokumenten, das über die Zugehörigkeit zum Cluster entscheiden lässt, wird ein sogenanntes ”score-limit”(S_i) ausgerechnet.

$$S_i = ln(\frac{s_{i-1}}{s_i})$$

Hierbei ist s_i ein Lucene-Scoring für das aktuelle Dokument und s_{i-1} ist das Lucene-Scoring für das zuletzt geclustertes Dokument. Der natürlicher Logarithmus ist hier zwecks Normierung eingeführt worden.

Abschließend stellt ein Klassendiagramm die Zusammensetzung der gesamten Anwendung und die logische Gliederung seiner Komponenten dar - Abbildung 3.14.

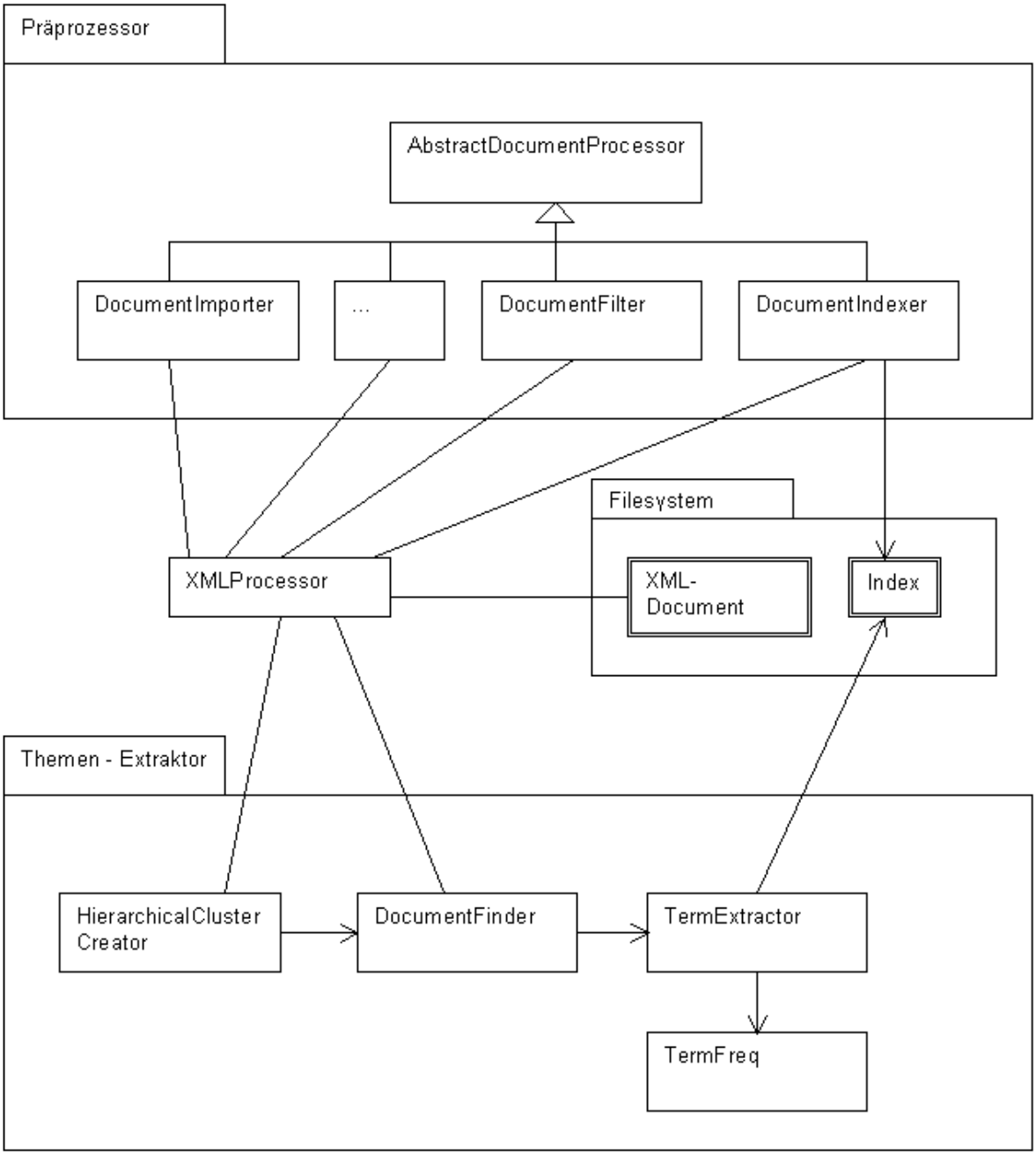

Abbildung 3.14.: Klassenkommunikation

4. Evaluierung

4.1. Allgemeines

Für die Evaluierung des Systems standen ca. 26000 Nachrichtentexte in drei Sprachen - Englisch, Deutsch und Französisch - bereit. Der überwiegende Teil davon ist in Englisch und Deutsch (12500 und 9250) verfasst. Daher werden Terme und Texte auf französich nachfolgend außer acht gelassen.

Als Testsystem fundierte ein handelsüblicher Arbeitsrechner mit dem ”Intel Pentium 4 (3000 MHz)” Prozessor, SATA-Festplatte (80 GB) und physikalischem Speicher von 1024 MB. Alle Leistungsangaben sind nachfolgend darauf zu beziehen.

Kriterien zur Evaluierung sind letzendlich auf Nützlichkeit der Ergebnisse für Endnutzer aufzuführen, daher ist die Nutzer-Evaluierung zur Beurteilung der Effizienz von Termen-Extraktors und Gesamtsystems als entscheidend bezeichnet worden. Außerdem sind auch zwei übliche Maße zur Beschreibung der Güte eines Suchergebnisses beim Information-Retrieval ”Recall”(Vollständigkeit) und ”Precision”(Genauigkeit) in Betracht genommen worden (siehe [Rijsbergen (1979)]).

