Web Log Mining

Seminar WWW und Datenbanken SS 2001
Lehrstuhl für Datenbanken und Informationssysteme
Johann-Wolfgang-Goethe Universität Frankfurt – Fachbereich Biologie und Informatik
Betreuer: Dr. Christoph Schommer

Martin Klossek – martin@klossek3000.de

Inhaltsverzeichnis

1 Einleitende Zusammenfassung

Im World Wide Web werden täglich unzählbar viele Dokumente von Webservern an Arbeitsrechner ausgeliefert. Enthalten die Dokumente Nachrichten, Börseninformationen, Unterhaltung oder auch Online-Shopping-Angebote, ihnen gemein ist, dass sie auf einem technischen Standard, der Seitenbeschreibungssprache HTML, aufbauen. Zusätzlich werden Grafiken, Java- und Flash-Applets mit den Dokumenten versendet.

Jedes an einen Arbeitsrechner ausgelieferte Dokument ergänzt das Profil des davor sitzenden Anwenders, der in einer Sitzung, an einem Tag, in einer Woche oder auch in seinem ganzen Leben, verschiedene Interessen verfolgt und dementsprechend verschiedene Websites und dort unterschiedliche Bereiche ansteuert. Für Websitebetreiber ist es nicht nur interessant sondern aus wirtschaftlichen Gesichtspunkten sogar notwendig, das Verhalten ihrer Besucher und deren Aktivitäten auf der eigenen Website zu untersuchen.

> *Warum ist das nötig und mit welchen Mitteln können solche Untersuchungen vorgenommen werden?*

Die folgende Arbeit versucht, diesen beiden recht prinzipiellen Fragen näherzukommen, eine entsprechende Motivation zu liefern und sinnvolle Antworten zu finden. Da primär in den Web Logdaten geschürft wird, lautet das Thema Web Log Mining. Genaue Begriffserklärungen werden an gegebener Stelle erfolgen.

Gesamtziel der Ausarbeitung ist letztlich, einerseits einen Überblick über theoretische Grundlagen von Web Log Mining Verfahren und des Komplexes Webserver zu geben, andererseits aber auch praktische Herangehensweisen mit konkreten Fragestellungen zu erörtern und falls möglich entsprechende Antworten zu geben.

1.1 Übersicht

Der soeben gegebenen Einleitung schließt sich ein motivierendes Kapitel an, in dem die Hintergründe und Notwendigkeiten von Web Log Mining beschrieben werden. Anschließend folgt der Themenkomplex Webserver mit Spezifikationen zu den sogenannten Logdateien, auf die dort ausführlicher eingegangen wird. Zudem ist dort eine knappe Abgrenzung von Data Mining zu anderen gebräuchlichen Analyseverfahren zu finden.

Sind die ersten beiden Kapitel zur Motivation und Begriffsklärung vorgesehen, geht es im Kapitel „Prozess Web Log Mining“ um die eigentliche Vorgehensweise zum Finden von Informationen aus Logdateien. An dieser Stelle fließen sowohl konkrete Fragestellungen ein, die von Websitebetreibern gestellt werden, als auch praktische Erfahrungen, die mit der Auswertung von Logdateien gesammelt werden konnten. Abgerundet wird der Abschnitt durch Interpretationshinweise und entsprechende Ideen zu den durch Web Log Mining gefundenen Ergebnissen.

Bleibt noch der Ausblick zu nennen, in dem versucht wird, potentielle Entwicklungs-möglichkeiten des Web Log Mining zu geben. Zudem findet sich dort auch eine persönliche Bewertung dieser Technik.

1.2 Anmerkungen

Die Ausarbeitung wurde mit Microsoft Word 2000 zwischen Mai und Juni 2001 verfaßt. Sie steht im Internet unter http://www.stormzone.de/uni/Hauptstudium/seminare/wwwdb/list.php3 zur Verfügung. Kontakt zum Autor gerne über mailto:martin@klossek3000.de.

2 Motivation

In diesem ersten Kapitel soll dem Leser nahegelegt werden, warum Data Mining in Web Logs praktiziert werden sollte. Da primär Motivation gegeben wird, ist der Betrachtungswinkel dementsprechend wenig techniklastig. Vielmehr interessieren grundlegende Fragestellungen zu Websites und ihren Besuchern, wie sie von Betreibern und Verantwortlichen gestellt werden könnten. Der Ansatz ist also eher ein betriebswirtschaftlicher, mit einer Differenzierung von Data Mining und hypothesengestützter Analyse wird am Ende des Kapitels aber der Übergang zur technischen Betrachtung gelegt, die im weiteren Verlauf der Arbeit dominieren wird.

2.1 Wozu Analysen von Web Logdaten?

Versetzen wir uns in die Rolle eines Betreibers einer Website. Beispielsweise eines Online-Shops zum Verkauf von Weinen. Neben der Auflistung der einzelnen Produkte und Bestellmöglichkeiten in Warenkorbform, biete diese Website mit einem Weinlexikon auch allgemeine Informationen für Weinliebhaber an.

Abbildung 1 - Nicht unbedingt Wein, aber dafür massenhaft Bücher bei amazon.com!

Natürlich könnte unser Weinhändler primär die Bestellungen seiner Kunden abwickeln und sich um Warenwirtschaft kümmern. Wir stellen uns aber einen fortschrittlichen Händler vor, der um seine Kunden besorgt ist und Kundenbeziehungsmanagement oder Englisch CRM betreiben möchte. Die zu betrachtenden Akteure sind also die bereits vorhandenen Kunden, zukünftige Kunden und deren Aktivitäten, die in den Web Logs protokolliert sind. Technische Details hierzu folgen später, gehen wir zunächst zu den grundlegenden Fragestellungen über:

2.2 Welche Fragen interessieren Websitebetreiber?

Der vordergründige Antrieb des Weinhändlers ist natürlich der Verkauf von Wein an bestehende und neue Kunden. Gleichzeitig möchte er aber seine Kunden nicht nur für einzelne Bestellungen an sich binden, sondern soweit zufriedenstellen und informieren, dass sie immer neue Weinkäufe bei ihm tätigen. Das Gewinnen eines neuen Kunden ist bekanntlich wesentlich schwieriger als das Halten von bestehenden.

Dazu sind aber genaue Informationen über die Präferenzen von Kunden, ihre Vorlieben und Neigungen sowie selbstverständlich ihr Kaufverhalten nötig. Sind alle diese Kenntnisse vorhanden, können dem einzelnen Kunden gezielte Offerten gemacht oder spezielle Produkte in Kombination angeboten werden. Fragen der Reklamation können gezielter gelöst werden und der Umtausch wird durch gezieltere Bestellungen seltener.

Um die Kunden seines Online-Shops kennenzulernen, ihr Verhalten nachvollziehen und entsprechende Maßnahmen ergreifen zu können, stellt der Weinhändler stellvertretend für alle anderen Website- oder Shopbetreiber beispielsweise folgende Fragen

- Wer besucht meine Website? Wenn ich einen Shop betreibe, wer kauft dort ein? Wer sind meine Kunden? Aus welchen Ländern und Regionen kommen sie?
- Welche Seiten besuchen meine Besucher in einer Sitzung zusammen? Welche Seiten besuchen sie hintereinander (Sequenz)?
- Welche Werbemaßnahmen – beispielsweise welche Werbebanner – sollte ich für welche Kunden einsetzen? Wieviele Leute besuchen meine Website täglich?
- Wie unterscheiden sich Käufer von Nicht-Käufern? Wie mache ich einen Nicht-Käufer zum Nicht-Käufer?
- Unterscheidet sich das Verhalten von registrierten und nicht registrierten Benutzern?
- Wie erhöhe ich die Verweildauer und die Anzahl der Besuche meiner bisherigen Besucher und Kunden? Wie steigere ich ihre Zufriedenheit?

Diese und weitere Fragen lassen sich mit Web Log Mining beantworten. In den folgenden Kapiteln werden die dazu nötigen Verfahren und Vorgehensweisen vorgestellt.

Um welche Verfahren und Ansätze handelt es sich dabei?

Nun, dieser Frage wird im folgenden Abschnitt nachgegangen…

2.3 Ansätze

An dieser Stelle wird ein Überblick über die Möglichkeiten der Auswertung von Webserverprotokollierung gegeben. Dabei werden zunächst die „traditionellen“ Verfahren betrachtet, wie sie bereits aufgrund der großen Marktnachfrage in vielen Softwarepaketen zum Einsatz kommen. Mit diesen Verfahren können jedoch nur bestimmte Fragestellungen beantwortet werden. Im Anschluss werden daher Ansätze aus dem Data Mining vorgestellt, die weitergehende Informationsbedürfnisse befriedigen können.

2.3.1 „Traditionelle“

Die Technologie von Internet und im Speziellen dem World Wide Web ist eine sehr junge. Entsprechend sind auch Verfahren zur Auswertung von Nutzungsdaten noch recht jung und wenn ich hier von traditionell spreche, dann meine ich damit traditionell in Form von naheliegend und aufgrund ihrer Offensichtlichkeit schnell entwickelt.

Betrachtet man die Arbeitsweise eines Webservers, so wird ersichtlich, das sehr viele einzelne Dokumente und Bilder zwischen dem Servercomputer und einem Arbeitsrechner ausgetauscht werden. Jeder Austausch eines Objekts wird mit einigen Metainformationen wie Uhrzeit, Identifikation des empfangenden Rechners (Remotehost oder IP-Adresse – hierzu siehe später mehr) in einer Logdatei – dem Web Log – gespeichert.

Zählen von Einträgen im Web Log liefert erste Erkenntnisse

Aus diesen Daten lassen sich auf einfache Weise die Anzahl der Gesamtzugriffe über Tage, Wochen und Monate hinweg ermitteln. Man spricht von sogenannten „*Hits*“. Ebenfalls lassen sich damit die Anzahl der Aufrufe bestimmter Bereiche im Informationsangebot einer Website ermitteln, man spricht von „*PageImpressesion*“. Beide Informationen sind nützlich, um Bannerwerbung auf Websites zu verkaufen, da die inserierenden Firmen die Anzahl der

Einblendungen ihrer Banner kennen möchten und entsprechend mit dem Sitebetreiber abgerechnet wird.

Ferner sind mit diesen einfachen Mitteln, die lediglich die Anzahl von Einträgen in den Web Logs zählen, auch erste Aussagen über die Benutzer der Websites möglich. Beispielsweise lässt sich ermitteln, welche Browsertypen im Durchschnitt am häufigsten benutzt werden und in gewissen Grenzen auch aus welchen Ländern die einzelnen Besucher kommen.