4.2. Termen-Extraktor und Filter

Die Optimierung von Listen der häufigsten Terme erfolgt durch Einsatz und Justierung verschiedener Filter für das Feld ”METADATA” der XML-Dokumente. Das heisst, dass nach jeder Filteranpassung die gesamte Daten komplett neu gefiltert und indexiert werden mussten.

Die ersten Ergebnisse wurden mit standarten Stopwörtern aus dem ”Lucene-Analizers” Bibliothek erreicht, ohne Einsatz von Filter (siehe Tabelle 4.1) und über die ganze Dokumentenmange ohne Zeitbeschränkung.

In der nachfolgender Tabelle sind die 25 am häufigsten auftretender englischen Terme dargestellt (gekürzt, längere Liste aus 100 Termen - im Anhang A.1). Die ersten 3 Spalten zeigen die statistischen Daten zum jeweiligen Term. Die nächsten 2 - die Nutzerbeurteilung dazu.

Tabelle 4.1.: Termen-Liste ”Top 25 Englisch”

Platz	Term	Frequenz	Stopwort	Sprache
1	de	35301		fr
2	said	30392	ja	
3	airbus	19990		
4	its	17689	ja	
5	la	16546		fr
6	le	13758		fr
7	company	13740		
8	has	13354	ja	
9	à	12101		fr
10	2007	11747	ja	
11	boeing	11726		
12	eads	11102		
13	des	10424		de
14	equity	10345	ja	
15	et	10295		fr
16	afx	10241	ja	
17	news	10110	ja	
18	les	10018		fr
19	france	9805		
20	year	9777	ja	
21	all	9614	ja	
22	billion	9311	ja	
23	new	9217	ja	
24	jones	9172	ja	
25	million	9043	ja	

Tabelle 4.1.: Termen-Liste ”Top 25 Englisch”

In der nachfolgeder Tabelle sind die 25 am häufigsten auftretender deutschen Terme dargestellt (gekürzt, längere Liste aus 100 Termen - im Anhang A.2). Die ersten 3 Spalten zeigen die statistischen Daten zum jeweiligen Term. Die letzte - die Nutzerbeurteilung dazu.

Tabelle 4.2.: Termen-Liste ”Top 25 Deutsch”

Platz	Term	Frequenz	Stopwort
1	airbus	14931	
2	dpa	8912	ja
Fortsetzung auf nächster Seite			

Platz	Term	Frequenz	Stopwort
3	auflage	8060	ja
4	um	6486	ja
5	eads	6040	
6	zl	5955	ja
7	hat	5789	ja
8	sagte	5354	ja
9	euro	5309	
10	galileo	5242	
11	eu	4531	
12	ca	4138	ja
13	haben	3808	ja
14	mehr	3678	
15	seite	3444	
16	sei	3433	ja
17	2007	3388	ja
18	will	3305	
19	milliarden	3130	
20	noch	3054	ja
21	hamburg	3037	
22	07	2919	ja
23	erscheinungsdatum	2879	ja
24	verbr	2879	ja
25	verk	2879	ja

Tabelle 4.2.: Termen-Liste ”Top 25 Deutsch”

Man sieht, dass nur wenige von diesen Wörtern einen Wert für Nutzer haben - manche gehören ausgefiltert (was zum normalen Optimierungsprozess gehört), die Anderen (siehe Tabelle 4.1) befinden sich überhaupt fehlerhaft in der Liste und kommen aus anderen Sprachen. Die Analyse von der Texten ergab, dass verfasste Nachrichtentexte nicht immer die richtige Sprache-Bezeichnung bekommen. Wie sich später herausgestellt hat, hatten schätzugsweise 18% angeblich englischsprachige Dokumente falsche Markierung. Ist das ein menschlicher Faktor (vom Herausgeber nicht angedeutet und als Default-Wert englisch genommen) oder Parser-Fehler, ist nicht klar und auch nicht so wichtig. Die Schlussfolgerung ist gleich - man benötigt einen Korrektor (siehe Unterkapitel 4.3).

Betrachtet man die Terme, die als Stopwörter bezeichnet waren, stellt man fest, dass sie sich in zwei Gruppen unterteilen. Ein Teil schließt typische Stopwörter, die bei Indexierung überlistet werden (z.B. ”after”,”has”,”um”,”nur”...), der andere entspricht spezifischen Abkürzungen, Phrasen oder Wortgruppen, die für Texte der Nachrichtenagenturen typisch sind

(wie ”afx” von ”Afx News Limited”, ”jones” von ”Dow Jones”,”dpa” oder auch ”verbr” und ”erscheinungsdatum”). Diese erkannte Phrasen muss man in einen `CustomPhraseFilter` eintragen um sie von den ähnlichen Termen, die aber in Meldungen-Kontext erscheinen zu trennen.

Bei weiteren Durchläufen wurden die Filterungs- und Stopwortlisten korrigiert und erweitert. Als nächster Schritt wurden sämtliche Email-Adressen im `EMailFilter` eliminiert, Datum-/Zeitangaben (Beispiel - ”09.06.2007”) und alleinstehende Zahlen (wie ”2006”, aber nicht ”A380”) im `NumberFilter`. Anschließend wurden spezielle Zeichen (wie z.B. ”CARRIAGE RETURN”, ”&”, ”?”...) ausgefiltert.

Die endgültige Filterreihenfolge wurden in einem Konfigurations-XML wie folgt eingetragen:

```
<?xml version="1.0" encoding="UTF-8"?>
<WORKFLOW>
    <FILTER>
        com.uknow.filters.CustomPhraseFilter
    </FILTER>
    <FILTER>
        com.uknow.filters.EMailFilter
    </FILTER>
    <FILTER>
        com.uknow.filters.NumberFilter
    </FILTER>
    <FILTER>
        com.uknow.filters.SpecialSignFilter
    </FILTER>
    <FILTER>
        com.uknow.filters.MultipleWhiteSpaceFilter
    </FILTER>
</WORKFLOW>
```

Anschließend sehen die Termlisten viel plausibler (auch wenn nicht perfekt) aus, Der Term-Extraktor liefert schon brauchbare Ergebnisse, die für die weiteren Zwecke dieser Arbeit ausreichen.

Platz	Term	Frequenz
1	airbus	23463
2	company	18831
3	boeing	16862
Fortsetzung auf nächster Seite		

Platz	Term	Frequenz
4	year	14007
5	billion	13014
6	percent	12904
7	aircraft	12576
8	million	12493
9	new	12439
10	eads	11669
11	air	11004
12	cn	10804
13	more	10054
14	news	8287
15	shares	7877
16	first	7664
17	after	7389
18	contract	7378
19	euros	7089
20	based	6974
21	story	6711
22	group	6657
23	order	6608
24	european	6397
25	press	6377

Tabelle 4.3.: Endgültige Termen-Liste ”Top 25 Englisch”

Platz	Term	Frequenz
1	airbus	22913
2	eads	10578
3	euro	7756
4	prozent	5652
5	mehr	5471
6	hamburg	5435
7	eu	5392
8	galileo	5389
9	will	4853
10	deutschland	4811
11	seite	4482
12	war	4436
13	milliarden	4193
Fortsetzung auf nächster Seite		

Platz	Term	Frequenz
14	unternehmen	4011
15	deutschen	3814
16	boeing	3618
17	de	3508
18	us	3489
19	jahr	3468
20	keine	3433
21	berlin	3421
22	beim	3332
23	damit	3285
24	gegen	3232
25	a380	3223

Tabelle 4.4.: Endgültige Termen-Liste ”Top 25 Deutsch”

Wird die Zeitspanne auf ein oder zwei Tage begrenzt, gibt der Term-Extraktor andere Listen zurück, die in erster Linie die Terme enthalten, die an diesen konkreten Tagen vorwiegend erwähnt werden.
Log-Ausschnitt mit einem Beispiel:

```
Searched from 29.05.2007 17:00:00 to 31.05.2007 17:00:00
Number of terms for this time = 41942
Top 25:
1:Term=airbus; Frequency=76
2:Term=a350; Frequency=36
3:Term=eads; Frequency=23
4:Term=airways; Frequency=21
5:Term=qatar; Frequency=21
6:Term=a350xwb; Frequency=20
7:Term=maschinen; Frequency=18
8:Term=auftrag; Frequency=17
9:Term=avianca; Frequency=16
10:Term=chef; Frequency=15
11:Term=forgeard; Frequency=15
12:Term=boeing; Frequency=14
13:Term=kunden; Frequency=14
14:Term=mittwoch; Frequency=14
15:Term=flugzeuge; Frequency=13
16:Term=will; Frequency=12
```

```
17:Term=a380; Frequency=11
18:Term=erste; Frequency=11
19:Term=fluggesellschaft; Frequency=11
20:Term=a330; Frequency=10
21:Term=dollar; Frequency=10
22:Term=northrop; Frequency=10
23:Term=wurde; Frequency=10
24:Term=bestellt; Frequency=9
25:Term=jahr; Frequency=9
searchtext = 'DATE:[20070529170000 TO 20070531170000]
airbus a350 eads airways qatar a350xwb maschinen auftrag
avianca chef forgeard boeing kunden mittwoch flugzeuge
will a380 erste fluggesellschaft a330 dollar northrop
wurde bestellt jahr'
```

Liste der Dokumente, die nach Anfrage mit diesen Termen von Suchmaschine erstellt wird:

- (Zusammenfassung 1600) Riesenauftrag für Airbus: Qatar Airways bestellt 80 A350XWB
- Qatar Airways bestellt 80 A350XWB bei Airbus
- UPDATE: Qatar Airways bestellt 80 A350XWB bei Airbus
- Qatar Airways bestellt 80 Airbus A350 (zwei)
- Qatar Airways bestellt 80 Airbus A350 - Bisher größter Auftrag
- Airbus: Avianca bestellt 70 Flugzeuge
- (Riesenauftrag für Airbus - Zusammenfassung 1600 - Paris) Avianca kauft 38 Airbusse - Dazu 32 Optionen
- Avianca erwägt Bestellung von über 40 Airbus-Maschinen
- Avianca erwägt Bestellung von über 40 Airbus-Maschinen - Tribune
- AIRBUS - KOLUMBISCHE FLUGGESELLSCHAFT AVIANCA ORDERT 70 FLUG-ZEUG
- EADS: Forgeard attackiert deutsche Manager und Hamburger Airbus-Werk
- Ryanair bestellt 27 Boeing - Listenpreis ist 1,9 Mrd. Dollar
- Ex-EADS-Chef gibt Deutschen Schuld für seinen Weggang
- "L'Indépendant du Midi": Erste Gewissensentscheidung für Sarkozy

- Siemens erhält Auftrag von Northrop Grumman
- Siemens erhält Auftrag von Northrop Grumman über 161 Mio EUR
- Ryanair kauft 27 Boeing 737-800 im Wert von 1,9 Mrd USD
- Airbus will noch in diesem Jahr Partner für Werke finden
- Airbus erwartet 200 Bestellungen der A350 XBW 2007
- Airbus beginnt im Juni Gespräche über Stellenabbau für Deutschland.
- MARKT/Orders für ”A-350” von Airbus zeigen allmählich Momentum
- Großbritannien erwartet Rüstungs-Aufträge aus Libyen
- WDHLG-Großbritannien erwartet Rüstungs-Aufträge aus Libyen

Eine Beurteilung des Endbenutzes: diese oder ähnliche Dokumente wären auch als entscheidend für diese Zeitspanne von einem Redakteuren (Lektoren) manuell ausgewählt.