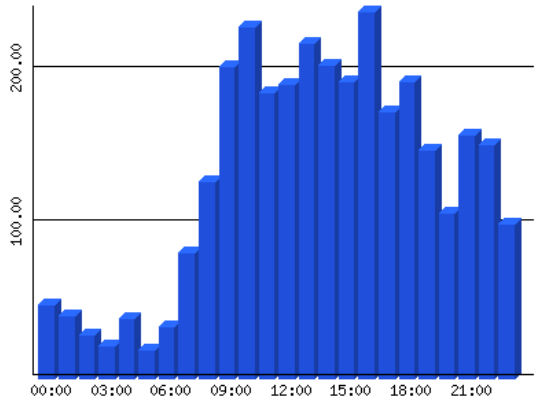

Abbildung 2 - Diagramm der Gesamtzugriffe eines Monats verteilt über die Tageszeiten

Aus diesen Informationen lassen sich einerseits Rückschlüsse über den Erfolg der Website ziehen, andererseits die Hauptnutzergruppen betrachten und für die technische Optimierung eine Ausrichtung auf die laut Statistik häufig verwendeten Webbrowser vornehmen.

Für den Betreiber der Website der Website bieten diese Angaben kumulierte Informationen über das Verhalten der Gesamtnutzerschaft. Aussagen über das Verhalten einzelner Kunden sind damit nicht zu machen. Hier bieten die Verfahren aus dem Data Mining Bereich weitergehende Konzepte an.

2.3.2 Ansätze aus dem Data Mining Bereich

Während einfache Programme zur Auswertung von Logdaten mehr oder weniger darüber informieren, dass Aktivität auf der Website stattfindet und in welchem Maße, können Data Mining Anwendungen mehr über das „Wie“ zu Tage fördern.

Zusammen besuchte Seiten - Assoziationen und Clickstreams

Aus den reinen Logdaten lassen sich mit Verfahren des Data Mining Regeln extrahieren, die Angaben über das Surfverhalten der Benutzer machen. So liefern entsprechende Algorithmen Aussagen darüber, welche Seiten in einer Sitzung zusammen besucht werden oder in welcher Reihenfolge einzelne Informationsangebote besucht werden. Man spricht bei der Abfolge einzelner Dokumente in der Informationssuche von sogenannten Clickstreams, da jeder Mausklick des Benutzers auf einen Hyperlink das Ansteuern eines neuen Dokuments bedeutet und die Folge von Mausklicken so eine Folge von Informationsseiten darstellt – einen Clickstream.

Hat ein Data Mining Verfahren solche Ergebnisse aus den Logdaten extrahiert, können die Verantwortlichen und im speziellen Marketingfachleute und Webdesigner überprüfen, ob das Ansteuern der einzelnen Informationen auch den zugrundeliegenden Ideen entspricht oder ob sich die Benutzer etwa eigene, nicht vorhergesehene Wege angeeignet haben. Entsprechend ist dann eine Umgestaltung der Website denkbar, die Einfluss auf das Surfverhalten zu nehmen versucht, beispielsweise einer Umsortierung der Informationsangebote und entsprechende Verlinkung der gewünschten Seiten. Eine regelmäßige Anwendung des Data Mining mit Interpretation der Ergebnisse und darauf folgender Umstrukturierung des Informationsangebots kann so zu einer kontinuierlichen

Verbesserung der Websiteattraktivität und einer besseren Steuerung des Benutzerverhaltens führen.

Bekannter Vertreter solcher Verfahren zur Ermittlung von Assoziationen im Benutzerverhalten im Webumfeld ist sicherlich die Website des Buchhändlers amazon.com, bei der zu jeder Buchbeschreibung auch eine Reihe von weiteren Büchern präsentiert wird, die andere Kunden zusätzlich zu dem gerade betrachteten Buch gekauft haben. Hierbei wird davon ausgegangen, dass alle oder zumindest viele Menschen, die für bestimmte Bücher Interesse zeigen auch an einer gemeinsamen Reihe anderer Bücher Gefallen finden. Zwar handelt es sich bei diesem Beispiel nicht um das Mining von Web Logs, sondern eher um die Auswertung von Bestelldatenbanken, aber das Prinzip kann auch auf Seiten einer Website angewendet werden.

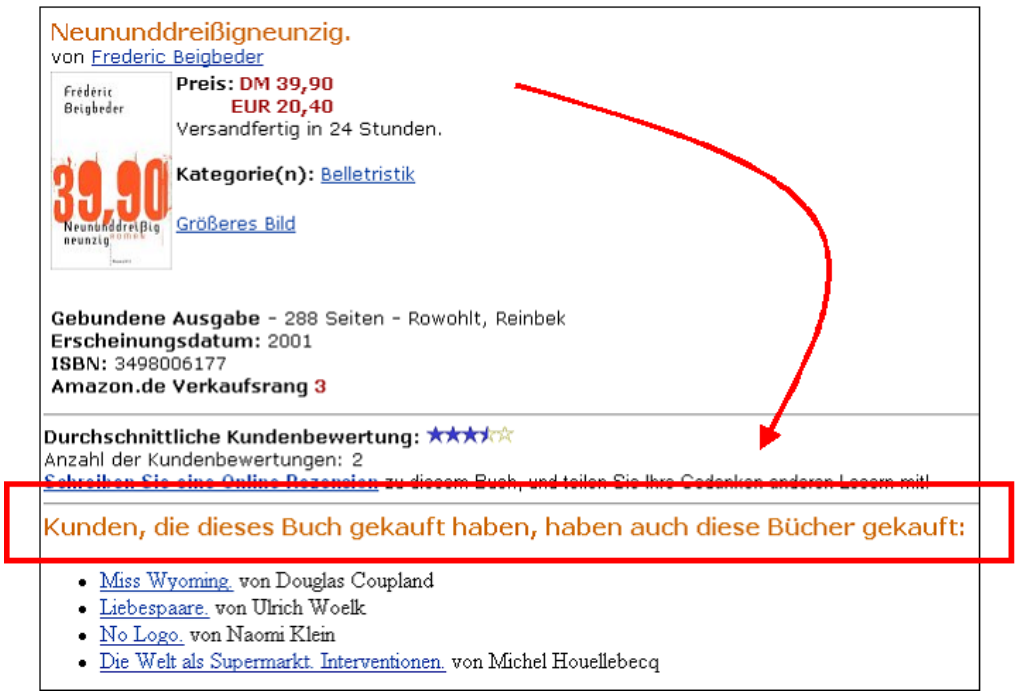

Abbildung 3 - Assoziationen von Büchern bei amazon.com

Mit der Betrachtung dieses Suchverhaltens auf der Website wird der Benutzer noch im Allgemeinen betrachtet. Seine Individualität ist zwar im Ansatz durch unterschiedliche Informationspfade erkennbar, eine Zuweisung zu konkreten Personen bleibt jedoch aus. Steht eine Benutzerdatenbank bereit, beispielsweise weil das Webangebot eine Anmeldung von Benutzern und Personalisierung ermöglicht, können die individuellen Navigationsvorgänge jedes Benutzers seinem Profil zugeordnet und gespeichert werden.

Ein Benutzer kann mehrere Sitzungen haben, in der er verschiedene Informationsangebote auf der Website angesteuert hat. Die Protokolldaten des Webservers werden hier mit einer Datenbank von Benutzerinformationen verknüpft, die beispielsweise Angaben über das Alter, Geschlecht, Einkommen und den Wohnort des Benutzers enthält. Wendet man Data Mining Verfahren auf den Gesamtdatenbestand an, kann eine *Segmentierung* der Benutzer vorgenommen werden.

Segmentierung der Benutzerdatenbank

Bei einer Community mit Online-Shop könnten durch die Software beispielsweise ein *Cluster* von Personen gefunden werden, der gerne mit anderen Mitgliedern der Community kommuniziert und bestimmte Merkmale wie Lokalität oder Alter gemeinsam hat, ein anderer Cluster könnte eher Benutzer enthalten, die im Online-Shop navigieren und Produkte bestellen, dabei über ein bestimmtes Einkommen verfügen. Der Betreiber lernt durch solche Segmentierungen grundlegende Eigenschaften seine Benutzer kennen und kann

beispielsweise Fehleinschätzungen durch eine Umgestaltung des Informationsangebots korrigieren oder die bestehende Vorgehensweise in ihrer Richtigkeit bestätigen.

Besonders interessant sind eine solche Benutzerdatenbank und Analyse für Online-Shops, um die Kundenzufriedenheit und durch stärkere Ausrichtung auf die umsatzstarken Cluster möglicherweise den Umsatz zu steigern. Zudem fließen hier noch die bisherigen Bestellungen des Benutzers mit in die Datenbank ein und erweitern sein Profil. Eine ausschließliche Betrachtung von Online-Shops wäre allerdings zu eng. Auch nicht direkt auf den Verkauf ausgerichtete Websites profitieren von Benutzerregistrierung und entsprechenden Analysen des Datenbestands, um ihr Angebot stärker auf die Zielgruppe auszurichten, mögliche Zielgruppenveränderungen zu entdecken und die Besuchshäufigkeit zu steigern.

Klassifikation der Benutzerschaft

Interessant ist der Einsatz Data Mining Verfahren auch, um eine Klassifikation von Benutzern vornehmen zu können. Anhand der bestehenden und registrierten Benutzer und der aus den Weblogs gewonnenen Verhaltensweisen lässt sich eine Klassifikation für neue Benutzer vornehmen. Hat ein Benutzer beispielsweise sein Profil bei der Neuanmeldung auf einer Website eingegeben, können ihm entsprechend seiner Klassifikation bestimmte Informationsangebote oder Produkte zum Kauf präsentiert werden.

Auch für bestehende Benutzer ist hier eine sehr dynamische Personalisierung der Website möglich, da gemäß der Klassifikation eines Benutzers bestimmte Informationen ein- oder ausgeblendet werden. Im Bereich des eCommerce ist auch Klassifizierung derart möglich, dass das Kaufverhalten von Benutzern abgeschätzt wird. Das leitet über zur

Prognose von Besucherverhalten

Während mit einer Klassifikation die aktuelle Zuordnung des Benutzers zu einer Gruppe mit bestimmten Eigenschaften gegeben ist, ist für den Betreiber der Website auch die Prognose zukünftigen Verhaltens von Interesse. Eine Einschätzung des Besucherverhaltens und ein möglicher Änderung in der Klassifikation des Benutzers kann ebenfalls mit Verfahren des Data Mining angewendet auf die mit der Benutzerdatenbank verknüpften Weblogs erfolgen.