4.3. Sprachkorrektor

Sprachkorrektor wird als Bestandteil des Präprozessors eingesetzt und daher als Erbe der Klasse `AbstractDocumentProcessor` implementiert. Er versucht die Originalsprache des Dokumentes durch Vergleich von Termenliste mit Stopwordlisten zu erkennen. Die Grundidee kann man leicht mit einem Pseudocode beschreiben (Osinski u. a. (2004)):

```
für jedes Dokument {
  für jede Stopwort-Liste {
    Alle Übereinstimmungen von Dokumenten-Termen
    mit den aus der Stopwort-Liste zählen
  }
  Die Stopwort-Liste mit
  Wenn (höhste Zähler > 1) {
    Sprache ist die von höhsten Zähler
  } sonst {
    Sprache ist nicht erkannt - Defaultwert
  }
}
```

Da viele Texte heutzutage auch internationale (meistens englische) Terme benutzen, wurde zusätzlich ein Schwelle-Parameter eingefügt, der sich aus gezählten Treffern geteilt durch die gesamte Anzahl von Termen im Dokument ergibt. Wird die Schwelle von 20% nicht erreicht, bleibt die Sprache undefiniert.
Nach der Bearbeitung bestehender Daten wurden 2717 angeblich englischsprachige Dokumente mit hoher Wahrscheinlichkeit als französische bzw. deutsche erkannt. Als Resultat konntebeobachtet werden, dass keine Fremdwörter in der Liste der englischer Terme (siehe Tabelle 4.4) auftauchten.

4.4. Themen-Extraktor

Das im Unterkapitel 3.3.3 beschriebene Clustering-Verfahren hat bei verschiedenen Daten auf unterschiedliche Wegen funktioniert. Es waren jedoch zwei wesentliche Typen deutlich zu unterscheiden: mit normaler und flacher Verteilung von ”score-limit” bei der Dokumentensuche.

Clustering-Beispiel für normale Scoring-Verteilung (fett- Clusternamen mit zugehörigen Dokumenten-IDs und Dokumententitel):

Qatar Airways bestellt 80 A350XWB bei Airbus:
19387: Qatar Airways bestellt 80 A350XWB bei Airbus
19282: (Zusammenfassung 1600) Riesenauftrag für Airbus: Qatar Airways bestellt 80 A350XWB
19284: Qatar Airways bestellt 80 Airbus A350 (zwei)
19297: UPDATE: Qatar Airways bestellt 80 A350XWB bei Airbus
19269: Qatar Airways bestellt 80 Airbus A350 - Bisher größter Auftrag
Airbus: Avianca bestellt 70 Flugzeuge:
19407: Airbus: Avianca bestellt 70 Flugzeuge
19306: (Riesenauftrag für Airbus - Zusammenfassung 1600 - Paris) Avianca kauft 38 Airbusse - Dazu 32 Optionen
19280: Avianca erwägt Bestellung von über 40 Airbus-Maschinen
19259: Avianca erwägt Bestellung von über 40 Airbus-Maschinen - Tribune
19301: AIRBUS - KOLUMBISCHE FLUGGESELLSCHAFT AVIANCA ORDERT 70 FLUGZEUG
EADS: Forgeard attackiert deutsche Manager und Hamburger Airbus-Werk:
19277: EADS: Forgeard attackiert deutsche Manager und Hamburger Airbus-Werk
19378: Ex-EADS-Chef gibt Deutschen Schuld für seinen Weggang
Ryanair bestellt 27 Boeing - Listenpreis ist 1,9 Mrd. Dollar:
19418: Ryanair bestellt 27 Boeing - Listenpreis ist 1,9 Mrd. Dollar
19386: Ryanair kauft 27 Boeing 737-800 im Wert von 1,9 Mrd USD
”L’Indépendant du Midi”: Erste Gewissensentscheidung für Sarkozy
19369: ”L’Indépendant du Midi”: Erste Gewissensentscheidung für Sarkozy

19319: Airbus erwartet 200 Bestellungen der A350 XBW 2007
19357: MARKT/Orders für ”A-350” von Airbus zeigen allmählich Momentum
19409: Airbus will noch in diesem Jahr Partner für Werke finden
Siemens erhält Auftrag von Northrop Grumman über 161 Mio EUR:
19295: Siemens erhält Auftrag von Northrop Grumman über 161 Mio EUR
19296: Siemens erhält Auftrag von Northrop Grumman
Großbritannien erwartet Rüstungs-Aufträge aus Libyen:
19244: Großbritannien erwartet Rüstungs-Aufträge aus Libyen
19247: WDHLG-Großbritannien erwartet Rüstungs-Aufträge aus Libyen
...:
19417: Airbus beginnt im Juni Gespräche über Stellenabbau für Deutschland.

Die in Abbildung 4.1 dargestellte Grafik zeigt eine typische Scoring-Verteilung nach einer Dokumentensuche bei der Clusterbildung. Die Kurve des Lucene-Score fängt ziemlich hoch (bei 0.86 mit maximal 1.0) an. Die Dokumente 1 bis 4 liegen nah bei einander (die Score-Limits dementsprechend sind nah zum ”0”) - sie gehören klar zu einem Cluster. Danach steigt die Kurve schnell ab und je nach Schwellengrösse werden weitere Dokumente aus dem Cluster ausgeschlossen.

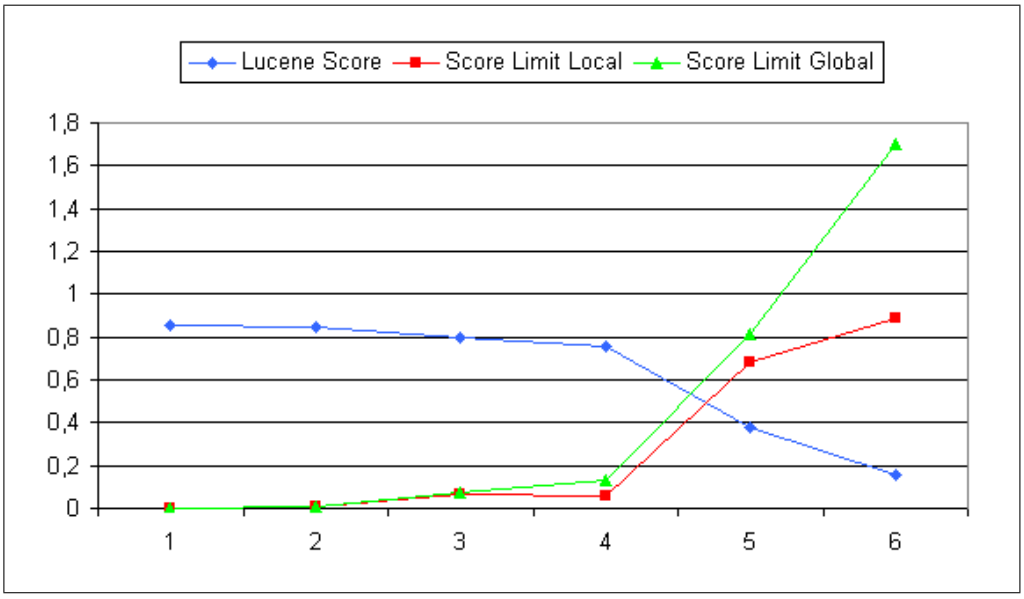

Abbildung 4.1.: Beispieldaten für ”Normale” Scoring-Verteilung

Clustering-Beispiel für eine flache Scoring-Verteilung (hier für den direkten Vergleich - englischsprachige Meldungen aus der gleicher Zeitspanne wie im Beispiel 4.4):

Airbus Wins Qatar Commitment for 80 A350s Worth $16 Billion :

19274: Airbus Wins Qatar Commitment for 80 A350s Worth $16 Billion
19272: Airbus Wins $16 Billion Pledge From Qatar for A350s (Update1)
19294: Airbus Wins $16 Billion Pledge From Qatar for A350s (Update4)
19270: Airbus Wins $16 Billion Pledge From Qatar for A350s (Update2)
19317: Airbus Wins $16 Billion Pledge From Qatar for A350s (Update6)
19285: Qatar Airways signs new contract for more A350 jets
19287: Qatar Airways signs new contract for more A350 jets
19328: 2ND UPDATE: Airbus Wins $16 Bln A350 WXB Order From Qatar Airways
19325: Qatar Airways signs new contract for more A350 jets
19370: Air Astana May Place Plane Order by End of This Year (Correct)
19318: Air Astana May Order Planes This Year, Double Fleet by 2015
19275: Finmeccanica Expects Orders to Grow 8 Percent by 2008 (Update2)
19273: Finmeccanica Expects Orders to Grow 8 Percent by 2008 (Update1)
19371: French Stocks Advance, Led by EADS, EDF, Vallourec Shares
19393: Ryanair Orders 27 Boeing Jets Valued at $1.9 Billion (Update2)
19305: Finmeccanica Expects Orders to Grow 8 Percent by 2008 (Update3)
19389: Germany's DAX Gains, Led by E.ON, Infineon, Fresenius Medical
19390: Ryanair Orders 27 Boeing Jets Valued at $1.9 Billion (Update3)
19333: Business Highlights
19355: Air France, Alcatel, Danone and EADS: French Equity Preview
...:
19337: Zoellick Must Restore Calm at World Bank
19334: Zoellick faces new challenge: Restoring confidence at World Bank
19336: Bush chooses former trade chief Zoellick as nominee for World Bank president
19338: AP Interviews: Zoellick's trade experience prepared him for World Bank job, former rivals say
19339: EU FIN Zoellick World Reax

Es ist sehr deutlich erkennbar, dass das Clustering mit dem ausgewählten Abstands-Parametern hier keine nutzbare Ergebnisse liefert. Zum größten Teil sind die Dokumente in ein einziges Cluster gruppiert, obwohl die Themen sich stark von einander unterscheiden. Außerdem, man kann feststellen, dass nicht klassifizierte Dokumente (in letztem Abschnitt) gehören klar zum gleichen Thema, das hier nicht erkannt wurde.

Grund dieses Verhaltens ist eine flache Scoring-Verteilung bei der Dokumentensuche (siehe Abbildung 4.2). Der Lucene-Score fängt sehr tief an (bei 0.54) und steigt langsam ab. Auch bei dem Score-Limit gibt es keine sichbare Ausreißer bis auf Dokument 21, was schon sehr spät ist - alle bevorstehende Dokumente sind zu einem Cluster gruppiert worden.