Besonders für Online-Shops dürfte eine potentielle Veränderung der Klassifikation des Benutzers von Nicht-Käufer zum Käufer von Interesse sein, da dann mit verschiedenen Mitteln die Kauflaune des möglichen Kunden gesteigert werden kann. Entsprechend den Merkmalen bisheriger Käufer könnte beispielsweise die Content-Management-Software hellhörig werden, wenn ein Nicht-Käufer mehr und mehr Merkmale eines Käufers zeigt, beispielsweise den verstärkten Besuch von Shoppingbereichen auf der Website. Denkbar wäre dann eine dynamische Einblendung von Informationen, die auch andere Kunden des gleichen Clusters bzw. der gleichen Klassifikation zu einem Kauf veranlasst haben.

2.4 Überblick Data Mining

Web Log Mining ist in den Komplex des Web Mining einzuordnen, der zwischen Web Content und Web Usage Mining unterscheidet. Web Content Mining untersucht die Inhalte von Websites und die Abhängigkeiten dieser Informationsangebote untereinander, während Web Usage Mining die Aktivität von Benutzern auf Websites untersucht. Web Usage Mining und Web Log Mining sind daher nahezu gleichzusetzen, wobei die Fokussierung auf das Mining der Web Logs – also den Protokolldateien des Webservers – bei Web Log Mining festgelegt ist.

Was verbirgt sich aber hinter dem Begriff und der Technologie des Data Mining? Hier ein knapper Überblick...

2.4.1 Hypothesenfreie vs. hypothesengestützte Verfahren

Die Menge an elektronisch verfügbaren Datenbeständen nimmt durch immer leistungsstärkere Rechner und ihren Einsatz in praktisch allen kommerziellen und

wissenschaftlichen Anwendungsfeldern rapide zu. Gleichzeitig besteht ein immer stärkeres Bedürfnis, den Datenbergen Informationen zu entlocken und diese nutzbringend für die entsprechenden Anwendungen einzusetzen. Leider ist der Mensch mit der Auswertung großer Datenmengen ohne die richtigen Werkzeuge heillos überfordert.

Abhilfe ist nötig, um Informationen aus den Daten zu gewinnen und im Fall des Web Log Mining die Erzeugung von Protokolldaten nicht zum Selbstzweck des Webservers werden zu lassen. Die Lösung ist auch wieder in der Informationstechnik zu suchen, die einerseits zwar erst zu Datenbeständen in erheblicher Größenordnung geführt hat, sich aber selbst gewissermaßen hilft und versucht, die entsprechenden Tools für die Betrachtung und Analyse der Daten zu liefern.

Im Groben kann man hier zwischen zwei Lösungsansätzen unterscheiden, der *hypothesengestützten* und der *hypothesenfreien* Entdeckung von Information. Hier sei noch anzumerken, dass zwar bisher abstrakt von Datenbeständen gesprochen wurde, diese aber in einer dafür geeigneten technischen Form gelagert werden müssen. Beispielsweise in Datenbanken, Protokolldateien oder einzelnen Dateien. In Zusammenhang mit Data Mining tauchen öfters die Begriffe *Data Warehouse* oder *Data Mart* auf, die letztlich gesonderte Datenbanksysteme meinen, die nicht den eigentlichen Produktionssystemen entsprechen, sondern speziell für die Datenanalyse bereitgestellt werden, um den Produktionsprozess nicht zu stören.

Aber zurück zur Entdeckung von Informationen. Bei der hypothesengestützten Analyse besteht seitens der Anwender bereits eine Vorstellung – eine Hypothese – eines Modells, dem die Daten folgen. Mit Hilfe von hypothesengestützten Verfahren kann diese Hypothese anhand des Datenbestands überprüft werden. Hierzu werden beispielsweise einfache Methoden wie Volltextsuche bei Textdateien oder SQL-Anfragen bei strukturierten Datenbankdaten eingesetzt. Statistiksoftware bietet weitergehende Verfahren, um die aufgestellten Hypothesen zu prüfen.

Weitere Analysemöglichkeiten, die den hypothesegestützten Verfahren zuzuordnen sind, stellen interaktive Instrumente dar, bei denen der Anwender durch sukzessive Parameterveränderungen Daten betrachtet und Modelle überprüfen kann. Hier ist das große Thema OLAP zu nennen.

Dabei handelt es sich um Verfahren, bei denen multidimensionale Datenbestände in Form eines Datenwürfels (Hypercube) von verschiedenen Seiten interaktiv betrachtet werden können. Der Anwender kann mit solchen Utilities verschiedene Sichten auf große Datenbestände erhalten und dadurch Strukturen und Muster erkennen. Interessant sind auch alternative Verfahren, bei denen visuelle Abfragen auf große Datenbestände möglich sind und durch Verbergen von Informationen eine Reduktion an Komplexität vorgenommen werden kann. Auch so können Hypothesen von Modellen bestätigt oder widerlegt werden (weitere Details unter „Dynamic Queries“, Quelle [6] im Literaturverzeichnis).

Wesentliches Merkmal aller hypothesengestützten Verfahren ist, dass der Anwender die zentrale Rolle einnimmt. Er muss über intensive Kenntnisse des untersuchten Datenmaterials und den Einsatz der Verfahren verfügen und plausible Hypothesen aufzustellen in der Lage sein. Man spricht daher auch vom benutzergetriebenen Vorgehen, dass der Verifikation von Modellen dient.

Abbildung 4 - Data Mining findet selbstständig Muster in großen Datenmengen

Anders dagegen die Verfahren der hypothesenfreien Entdeckung von Information, bei denen die Algorithmen – also die Maschine – autonom nach Informationen in der Datenbasis suchen.

Der Anwender geht dabei ohne konkrete Vorstellungen über die aus den Daten zu fördernden Modelle an die Analysetätigkeit heran. Die Software durchforstet den Datenbestand auf verschiedene Arten und liefert selbstständig Muster und Regeln zurück, die dann vom Anwender verwendet und auf ihre Plausibilität überprüft werden müssen. Die Instrumente des Data und Text Mining sind zu diesen hypothesenfreien Verfahren zu zählen. Man spricht im Gegensatz zu den benutzergetriebenen, hypothesengestützten Analysemethoden von datengetriebenen Verfahren zur Aufdeckung von Information.

Folgendes englisches Zitat verdeutlicht noch einmal den Kontrast zwischen hypothesenfreien und hypothesengestützten Verfahren

> *„Our goal is to challenge the data to ask questions, rather than asking questions to the data" (siehe "Business Intelligence", Quelle [3], S. 179)*

Welche Analysemethoden dem Bereich Data Mining gemäß obiger Definition zugeordnet werden können und welche im Speziellen für Web Log Mining zum Einsatz kommen, wird im folgenden und späteren Abschnitten erläutert werden.

2.4.2 Wesentliche Verfahren des Data Mining

Im Groben lassen sich beim Data Mining Fragestellungen der Assoziation, Segmentierung, Klassifikation und Prognose auf Datenbeständen angehen. Welche Anwendungen für diese Verfahren beim Web Log Mining in Frage kommen, wurde weiter oben bereits skizziert. Hier noch einmal ein schematischer Überblick in Form einer Grafik, der Fragestellungen des Web Log Mining aufführt aber auch allgemein betrachtet werden kann.

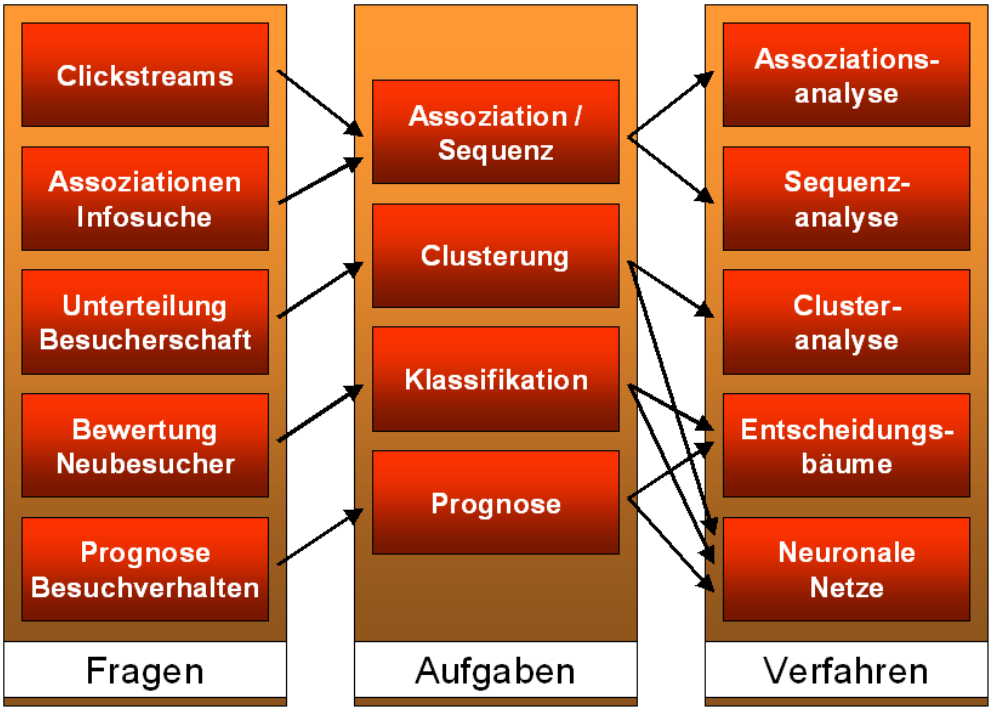

Abbildung 5 - Fragen, Aufgaben und Methoden beim Web Log Mining

Im linken Teil sind einige der weiter oben bereits aufgeführten Fragestellungen genannt, denen in der Mitte die entsprechenden Aufgaben des Data Mining zugeordnet sind. Auf der rechten Seite schließlich die technischen Verfahren, deren populärste Vertreter Assoziations-, Sequenz- und Clusteranalyse sowie Neuronale Netze und Entscheidungsbäume sind.

3 Die Domäne des Web Log Mining

Im Folgenden wird ein Überblick über den Lebensraum von Web Log Mining Aktivitäten gegeben. Dabei werden die Architektur von Webservern, das daran angeschlossene Protokollmodul sowie das Format der erzeugten Protokolldaten – den Logdateien – vorgestellt.

Dabei wird auch auf die Schwierigkeiten eingegangen, die bei der Verarbeitung der Logdateien, dem Rohmaterial und Untersuchungsgegenstand des Web Log Mining, auftreten und zu beachten sind. Hier werden auch erste Hinweise zum Preprocessing der Logdateien gegeben, die im folgenden Kapitel in die Praxis umgesetzt werden. Aber zunächst zum Aufbau von Webservern.