Die anfängliche Abbruch-Bedingung nach localen "Score-Limit" (S_i aus dem Unterkapitel 3.3.3) musste mit einer weiteren Abbruch-Bedingung nach globalen "Score-Limit" (S_j)

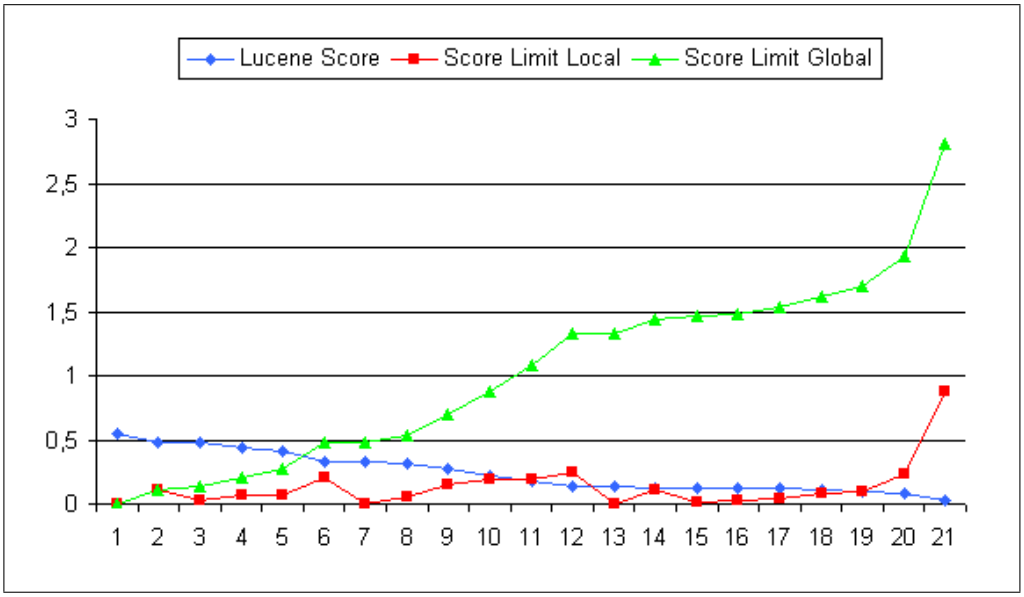

Abbildung 4.2.: Beispieldaten für ”Flache” Scoring-Verteilung

ergänzt werden:

$$S_j = ln(\frac{s_i}{s_{max}})$$

Hierbei ist s_i ein Lucene-Scoring für aktuelles Dokument und s_{max} ist das Lucene-Scoring für das aktuelle Top-Dokument.

Werden beide ”Score-Limits” auf 0.7 gesetzt (einstellbar), ergibt sich ein wesentlich besseres Clustering-Ergebnis:

Airbus Wins Qatar Commitment for 80 A350s Worth $16 Billion :
19274: Airbus Wins Qatar Commitment for 80 A350s Worth $16 Billion
19272: Airbus Wins $16 Billion Pledge From Qatar for A350s (Update1)
19294: Airbus Wins $16 Billion Pledge From Qatar for A350s (Update4)
19270: Airbus Wins $16 Billion Pledge From Qatar for A350s (Update2)
19317: Airbus Wins $16 Billion Pledge From Qatar for A350s (Update6)
19285: Qatar Airways signs new contract for more A350 jets
19287: Qatar Airways signs new contract for more A350 jets
19328: 2ND UPDATE: Airbus Wins $16 Bln A350 WXB Order From Qatar Airways
19325: Qatar Airways signs new contract for more A350 jets
Air Astana May Place Plane Order by End of This Year :
19370: Air Astana May Place Plane Order by End of This Year (Correct)
19318: Air Astana May Order Planes This Year, Double Fleet by 2015
Zoellick Must Restore Calm at World Bank:

19337: Zoellick Must Restore Calm at World Bank
19334: Zoellick faces new challenge: Restoring confidence at World Bank
19336: Bush chooses former trade chief Zoellick as nominee for World Bank president
19338: AP Interviews: Zoellick's trade experience prepared him for World Bank job, former rivals say
19339: EU FIN Zoellick World Reax

Air France, Alcatel, Danone and EADS: French Equity Preview :

19355: Air France, Alcatel, Danone and EADS: French Equity Preview
19371: French Stocks Advance, Led by EADS, EDF, Vallourec Shares

Finmeccanica Expects Orders to Grow 8 Percent by 2008 :

19275: Finmeccanica Expects Orders to Grow 8 Percent by 2008 (Update2)
19273: Finmeccanica Expects Orders to Grow 8 Percent by 2008 (Update1)
19305: Finmeccanica Expects Orders to Grow 8 Percent by 2008 (Update3)

Ryanair Orders 27 Boeing Jets Valued at $1.9 Billion :

19390: Ryanair Orders 27 Boeing Jets Valued at $1.9 Billion (Update3)
19393: Ryanair Orders 27 Boeing Jets Valued at $1.9 Billion (Update2)

...:

19389: Germany's DAX Gains, Led by E.ON, Infineon, Fresenius Medical
19333: Business Highlights

Die Ausführungszeiten des Themen-Extraktors spielen geradezu entscheidende Rolle bei der Einschätzung der Anwendungsgeschwindigkeit. Wichtig dabei sind die Datenmengen an Nachrichten, die indexiert werden. Beispielsweise für die vorliegende Testdaten (beschrieben in 4.1) betrugen die Ausführungszeiten ca. 40-60 Sekunden.

5. Resümee

5.1. Zusammenfassung

Ein Hauptziel des vorliegendes Projekts ist eine Prototyp-Anwendung zu entwickeln, die den menschlichen Lektoren monotone Aufgaben der Suche und Erstellung von aktuellen Tagesthemen aus unstrukturierten Nachrichtenmengen abnimmt.

Im Rahmen dieser Arbeit wurden notwendige Komponenten eines solchen Systems identifiziert, eine modulare Architektur entwickelt und in Form von Java-Klassen implementiert. Weiterhin wurde erstellte Anwendung mit vorhandenen Testdaten konfrontiert und nach der Evaluierung optimiert.

Die Anwendung ”TagesThemen” ist für die Verwendung innerhalb eines Systems für Nachrichtenerfassung (wie ”uknow NewsManager”, beschrieben im Kapitel 2.1) konzipiert und orientiert sich auf große bis mittelgroße Unternehmen mit hohen Medienpräsenz und eigener Pressestelle.

Dabei wird die monotones Teil der Arbeit - Aufsuchen und Zusammenfassung von ähnlichen Texten - automatisch zur bestimmter Uhrzeit oder nach Aufforderung von Software erledigt. Den Redakteuren bleibt mehr Zeit für kreative Aufgaben, was die Kosten senkt und gleichzeitig die Qualität bei der Informierung von Kunden wesentlich steigert.

5.2. Ausblicke

Es wurden zwar alle Ziele der Diplomarbeit erreicht, aber nach Evaluierung der Testdaten, ergeben sich jedoch weitere wichtige Aspekte.

Die Struktur des Präprozessors ist keineswegs fixiert. Sie sollte bei Bedarf angepasst und/oder erweitert werden. Ein Beispiel dazu - das Einfügen des Sprachkorrektors im Projekt, das sich nur nach der erster Evaluierungsphase als nötig erwiesen hat. Eingesetzte Textfilter und Stopwortlisten sind weitaus beispielhaft und müssen für jeden spezifischen Kunden angepasst oder neu entwickelt werden. Beispiel: Wörter mit Stammwörtern und/oder Synonymen ersetzen.

Geschwindigkeit des Term-Extraktors (und somit der Anwendung insgesamt) ist stark von der Größe des Indexes abhängig (genaue gesagt - von der Anzahl der indexierten Texten). Mit vorausgesehenen 1000 Dokumenten pro Tag, kommt man auf ungefähr 30000 pro Monat. Eine Themen-Extraktion für diese Zeitspanne wird schätzungsweise 3 Minuten dauern. Für die größere Datenmengen wird die Durchlaufszeit linear aufsteigen. Daher wäre es ratsam, den verwendete Index so klein wie möglich zu halten. Dies kann man erreichen indem man veraltete Dokumente, die nicht mehr erfasst werden, in einem getrenntem Prozess entfernen lässt.

Das hierarchische Clustering-Verfahren im Projekt liefert erste brauchbare Ergebnisse. Die Qualität der Ergebnisse bei der Zuordnung von Dokumenten zu einem Thema ändert sich noch sehr stark in Bezug auf Textinhalte und Parametereinstellung. Desweiteren ist die Erzeugung von Themennamen sehr einfach gestaltet und wird im wesentlichen noch dem Nutzer überlassen. Der Clustering-Mechanismus sollte daher weiter optimiert und entwickelt werden. Eine solche Entwicklung ist aber ein Thema für sich und würde den Rahmen dieser Abschlussarbeit sprengen.

Der erste Schritt zum Aufbau eines effektives Systems für die automatische Themenerzeugung wurde getan. In der vorliegender Arbeit wurden sowohl Grundkonzepte definiert als auch eine Prototyp-Implementierung mit Open-Source Biblioteken durchgeführt. Die Weiterentwicklung des Systems für die kommerzielle Nutzung sieht in diesem Kontext vielversprechend aus.

Literaturverzeichnis

[Cunningham 1997] CUNNINGHAM, H.: *Information extraction - a user guide*. 1997. – URL citeseer.ist.psu.edu/cunningham99information.html

[Cunningham 2006] CUNNINGHAM, Hamish: Information Extraction, Automatic. Preprint, 18th November 2004, at http://gate.ac.uk/sale/ell2/ie/main.pdf. In: *Encyclopedia of Language and Linguistics, 2nd Edition, Elsevier* 5 (2006), November, S. 665–677. – URL http://gate.ac.uk/sale/ell2/ie/main.pdf

[Doucet und Ahonen-Myka 2002] DOUCET, Antoine ; AHONEN-MYKA, Helena: Naive clustering of a large XML document collection. In: *Proceedings of the First Workshop of the Initiative for the Evaluation of XML Retrieval (INEX)*. Schloss Dagsuhl, Germany, 2002, S. 81–87

[Hatcher und Gospodnetic 2004] HATCHER, Erik ; GOSPODNETIC, Otis: *Lucene in Action*. Manning Publications, December 2004. – URL http://www.manning.com/hatcher2/. – ISBN 1932394281

[Osinski 2006] OSINSKI, Stanislaw: Improving Quality of Search Results Clustering with Approximate Matrix Factorisations. In: *ECIR*, 2006, S. 167–178

[Osinski u. a. 2004] OSINSKI, Stanislaw ; STEFANOWSKI, Jerzy ; WEISS, Dawid: Lingo: Search Results Clustering Algorithm Based on Singular Value Decomposition. In: *Intelligent Information Systems*, 2004, S. 359–368

[Rijsbergen 1979] RIJSBERGEN, C. J. v.: *Information retrieval*. 2. London : Butterworths, 1979. – URL http://www.dcs.glasgow.ac.uk/Keith/Preface.html

[uknow 2007] UKNOW: *News Manager*. 2007. – URL http://www.uknow.com

[Zhang u. a. 2005] ZHANG, Yongzheng ; ZINCIR-HEYWOOD, Nur ; MILIOS, Evangelos: Narrative text classification for automatic key phrase extraction in web document corpora. In: *WIDM '05: Proceedings of the 7th annual ACM international workshop on Web information and data management*. New York, NY, USA : ACM, 2005, S. 51–58. – ISBN 1-59593-194-5

A. Anhang

A.1. Termen-Liste Englisch

Tabelle A.1.: Termen-Liste ”Top 100 Englisch”