3.1 Webserverarchitektur

Unter dem Begriff Webserver versteht man im Internet-Jargon einen Rechner, der Anfragen von Benutzern mit der Lieferung der entsprechenden Webdaten an die anfordernden Rechner befriedigt. Letztlich verrichtet dabei eine Serversoftware – auch Daemon genannt – auf dem Serverrechner ihre Arbeit und wartet auf Anfragen. Kommt eine solche Anfrage nach einem *Requestobjekt* wie einer Html-Datei, einer Grafik oder auch einem Java-Applet, sucht die Serversoftware im Dateisystem nach dem entsprechenden Objekt oder ruft ein Skriptprogramm auf, das dynamischen Inhalt generiert. Anschließend werden die *Daten* an den aufrufenden Rechner gesendet (das eigentliche „*serven*“ wenn man so will).

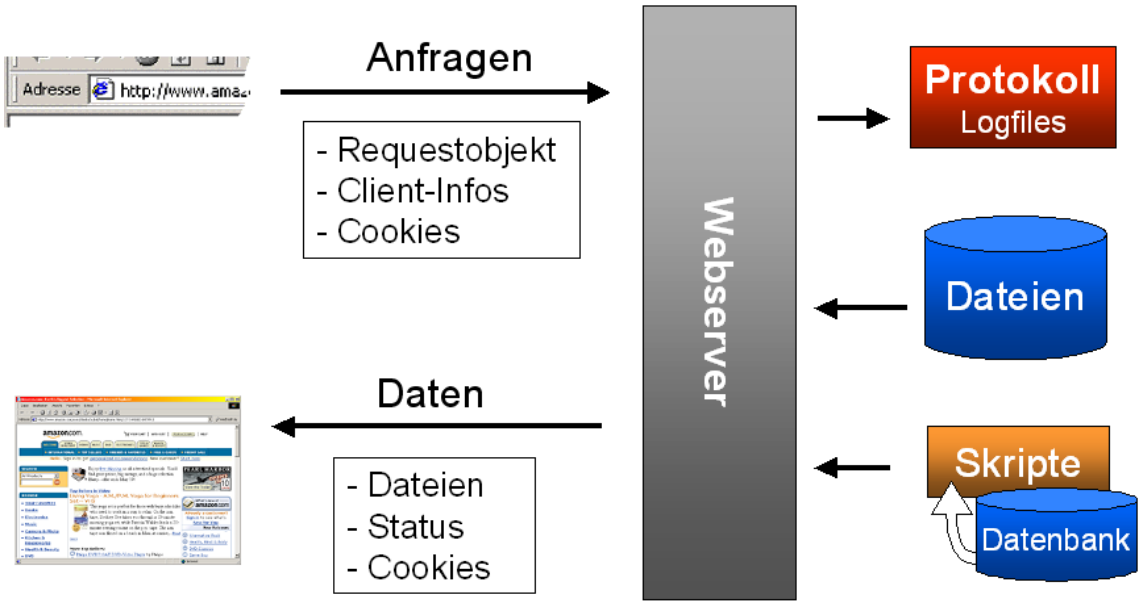

Abbildung 6 - Architektur von Webservern

Dabei wird jeder Zugriff von aufrufenden Rechnern in einem Protokoll – der Logdatei – auf dem Server gespeichert. Gespeichert werden das angeforderte Objekt, also ein Dateiname auf eine Grafik, ein Html-Dokument oder ein Skript sowie Uhrzeit und Informationen, die vom Arbeitsrechner mit dem Request geliefert werden. Die Protokollierung erfolgt in Form einer Textdatei, in der jede Zeile einen Zugriff repräsentiert. Im folgenden Abschnitt wird genauer auf dieses Format eingegangen.

Anzumerken ist noch, dass neben den eigentlichen Nutzdaten aus dem angeforderten Objekt weitere Informationen zwischen Arbeitsrechner und Server ausgetauscht werden. Dabei handelt es sich einerseits um Metainformationen wie Browsertyp und zuletzt besuchte Seite auf dem Client oder Dateigröße und -typ des gelieferten Objekts, andererseits um sogenannte Cookies.

Cookies stellen kleine Datenspeicher dar, die vom Webserver auf dem Clientrechner angelegt werden können. Da aktiv Operationen auf dem Client vorgenommen werden, ist der Aktionsradius von Cookies durch geringe Größe und feste Bindung an die jeweilige Website eingeschränkt. Sicherheitsrisiken für den Anwender des Arbeitsrechners bestehen im Allgemeinen nicht, auch wenn sich hier hartnäckige Gerüchte halten. Vorteil der Cookie-Technik ist, dass das an sich zustandslose http-Protokoll zwischen Webserver und Arbeitsrechner um einen Zustand ergänzt werden kann.

Bei jedem Aufruf einer Seite kann der Webserver einen „Cookie“ an den Arbeitsrechner senden, der ihn speichert und bei jedem weiteren Seitenanruf an den Webserver mitschickt. Mit einem Cookie kann eine Nummer zur Identifikation einer Arbeitssitzung beim ersten Seitenanruf an den Client geschickt werden und bei jedem weiteren Seitenaufruf wird sie via Cookie an den Webserver zurückgesendet. So kann ein Skript auf dem Webserver diese Nummer einem Kontext zuordnen und beispielsweise personalisierte Inhalte ausliefern.

3.2 Logdateiformat

Auf dem Markt für Webserversoftware sind eine Reihe von Produkten zu finden, die aber glücklicherweise ein gemeinsames Format für Protokolldateien benutzen, das sogenannte Common Log Format. Bei den Produkten handelt es sich in erster Linie um den Open-Source-Server Apache, den Internet Information Server von Microsoft, Netscapes iPlanet sowie Software von NCSA und CERN (unter http://www.netcraft.com finden sich aktuelle Statistiken zu den Marktanteilen der einzelnen Serverprodukte).

Common Log Format (CLF)

Trotz unterschiedlicher technischer Ansätze der Webserverprodukte wird das ehemals von NCSA (National Center for Supercomputing Applications) entworfene Common Log Format (CLF) eingesetzt, das sich als Standard für Protokolldaten etabliert hat. Dabei handelt es sich um ein datensatzbasiertes Format, das jeden Zugriff auf den Webserver mit verschiedenen Attributen festhält. Wesentlich ist, dass die Daten in einer ASCII-Datei gespeichert werden und jede Zeile einem Zugriff entspricht. Im Folgenden ein Auszug einer solcher Logdatei.

```
ao3-m182.net.hinet.hr - - [08/Jun/2001:01:00:10 +0200] "GET /auswahl.phtml HTTP/1.1" 200 4065
ao3-m182.net.hinet.hr - - [08/Jun/2001:01:00:13 +0200] "GET / HTTP/1.1" 200 3731
ao3-m182.net.hinet.hr - - [08/Jun/2001:01:00:25 +0200] "GET /index.html HTTP/1.1" 200 4731
ao3-m182.net.hinet.hr - - [08/Jun/2001:01:00:30 +0200] "GET /auswahl.phtml HTTP/1.1" 200 4065
141.2.114.129 - - [08/Jun/2001:01:00:32 +0200] "GET /internet/sonstige/12/ HTTP/1.0" 200 91053
141.2.114.129 - - [08/Jun/2001:01:00:40 +0200] "GET /a.phtml?fkatid=146 HTTP/1.1" 200 21817
ao3-m182.net.hinet.hr - - [08/Jun/2001:01:00:49 +0200] "GET /auswahl.phtml HTTP/1.1" 200 4065
ao3-m182.net.hinet.hr - - [08/Jun/2001:01:00:51 +0200] "GET / HTTP/1.1" 200 3731
sigma.mails-media.de - - [08/Jun/2001:01:02:05 +0200] "HEAD / HTTP/1.0" 200 0
141.2.114.129 - - [08/Jun/2001:01:05:52 +0200] "GET /dienste/c.html/ HTTP/1.0" 200 62455
```

Abbildung 7 - Auszug aus einer Logdatei im Common Log Format (CLF)

Die einzelnen Felder eines Datensatzes und damit einer Zeile sind durch Leerzeichen voneinander getrennt. Zeichenketten mit Leerzeichen sind in Anführungszeichen eingeschlossen, um Mehrdeutigkeit zu verhindern. Eine Zeile enthält dabei sieben Attribute, was mit konkreten Beispieldaten wie in folgender Tabelle aussehen könnte.

remotehost	rfc931	authuser	Datum	requeststring	status	bytes
141.2.114.129	-	-	[08/Jun/2001:01:05:52 +0200]	"GET / HTTP/1.1"	200	3731

Im Feld *remotehost* wird der Rechnername oder die IP-Adresse des an den Webserver anfragenden Rechners gespeichert. Ist mittels DNS (Domain Name Service) keine Auflösung des Rechnernamens möglich, wird die IP-Adresse gespeichert. Insbesondere bei Benutzern, die einen nicht permanenten Internetzugang verwenden, wird häufig nur eine IP-Adresse geliefert. Zudem teilen sich in der Regel mehrere Benutzer eine IP-Adresse oder einen

Hostnamen. Sei es entweder parallel beim Einsatz von Proxy-Servern in Unternehmen oder seriell bei Internet-Service-Providern, die nur über einen begrenzten Pool von Adressen verfügen. Zu beachten ist daher, dass eine Identifikation des tatsächlichen Benutzers oder gar Kunden anhand dieses Feldes in der Praxis nahezu unmöglich ist. Zur Unterscheidung zweier Benutzer in einer kurzen Zeitspanne kann das Feld *remotehost* allerdings mit herangezogen werden, wie später noch gezeigt wird.

Mit *rfc931* ist ein Feld bezeichnet, das den Eigentümer einer Verbindung zum Zeit des Zugriffs aufnimmt (siehe http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc0931.html, wurde obsolet durch rfc1413). In der Praxis der Logdateien hat es aber keine Bedeutung erlangt und ist daher leer, was durch ein Minus „-“ gekennzeichnet ist. Dies gilt übrigens auch für alle anderen Felder, die ein „-“ aufweisen, wenn sie leer sind.

Das Attribut *authuser* enthält den Namen des Benutzers, falls es sich um einen Zugriff mit Benutzerauthentifizierung handelt. Hierbei müssen Name und Passwort an den Webserver übermittelt werden, wodurch ein Zugriffsschutz für Bereiche von Websites realisiert werden kann. Bei anonymen Verbindungen ist dieses Feld ebenfalls leer und enthält ein Minus „-“.