Platz	Term	Frequenz	Stopwort	Sprache
1	de	35301		fr
2	said	30392	ja	
3	airbus	19990		
4	its	17689	ja	
5	la	16546		fr
6	le	13758		fr
7	company	13740		
8	has	13354	ja	
9	à	12101		fr
10	2007	11747	ja	
11	boeing	11726		
12	eads	11102		
13	des	10424		de
14	equity	10345	ja	
15	et	10295		fr
16	afx	10241	ja	
17	news	10110	ja	
18	les	10018		fr
19	france	9805		
20	year	9777	ja	
21	all	9614	ja	
22	billion	9311	ja	
23	new	9217	ja	
24	jones	9172	ja	
25	million	9043	ja	
26	percent	8719	ja	

Fortsetzung auf nächster Seite

Platz	Term	Frequenz	Stopwort	Sprache
27	en	8717		fr
28	aircraft	8260		
29	2006	8236	ja	
30	c	7828	ja	
31	rights	7318	ja	
32	du	7174		fr
33	more	7015		
34	co	6658	ja	
35	der	6561		de
36	dow	6524	ja	
37	air	6501		
38	die	6260		de
39	press	5754	ja	
40	international	5509		
41	after	5366	ja	
42	euros	5361		
43	un	5275		fr
44	pour	5261		fr
45	first	5239		
46	contract	5208		
47	group	5151		
48	shares	5084		
49	based	5011		
50	a380	4987		
51	european	4841		
52	reuters	4722	ja	
53	dans	4701		fr
54	presse	4563		de
55	over	4499		
56	defense	4430		
57	last	4424		
58	sur	4254		fr
59	une	4242		fr
60	fp	4119	ja	
61	order	4055		
62	inc	4014	ja	
63	government	4011		
64	que	3967		fr

Fortsetzung auf nächster Seite

Platz	Term	Frequenz	Stopwort	Sprache
65	afp	3930	ja	
66	planes	3909		
67	french	3860		
68	au	3843		fr
69	story	3831		
70	qui	3731		fr
71	par	3701		fr
72	plan	3693		
73	sales	3691		
74	profit	3684		
75	force	3641		
76	financial	3551		
77	systems	3530		
78	s	3514	ja	
79	und	3512		de
80	one	3499		
81	limited	3488	ja	
82	years	3484	ja	
83	contact	3459		
84	10	3415	ja	
85	including	3408		
86	stock	3370		
87	see	3319	ja	
88	buy	3238		
89	chief	3220		
90	orders	3217		
91	sa	3202		fr
92	share	3202		
93	airline	3197		
94	corp	3191	ja	
95	plane	3171		
96	top	3153		
97	plans	3152		
98	editor	3129		
99	business	3094		
100	deal	3085		

Tabelle A.1.: Termen-Liste ”Top 100 Englisch”

A.2. Termen-Liste Deutsch

Tabelle A.2.: Termen-Liste ”Top 100 Deutsch”

Platz	Term	Frequenz	Stopwort
1	airbus	14931	
2	dpa	8912	ja
3	auflage	8060	ja
4	um	6486	ja
5	eads	6040	
6	zl	5955	ja
7	hat	5789	ja
8	sagte	5354	ja
9	euro	5309	
10	galileo	5242	
11	eu	4531	
12	ca	4138	ja
13	haben	3808	ja
14	mehr	3678	
15	seite	3444	
16	sei	3433	ja
17	2007	3388	ja
18	will	3305	
19	milliarden	3130	
20	noch	3054	ja
21	hamburg	3037	
22	07	2919	ja
23	erscheinungsdatum	2879	ja
24	verbr	2879	ja
25	verk	2879	ja
26	so	2872	ja
27	prozent	2862	
28	reichweite	2852	
29	unternehmen	2785	
30	deutschland	2707	
31	meldung	2499	ja
32	us	2414	
33	zusammenfassung	2377	ja
34	war	2304	ja
35	gedr	2298	ja

Fortsetzung auf nächster Seite

Platz	Term	Frequenz	Stopwort
36	man	2271	ja
37	jahr	2138	
38	nur	2125	ja
39	09.06.2007	2094	ja
40	chef	2057	
41	deutschen	2039	
42	gegen	2036	
43	habe	2033	ja
44	boeing	1987	
45	keine	1987	
46	seit	1967	ja
47	europa	1932	
48	berlin	1903	
49	ge	1894	ja
50	rund	1879	ja
51	30	1851	ja
52	deutsche	1822	
53	flugzeugbau	1819	
54	damit	1799	
55	10	1792	ja
56	jones	1768	ja
57	2006	1740	ja
58	pk	1721	ja
59	beim	1698	
60	000	1692	ja
61	all	1665	ja
62	heute	1630	ja
63	neue	1567	
64	bereits	1563	
65	freitag	1537	
66	zwei	1536	
67	europäische	1455	
68	gallois	1443	
69	usa	1439	
70	uhr	1426	ja
71	seien	1425	ja
72	wegen	1407	
73	a	1403	ja

Fortsetzung auf nächster Seite

Platz	Term	Frequenz	Stopwort
74	stellen	1393	
75	20	1391	ja
76	drei	1380	
77	ab	1374	ja
78	neuen	1347	
79	bandar	1331	
80	konzern	1326	
81	muss	1325	ja
82	be	1324	ja
83	8	1319	ja
84	feb	1308	ja
85	ten	1284	ja
86	a380	1280	
87	frankreich	1270	
88	etwa	1261	ja
89	gibt	1251	ja
90	bremen	1250	
91	werke	1250	
92	dann	1240	ja
93	ende	1240	
94	n	1239	ja
95	power8	1226	
96	spd	1222	
97	15	1217	ja
98	wollen	1211	ja
99	dow	1210	ja
100	millionen	1203	

Tabelle A.2.: Termen-Liste ”Top 100 Deutsch”

Versicherung über Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit im Sinne der Prüfungsordnung nach §24(5) ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, 4. Dezember 2007	
Ort, Datum	Unterschrift