Im folgenden Feld *date* wird die Uhrzeit des Zugriffs gespeichert gefolgt vom Feld *requeststring*, das den Befehl enthält, den der aufrufende Rechner an den Webserver stellt. Bei den Befehlen handelt es sich um die Kommandos es http-Protokolls. Am häufigsten ist hier der GET-Befehl mit einer Pfadangabe zu einem Dokument auf dem Webserver zu finden, der ein Objekt – beispielsweise ein Bild oder eine Html-Datei – inkl. seiner Metadaten zur Auslieferung anfordert. Ebenfalls ist der HEAD-Befehl zu nennen, der nur die Metadaten zu einem Objekt liefert sowie der POST-Befehl, der das Senden von größeren Datenmengen vom aufrufenden Rechner an den Webserver ermöglicht.

Dieses Feld repräsentiert die eigentlich angeforderte Information, die im Web Log Mining primär von Interesse ist und die mit den Verfahren des Data Mining analysiert wird. Eine genaue Betrachtung des Feldinhalts, insbesondere der Pfadangabe, ist daher essentiell für den Erfolg der Miningaktivitäten.

Zusätzlich werden zu jedem Zugriff im Feld *status* der zurückgelieferte Rückgabecode, der den Erfolg oder Misserfolg des Datentransfers sowie Fehler in der Webserversoftware beim Verarbeiten dieser Anfrage beschreibt, und im Feld *bytes* die Anzahl der im Erfolgsfall übertragenen Bytes gespeichert.

Extended Log Format

Zusätzlich zu den oben aufgeführten Feldern sind häufig noch zwei weitere Felder in den Logdateien anzutreffen. Dabei handelt es sich um das Feld *referrer*, das die vor dem aktuellen Zugriff besuchte Seite im Browser des Anwenders angibt und das Feld *user_agent*, das den Namen und Hersteller des verwendeten Browsers aufnimmt.

…	status	bytes	referrer	user_agent
	-	-	`http://www.heise.de/`	`Mozilla/4.0 (compatible; MSIE 5.5; Windows 98)`

Das Feld *user_agent* hat für sich genommen eine gewisse Aussagekraft, weil durch die Ermittlung der Anteile der einzelnen Browsertypen in den Protokolldaten entsprechende Optimierungen der Html-Dateien vorgenommen werden können. Auch identifizieren sich Suchmaschinen mit einer eigenen Kennung im diesem Feld. Entsprechende Zugriffe können also leicht erkannt werden. Unter Gesichtspunkten des Web Log Mining sind beide Felder nützlich, um Unterschiede zwischen einzelnen Zugriffen besser unterscheiden zu können. Das Feld *referrer* bietet zudem die Möglichkeit Pfade des Besuchers innerhalb der Website zurückzuverfolgen und bei fehlenden Daten Interpolationen vorzunehmen.

3.3 Hits, Visits und Sessions – Begriffsklärung

Das im World Wide Web verwendete Hypertext Transfer Protocol oder kurz http, mit dem Dokumente, Grafiken und Applets vom Webserver zu den Arbeitsrechnern transportiert werden, ist ein zustandsloses Übertragungsprotokoll. Das bedeutet, dass es zwischen zwei Zugriffen auf Objekte des Webservers von Natur aus keinen Zusammenhang gibt. Somit steht jeder Zugriff gleichberechtigt neben dem anderen. Ein beliebiger Zugriff wird auch als *Hit* bezeichnet.

Da wie im vorherigen Abschnitt erläutert der zugreifende Rechner mit dem Logdateifeld *remotehost* aufgrund der Knappheit von IP-Adressen nicht eindeutig identifiziert werden kann und ob des Einsatzes von Proxy-Servern, bei dem sich mehrere Benutzer eine IP-Adresse teilen, ebenfalls kein sicherer Rückschluss auf einen Benutzer möglich ist, ist die Identifikation einer Arbeitssitzung – auch Session genannt – nicht eindeutig möglich.

Vergleicht man aber die einzelnen Datensätze in Bezug auf die Felder *remotehost*, *date* und *user_agent* untereinander, so sind dennoch Ableitungen von *Sessions* – oder in der Fachterminologie der Werbetreibenden *Visits* genannt – möglich, wenngleich auch nur mit einer gewissen Wahrscheinlichkeit, was aber der späteren Anwendung der Data Mining Verfahren keinen Abbruch tut. Man spricht von *Transaktionsableitung* der Zugriffsdaten aus dem zustandslosen http-Protokoll. Bezieht man hier zusätzlich das Feld *referrer* ein, sind noch präzisere Zusammenhänge zwischen einzelnen Datensätzen der Logdatei möglich und die Wahrscheinlichkeit, die richtigen Zugriffe einer gemeinsamen Session zuzuordnen, steigt. Eine Session, die im Durchschnitt eine Dauer von 25 Minuten hat, kann man gemäß folgender Definition auffassen.

$$Session_{BenutzerA,20010521} = (\text{index.html,seite1.html,seite2.html,...})$$

Die Sitzung und damit der Besuchszeitraum eines Websitebesuchers wird damit durch den Zeitpunkt (hier 2001-05-21), eine mögliche Identifikation des Besuchers (hier BenutzerA) sowie und vor allem die Abfolge der besuchten Seiten charakterisiert. Letzteres ist ein wichtiger Schritt für die Vorverarbeitung der Logdaten, um sie den Data Mining Verfahren zuzuführen. Während die Daten in der Logdatei html-Dateien (= Seiten!), Grafiken und andere eingebettete Elemente enthalten, sind für das Mining nur die tatsächlichen abgerufenen Seiten von Interesse. In der Werbesprache spricht man auch in der Summe dieser Seitenabrufe von *PageImpressions.*

3.4 Datenanreicherung – Aggregation von Datenquellen

Werden nur die reinen Protokolldaten betrachtet und sind registrierte Benutzer oder Kunden für die Untersuchung nicht von Interesse, dann bieten die Logdateien und die daraus mit der im letzten Abschnitt vorgestellten *Transaktionsableitung* extrahierten Sessions schon eine gute Basis für Data Mining. Allerdings erlaubt eine Verbindung der Log- mit den Personendaten wie im Motivationskapitel beschrieben eine wesentlich weitreichendere Perspektive für das Web Log Mining. Betrachten wir eine solche Verknüpfung also näher.

3.4.1 Benutzerdatenbanken

Besteht für Besucher einer Website die Möglichkeit, sich anzumelden und fortan als registrierter Benutzer durch das Informationsangebot zu navigieren, so liegt es nahe, das Surfverhalten des einzelnen Besuchers in Verbindung mit seinem Profil zu betrachten. Besucher profitieren von einer Registrierung auf einer Website, weil sie sich beispielsweise das Informationsangebot nach eigenen Vorstellungen zusammensetzen können und personalisierten Content erhalten, sie an Diskussionsrunden teilzunehmen berechtigt sind oder einen Newsletter erhalten, der sie über Neuigkeiten und neue Produkte auf der Website informiert. Im Gegenzug verliert der registrierte Benutzer einen Teil seiner Anonymität und liefert dem Websitebetreiber bei der Registrierung wertvolle Informationen wie Wohnort,

Alter, Einkommensstufe und Interessenslage, mit denen sich Betrachtungen über die Zielgruppe aber auch direkte Kundenansprache realisieren lassen.

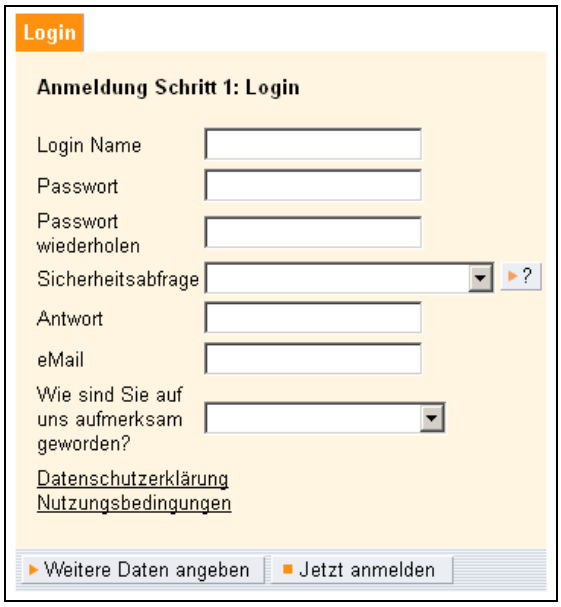

Abbildung 8 - Anmeldeformular für Benutzer (siehe http://www.moneyshelf.de/)

Ein Formular zur neuen Anmeldung eines Benutzers ist in Abbildung 8 gezeigt. Neben den obligatorischen Angaben eines Benutzernamens, Passworts und der Emailadresse wird hier auch gefragt, wie der Besucher auf das Angebot aufmerksam geworden ist. Weitere Daten können optional auf einer anderen Seite angegeben werden. Ein richtiges Maß zwischen Aufdringlichkeit und mäßiger Neugier seitens des Betreibers liefert ein gutes Mittel zwischen Menge an verwertbaren, nutzbringenden Informationen und Ablehnung der Besucher aus Datenschutzbedenken.

Die Benutzerdatensätze werden in einer entsprechenden Datenbank gespeichert. Bei jedem wiederholten Besuch auf der Website meldet sich der Benutzer neu an oder wird mittels eines clientseitigen Cookies automatisch neu angemeldet. Fortan besucht er die Seite als registrierter Benutzer. Realisiert wird dies durch einen Parameter in der URL, der die Session und damit indirekt den Benutzer identifiziert oder einen Cookie mit der gleichen Information, beispielsweise mit der folgenden URL

https://ssl.test.com/view.jhtml;$sessionid$P4AB000FXLOPKCQCECCSF

Durch die Verbindung der Session mit einem Benutzeraccount hat man gleich zwei Fliegen mit einer Klappe erschlagen, da ein Zugriff jetzt über die Sessionid eindeutig einer Sitzung zugeordnet werden kann und gleichzeitig nicht-anonyme Zugriffsdaten vorliegen. Allerdings muss hier auch erwähnt werden, dass schon der bloße Einsatz einer Sessionid in der URL die aufwändige Transaktionsableitung überflüssig macht – auch ohne Benutzerregistrierung.

In diesem Abschnitt geht es allerdings um die Anreicherung der Log- um Personendaten. So ist einerseits jedem Zugriff über die ebenfalls protokollierten Sessionid ein Benutzeraccount zugeordnet. Andererseits kann auch zu jedem Benutzeraccount eine Reihe von Sessions gespeichert werden. Während der eigentlichen Protokollierung dürfte hier kein Unterschied in der Vorgehensweise sein. Beim Aufbau eines Data Mart zur Anwendung der Mining Verfahren entscheidet man sich dann für eine Richtung, je nachdem ob man eher Fragestellungen in Bezug auf den einzelnen Benutzer oder auf die Seitenstruktur beantworten möchte.

3.4.2 Weitere Datenquellen und Protokolldaten

Die vom Webserver erzeugten Logdaten und die bei Websites mit Benutzerregistrierung vorhandene Benutzerdatenbank stellen die primären Säulen des Web Log Mining dar. Zunächst müssen diese Daten in einer für das Mining geeigneten Form vorliegen, worauf später in den einzelnen Miningschritten eingegangen wird.

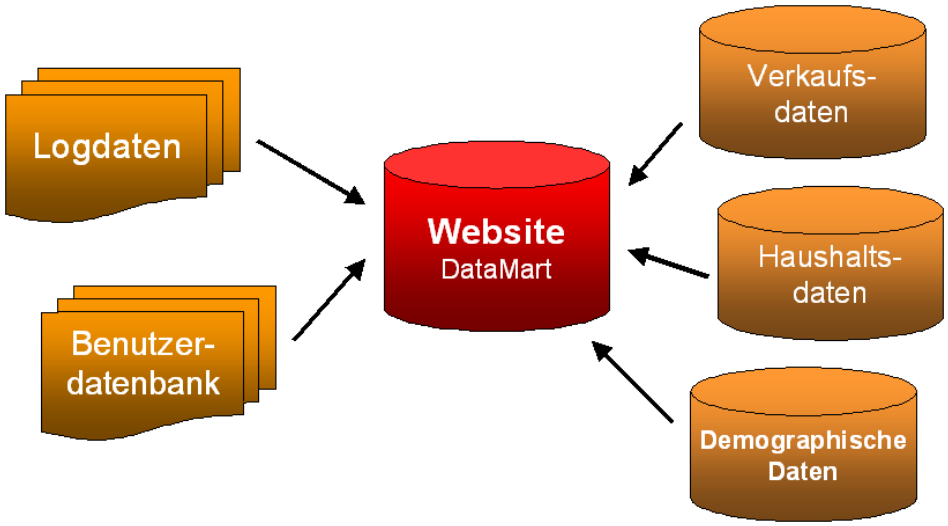

Abbildung 9 - Verschiedene Datenquellen können in die Analyse fließen

Interessant ist aber auch die Einbeziehung weiterer Datenquellen. Beispielsweise können bei Online-Shops die Protokolle über den Kauf von den von einzelnen Kunden gekauften Produkten einbezogen und damit eine Analyse des Warenkorbs durchgeführt werden.

$$Käufe_{BenutzerA,2001-06} = (\text{Produkt\_B,Produkt\_X,Produkt\_T,...}\ )$$

Aus solchen Informationen lassen sich dann beispielsweise von einem Kunden zusammen gekaufte Produkte extrahieren und Regeln für ein allgemeines Kaufverhalten ableiten, was dann zur Klassifikation anderer, in einigen Attributen gleicher Benutzeraccounts, genutzt werden kann. Entsprechend kann wie bei der Untersuchung von zusammenbesuchten Seiten auch bei zusammengekauften Produkten eine gezieltere Schaltung des Contents vorgenommen werden, um Besucher zum Kauf oder auch nur zur weiteren Informationssuche zu animieren (siehe dazu auch den Screenshot von amazon.com bei „Abbildung 3 - Assoziationen von Büchern bei amazon.com“).

Während diese Anreicherung der Protokoll- und Benutzerdaten aus konkretem Benutzerverhalten resultiert, sind andere Aggregationen mit allgemeineren Erfahrungswerten denkbar. Beispielsweise können *Haushaltsdaten* (auch als *Paneldaten* bekannt) und *Demographische Daten* einfließen, die ein allgemeines Verhalten von Menschen einer bestimmten Region, in bestimmten Alters- und Einkommensklassen sowie für bestimmte Produktgattungen aufzeigen. Aufgaben im Bereich der Klassifikation und Prognose erhalten durch die Anreicherung mit solchen Daten, die aus statistischen, teilweise repräsentativen Erhebungen stammen, eine stärkere Fundierung als die reinen, aus dem Websitebetrieb ermittelten Protokolldaten, können aber auch das Bild verfälschen oder wertlos sein, wenn allgemeine Daten und Benutzerschaft der Website zu stark voneinander abweichen.

4 Der Prozess des Web Log Mining

Nachdem wir uns motivierende Gedanken über das erhebliche Bedürfnis nach Web Log Mining gemacht haben, Data Mining als zugrundeliegende Basistechnik vorgestellt und auch die Datenkomponenten in Form von Webserverlogdateien vorgestellt wurden, wird es Zeit, die als Web Log Mining bezeichneten Vorgehensweise zu betrachten.

Der Prozess des Web Log Mining kann in einige Teilaufgaben zerlegt werden. In dieser Arbeit wurde eine Zerlegung in vier Schritte gewählt, die im folgenden Schaubild illustriert ist.

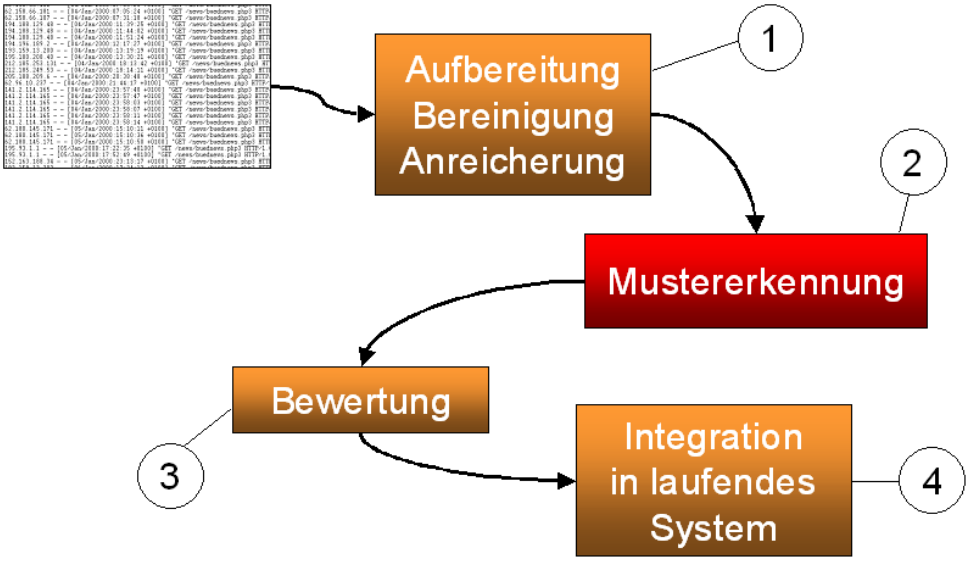

Abbildung 10 - Die Abläufe beim Web Log Mining

Der zeitaufwändigste Schritt bei der Datenanalyse der Website ist die Erfassung, Extraktion und Vorverarbeitung der Serverdaten, was sowohl die elementaren Logdateien des Webservers als auch weitere Datenquellen wie Benutzerdatenbanken einschließt. Dieser Schritt ist im Schaubild mit 1 bezeichnet. Aufwändig ist er deswegen, weil verschiedene Parameter individueller Themengebiete wie Marketing, Technik oder Human Resources von verschiedenen Personen gesetzt werden müssen, um eine optimale Auswertung der produzierten Daten vornehmen zu können. Als kleine Anmerkung sei hier vorausgesagt, dass die Kundenbindung von Unternehmen in Zukunft immer mehr an Bedeutung gewinnen dürfte und sich darauf aufbauend die Website als das primäre und direkte Medium zwischen Käufer und Verkäufer etablieren könnte. Insbesondere bei sehr kundenreichen Unternehmen sind Verlagerungen vom telefonischen oder direkten Kontakt hin zur Schnittstelle Web aus Kostengründen denkbar.

Vorverarbeitung und Data Mining

Dieser kleine Ausblick verdeutlicht, dass es ob der Bedeutung einer Website als neuem Medium wichtig ist, verschiedene Fachleute in die Planung von Architektur und Betrieb zu involvieren. Die richtigen Entscheidungen im Vorfeld tragen dazu bei, dass die von der Software generierten Daten zu einem späteren Zeitpunkt sinnvoll ausgewertet und dadurch Aussagen zur weiteren Entwicklung der Site gemacht werden können. Von technischer Seite ist es beispielsweise wichtig, dass einzelne Benutzersitzungen – oder wie es hier bislang als Session bezeichnet wurde – in den Logdateien klar voneinander abgetrennt werden, um genau Analysen über das Surfverhalten der Besucher machen zu können (siehe hierzu auch die Transaktionsableitung in Abschnitt 3.3 und Abschnitt 4.2).

Die Vertriebsabteilung liefert ihrerseits beispielsweise Vorgaben für die von Benutzern zu erfragenden Informationen wie Alter, Geschlecht oder Einkommenslage, um eine spätere Segmentierung der Benutzerschaft vornehmen und die Besucher identifizieren zu können.

Eine Verbindung von Logdaten und Personendatenbank muss ebenfalls realisiert und weitere Datenquellen falls vorhanden hinzugezogen werden (Aggregation). Eine andere Aufgabe kommt den Content- und Produktverantwortlichen hinzu, die im Vorfeld eine Auswahl an Themen und Produktkategorien sowie deren Anordnung vorgeben, an der idealerweise nach und abhängig von den Ergebnissen des Web Log Mining Anpassungen vorgenommen werden.

Sind die Daten in der gewünschten Form schließlich gesammelt, aggregiert und vorverarbeitet, können sie in dem im Schaubild als Schritt 2 bezeichneten Vorgang mit Verfahren des Data Mining analysiert werden. Dazu werden die üblicherweise in einem Datenbanksystem gespeicherten Daten einer Data Mining-Software zugeführt, die verschiedene Verfahren einzeln oder in Kombination abarbeitet und die daraus resultierenden Ergebnisse in verschiedenen Formaten bereitstellt. Beispielsweise werden Regeln in textueller Form, als Anweisungen, die direkt in Programmiersprachen übernommen werden können oder mit visuellen Mitteln wie Diagrammen ausgegeben.

Bewertung und Integration in das laufende System

Stehen Regeln und Muster oder – allgemeiner betrachtet – neue Informationen durch das Mining zur Verfügung, werden sie in Schritt 3 interpretiert. Hierzu kommen wieder Personen verschiedener Fachgebiete zum Zuge, je nachdem, in welche Richtung die Fragestellungen und Analysen gegangen sind. Beispielsweise geht es um Fragen, die das Surfverhalten betrachtet haben und die dem Webmaster wichtige Orientierungshilfen geben oder auch um Analysen der Besucher- und Kundenstruktur, die eher für Vertrieb und Marketing von Interesse sind.

Von zentraler Bedeutung ist, dass in diesem dritten Schritt aus den nüchternen Ergebnissen konkrete Entscheidungen abgeleitet werden, die im vierten und letzten Schritt in die bestehende Website und Internetpolitik des Betreibers integriert werden können. Damit schließt sich die Prozesskette des Web Log Mining, einem Prozess, der in regelmäßigen Abständen manuell oder automatisiert und während des laufenden Betriebs der Website vonstatten gehen kann. Aufgrund der Vielschichtigkeit der Fragestellungen von Websitebetreibern ist der Prozess äußerst fallbezogen, also für jede Website nicht nur in Nuancen anders. Inwieweit hier eine Standardisierung erfolgen kann, bleibt zu beobachten. Hierauf wird im Ausblick noch ein wenig näher eingegangen (siehe Abschnitt 5).

Zunächst aber eine detaillierte Vorstellung der einzelnen Prozessschritte sowie konkrete Fragestellungen, wie sie von Websitebetreibern gestellt werden können, nebst Verfahren zu deren Beantwortung.

4.1 Natur der Fragestellungen

Das World Wide Web enthält mittlerweile Informationen zu praktisch allen Lebensbereichen, Wissensgebieten und Tätigkeiten, die Menschen interessieren. Die Tendenz ist weiter steigend und wenn man Suchmaschinen bedient, die Angebote des World Wide Web indizieren, wird man bei nahezu jeder eingegeben Frage auf die eine oder andere Weise fündig. Was natürlich keinesfalls heißt, dass das Internet Antworten zu allen erdenklichen Fragen parat hält, lediglich partielle Informationseinheiten in kleiner oder großer Ausführung sind verfügbar.

Lange Rede, kurzer Sinn dieser Einleitung: Die Gesamtheit aller Websites deckt damit wohl empirisch betrachtet einen großen Teil des kollektiven menschlichen Wissens ab. Jede einzelne Seite allerdings widmet sich speziellen Themengebieten, bietet interaktive Dienste wie Online-Shopping oder lädt zum Kommunizieren ein. Problematisch daran ist für den Prozess Web Log Mining, dass es keinen einheitlichen Standard für die Präsentation von Informationen gibt, sieht man einmal vom Basisformat html ab, in dem html-Dokumente, also die einzelnen Seiten von Websites verfasst sind.

Da sowohl Technologie als auch Inhalt und Betreiberinteressen – zum Beispiel kommerzielle oder private Anwender – bei Websites höchst unterschiedlich sind, ist auch Web Log Mining

kein standardisierter Prozess. Jede Web Log Mining Anwendung zielt auf unterschiedliche Fragestellungen ab und muss auf einer anderen Datenbasis operieren. Analysen sind daher kontext- und anwendungsspezifisch. Eine gute Kenntnis über das zu betrachtende System muss bei den Anwendern von Data Mining Verfahren vorhanden sein und wird als *Domänenwissen* bezeichnet.

> *Data Mining und hier im Speziellen Web Log Mining-Prozesse sind nicht immer gleich, sondern anwendungs- und fragenabhängig*

Nachdem im Folgenden intensiver auf die *Vorverarbeitung* (weiterhin auch mit der englischen Übersetzung *Preprocessing* bezeichnet) der Serverdaten eingegangen wird, werden anschließend konkrete Fragestellungen an Web Log Mining betrachtet und Interpretationen der Ergebnisse angeboten.

4.1.1 Informationssites vs. Online-Shops

Anzumerken ist an dieser Stelle ein Unterschied zwischen zwei Grundtypen von Websites. Hier seien Websites mit reinem Informationscharakter und Online-Shops als häufige Vertreter gewählt, die in ihrer Struktur aber deutliche Unterschiede haben und dementsprechend anders auszuwerten sind.

Die Betreiberinteressen bei reinen Informationssites zielen entweder gar nicht oder zumindest weniger auf das Verkaufen ab als bei Online-Shops, stattdessen wird eine hohe Zufriedenheit von Besuchern in der Suche nach Informationen gewünscht, verkörpert in kurzen Klick-Folgen, um über möglichst kurze Wege zu den gesuchten Informationen zu gelangen. Bei Online-Shops sind die Eigenschaften von Kunden von starkem Interesse und damit interessiert die Frage, wieviel Aufwand für einen konkreten Besucher der Site betrieben werden sollte, um ihn zum Kaufen zu animieren. Möglicherweise kann durch eine Klassifikation des Benutzers sein Kaufinteresse frühzeitig erkannt und gesteigert werden.

Neben den unterschiedlichen Fragestellungen seitens des Sitebetreibers unterscheiden sich Informationssites und Online-Shops häufig auch durch einen unterschiedlichen Aufbau. Während Informationen häufig in Textform vorliegen und durch Hyperlinks intensiv verknüpft sind, eventuell Verzeichnisse eine Übersicht bieten, basieren Online-Shops in der Regel auf Abteilungen und Produktkategorien, durch die der Benutzer navigiert und letztlich auf die Beschreibung zu einzelnen Produkten gelangt.

Unterschiede zwischen Internetsites mit verschiedenen Ausrichtungen in Zielgruppe, Informationsangebot, Interaktionsmöglichkeiten und Betreibermodell als auch andere technische Ausgangsbedingungen sind also gegeben und beim Web Log Mining Ausgangspunkte für unterschiedliche Fragestellungen.

4.2 Datenaufbereitung

Wie schon einige Abschnitte weiter oben beschrieben ist die Datenaufbereitung ein aufwändiger Schritt im Prozess des Web Log Mining. Eine Ursache dafür ist, dass Tools, die für allgemeine Aufgaben des Data Mining konzipiert wurden, Datenbanken als Datenquelle bevorzugen. Die Logdaten von Webservern werden jedoch in der Regel in Textdateien abgelegt (deren Format in Abschnitt 3.2 beschrieben wurde). Eine Umwandlung von Dateiform in Datenbanktabellen muss also erfolgen.

Zudem enthalten die Logdateien viel an redundanter Information, die vor der Anwendung von Data Mining Verfahren herausgefiltert wird. Hierzu zählen in erster Linie die Grafiken, die in html-Dokumente eingebettet sind. Aber auch Navigationselemente, die von Webdesignern aus technischen Gründen eingebaut wurden und keine Information enthalten, können entfernt werden.

Andere Elemente stören in der Analyse ebenfalls, weil sie falsche Interpretationen ergeben. Beispielsweise greifen Suchmaschinen in regelmäßigen Abständen auf Websites zu, um deren Inhalt zu indizieren. Dabei handelt es sich allerdings nicht um die zu untersuchenden

und von Benutzern angestoßenen Zugriffe, sondern um maschinelle, deren Verhalten weniger von Interesse ist. Andere Einträge in der Logdaten sind Zugriffe auf nicht existente Seiten, deren Betrachtung nur für die Reparatur von Websites für Webmaster von Interesse ist und nicht Web Log Mining Verfahren unterzogen werden muss. Auch sie können folglich in der Vorverarbeitung der Daten herausgefiltert werden.

Neben der *Reinigung der Daten* und dem *Entfernen von Redundanz* ist auch die *Transaktionsableitung* aus den Protokolldaten von großer Bedeutung, um einzelne Zugriffe entsprechenden Benutzersitzungen zuordnen und Aussagen über das Suchverhalten machen zu können. Zur Bewältigung dieser Aufgabe werden die einzelnen Zugriffe miteinander verglichen und wenn ihre Felder eine hohe Ähnlichkeit aufweisen einer Session zugeordnet. Als Felder für Gleichheit kommen *remotehost*, *date*, *requeststring*, *referrer* und *user_agent* in Frage (siehe hierzu Abschnitt 3.2). Bei Architekturen, in denen jeder Zugriff schon bei der Auslieferung an den anfordernden Rechner einer Session zugewiesen ist, fällt die *Transaktionsableitung* natürlich entsprechend einfacher aus oder sogar komplett weg.

Verbleibt noch die *Aggregation* als großem Aufgabenblock des Preprocessing. Hier müssen Serverlogdaten, Benutzerdatenbank und weitere Datenquellen miteinander verknüpft werden, falls dies nicht bereits schon während des Betriebs der Fall ist. Beispielsweise können bestimmte interessante Felder der Benutzerdatenbank wie Alter, Geschlecht, Einkommenslage und Anzahl der Sitebesuche für die Analyse ausgewählt werden. Zudem werden diesen Benutzerdaten dann die jeweiligen Transaktionen – hier also die Seitenzugriffe der Logdatei – zugeordnet. Alternativ können jedem Logdatensatz die zugehörigen Benutzerdaten angehängt werden, falls eine mehr seiten- und weniger benutzerorientierte Analyse gewünscht wird.

4.2.1 Vorstellung der Schritte

Die im vorherigen Abschnitt aufgeführten Aufgaben müssen im Preprocessing vor der Anwendung von Data Mining Verfahren angewendet werden. Grob lassen sich dabei folgende Schritte ausmachen, die von Anwendungsfall zu Anwendungsfall auch voneinander abweichen können.

- Entfernen von Redundanz und Bereinigung der Daten
- Ableiten von Sessions (Transaktionsableitung)
- Verknüpfung mit weiteren Datenquellen
- Codierung der Daten für Data Mining Verfahren

Die beiden ersten Themen werden im Folgenden näher erläutert. Der dritte Punkt wurde in den vorangegangenen Abschnitten bereits intensiver behandelt. Beim vierten Punkt wurde auf eine Erläuterung verzichtet, da es sich hier um einen sehr technischen Vorgang handelt, der stark abhängig von der verwendeten Miningsoftware ist. Generell kann gesagt werden, dass die Daten von der Software gelesen werden müssen und entsprechende Konvertierungsschritte nötig sind, beispielsweise die Speicherung in einer SQL-Datenbank.

Für diese Arbeit wurden einige konkretere Untersuchungen an Logdateien einer Informationssite vorgenommen, die dem Autor freundlicher Weise zur Verfügung gestellt wurden. Dabei handelt es sich um ein Informationssystem, das auf einem Branchenbuch mit einer großen Anzahl von Beschreibungen zu Geschäften und Unterhaltungs- sowie Gastronomieeinrichtungen basiert. Zusätzlich werden in beschränktem Maße Nachrichten, Jobangebote und Veranstaltungstipps angeboten. Auch ein Online-Shop ist angebunden, bietet aber nur relativ wenige Inhalte. Das Angebot ist zu finden unter http://www.frankfurt-online.de/.

Abbildung 11 - Screenshot der Startseite von frankfurt-online (Stand Juni 2001)

In diesem Rahmen wurden einige Schwierigkeiten beim Preprocessing und der Anwendung von Data Mining Software beobachtet, die entweder mit Hilfe von kleinen Skripten gelöst oder zumindest Lösungswege angedacht wurden. In der Summe waren die Schwierigkeiten aufgrund der Architektur dieser Seite und der daraus resultierenden Logdaten aber so groß, dass zunächst eine Optimierung der Site hätte stattfinden müssen, um sinnvolles Web Log Mining zu betreiben. Dennoch und vielleicht gerade deswegen sind die gesammelten Erfahrungen sinnvoll und fließen in die folgenden Kapitel mit ein.

Sehen wir uns die Aufgaben an...

Entfernen von Redundanz und Bereinigung der Daten

Als erste Aufgaben des Preprocessing sind das Entfernen von Redundanz und die Bereinigung der Daten anzugehen. Dadurch schrumpfen auch die zu verarbeitenden Datenmengen, was als positiver Nebeneffekt in diesem Schritt zu sehen ist. Da die Logdaten in aller Regel in Form von Dateien vorliegen, bietet sich der Einsatz dateibasierter Kommandozeilenutilities wie „grep“ zum zeilenweisen Suchen von regulären Ausdrücken an.

Im World Wide Web im allgemeinen und bei html-Dokumenten im speziellen werden prinzipiell alle Elemente, die im Webbrowser auf einer Seite zusammenangezeigt werden, als einzelne Dateien übertragen. Die Anforderung einer Seite einer Website zieht in der Regel das Anfordern der darin enthaltenen Grafiken, Java-Applets, Stylesheets, usw. mit sich. Jede dieser Anforderungen wird in der Logdatei protokolliert, dabei handelt es sich um redundante Informationen und nur die Anforderung der eigentlichen Seite (dem html-Dokument) ist für Web Log Mining von Interesse.

Gehen wir von einer Logdatei mit dem Namen `access.log` als Eingabedatenquelle und einer Ausgabedatei mit den verarbeiteten Daten von `access_ready.log` aus. Zunächst werden alle Grafikdateien und Stylesheets ausgefiltert, die durch die Dateiendungen `.jpg`, `.gif`, `.png` und `.css` gekennzeichnet sind.

```
grep -v "GET .*\.jpg" access.log > ac1.log
```

```
grep -v "GET .*\.gif" ac1.log  > ac2.log
grep -v "GET .*\.png" ac2.log  > ac3.log
grep -v "GET .*\.css" ac3.log  > ac4.log
```

Anfragen mit dem HEAD-Befehl, die nur den Header der http-Datenübertragung ohne den eigentlichen Inhalt darstellen, können ebenfalls ausgefiltert werden.

```
grep -v "\"HEAD /" ac4.log  > ac5.log
```

Weitere überflüssige Datensätze (zur Erinnerung: jeder Datensatz = eine Zeile) stellen Anfragen von Suchmaschinen dar, die man an einem alternativen *Browserstring* oder einem bezeichnenden *Remotehost* erkennt (siehe 3.2 für das Logdateiformat). Sie sind nutzlos für die hier vorgestellten Anwendungen von Web Log Mining, da das Verhalten von Benutzern und nicht das von Suchmaschinen untersucht werden soll. Durch einige grep-Aufrufe mit entsprechenden regulären Ausdrücken entledigt man sich auch dieser Zeilen. Die hier angegebenen Suchmaschinen sind nur eine Auswahl. In der Praxis werden Websites von recht vielen Systemen indiziert.

```
grep -v "^marvin\.northernlight\.com" ac5.log  > ac6.log
grep -v "^rap\.fireball\.de" ac6.log  > ac7.log
grep -v "^j.*\.inktomi\.com" ac7.log  > ac8.log
grep -v "^crawl.*\.googlebot\.com" ac8.log  > ac9.log
grep -v "^crawl.*-public\.alexa\.com" ac9.log  > ac10.log
grep -v "^si.*\.inktomi\.com" ac10.log  > ac11.log
grep -v "^europa\.excite\.com" ac11.log  > ac12.log
grep -v "^crawler2\.bos2\.fast-search\.net" ac12.log  > ac13.log
```

Die bisherigen Filter können im Prinzip auf alle Websites angewendet werden. Spezieller wird es, wenn bestimmte Seiten nicht in die Analyse einfließen sollen. So sind bei frankfurt-online Administrationsseiten in den Logdateien, die aber nur vom Betreiber angefordert werden und für die Betrachtung des Benutzerverhaltens nur von eingeschränktem Interesse sind. Ähnlich verhält es sich mit der periodisch angeforderten Seite „calendarMessages.php“, die ebenfalls gefiltert werden kann. Auch stören Navigationsseiten, die besonders beim Einsatz von Frames auftreten, da sie keine Inhalte enthalten. So werden auch Dateien wie leer.html ausgefiltert.

```
grep -v "GET /administration/.*" ac13.log > ac14.log
grep -v "GET /calendarMessages.php" ac14.log > ac15.log
grep -v "GET /leer.html" ac15.log > access_ready.log
```

An dieser Stelle können noch nicht existente oder fehlerbehaftete Seiten ausgefiltert werden, die durch den Rückgabewert des Webservers gekennzeichnet sind, worauf hier verzichtet wird. Durch die vorangegangenen Schritte sind die Rohdaten von viel Redundanz befreit und störende Datensätze ausgefiltert worden. Die nächste Aufgabe wird darin bestehen, sie in ein für Data Mining Software verständliches Format zu überführen.

Ableiten von Sessions (Transaktionsableitung)

Aufgrund der Zustandslosigkeit des im World Wide Web verwendeten http-Protokolls, mit dem die Seiten zwischen Webbrowser und Webserver transportiert werden, besteht zwischen den einzelnen Anforderungen von Elementen kein Zusammenhang. Wurden Sessionids verwendet und über URLs oder Cookies transportiert, wurde diese Zustandslosigkeit aufgehoben und die einzelnen Zeilen der Logdatei können jeweils einer Session (Session = Benutzersitzung) zugeordnet werden.

Im Fall, dass es solche Erweiterungen nicht gibt, sind die Transaktionen (hier die Sessions) aus den reinen Logdaten abzuleiten. Prinzipiell können Datensätze der gleichen Session

zugeordnet werden, wenn sie in bestimmten Merkmalen (*remotehost*, *user_agent*) völlig und in bestimmten (*Datum*) annähernd übereinstimmen. Eine Session ist im Durchschnitt 25 Minuten lang, d. h. ebenfalls nach 25 Minuten Inaktivität seitens eines Benutzers endet die aktuelle Session für ihn. Weitere Details zu dieser Form der Transaktionsableitung finden sich unter 3.2 und 3.3. Hier geht es um die praktische Ableitung der Sessions aus den gegebenen Protokolldaten.

Der Autor hat sich zu Testzwecken für die Speicherung der Protokolldaten für das Datenbanksystem IBM DB2 entschieden, das problemlos mit IBM Intelligent for Miner als Data Mining-Software zusammenarbeitet, die später zum Einsatz kommt. Die Tabellenstruktur und der komplette Quellcode des in Visual Basis for Applications (VBA) geschriebenen Programms zur Transaktionsableitung finden sich im Anhang. Das Datenbanksystem anstelle von Textdateien wurde gewählt, weil Zugriffe auf Datenbanken im allgemeinen schneller vonstatten gehen als auf Textdateien und Geschwindigkeit spielt sowohl beim Preprocessing als auch bei der eigentlichen Analyse eine entscheidende Rolle, um Ergebnisse in akzeptabler Zeit zu erreichen.

Die Rohdaten werden bei der *Transaktionsableitung* aus den um Redundanz geminderten und bereinigten Logdateien gelesen und in eine Tabelle der Datenbank geschrieben, wobei gleichzeitig eine Aufteilung in Sessions vorgenommen wird (siehe Anhang 6.3 für Informationen über ein mögliches Tabellenschema). Dabei wird jeder Datensatz aus den Rohdaten gelesen, mit den bestehenden Einträgen der Tabelle verglichen und wenn Übereinstimmung mit einem Datensatz in den Feldern *remotehost*, *user_agent* und *date* besteht, mit der gleichen Sessionid wie dieser Datensatz in die Datenbank geschrieben.

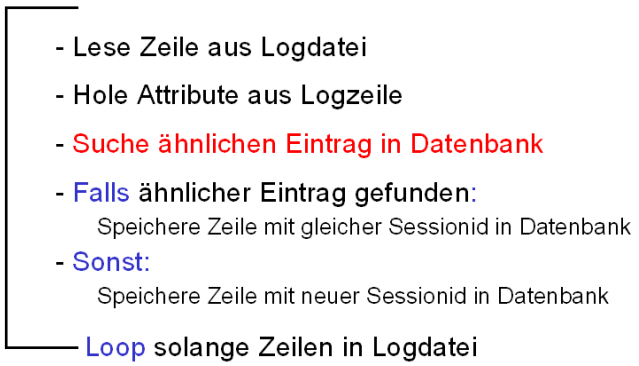

Abbildung 12 - Algorithmus für Transaktionsableitung bei Logdateien

Das Ergebnis dieses Vorganges, der von dem im Anhang abgedruckten Programm (siehe 6.2) durchgeführt wird, ist eine Tabelle, deren Datensätze die einzelnen Logdatensätze sind – angereichert um ein Feld für die Sessionid. Alle Datensätze mit der gleichen Sessionid bilden so eine Session, womit die Analysemethoden der Data Mining Software operieren können.

An dieser Stelle wird noch ein weiteres Problem gelöst, mit dem die Analyseverfahren des Data Mining zu kämpfen haben. Dabei handelt es sich um Folgendes: Bei großen Websites werden sehr viele verschiedene Informationen angeboten, die entweder in einzelnen, statischen Dateien oder dynamisch über Skripte verfügbar sind. Der Zugriff auf eine Informationseinheit (eine Seite) erfolgt mittels eine URL, die eine Seite eindeutig adressiert. Verfügt eine Website über sehr viele Seiten, dann tauchen in den Logdaten nur wenige Einträge auf, die den gleichen Seitenaufruf protokolliert haben. Der Inhalt einer einzigen Seite ist für Web Log Mining jedoch nicht von vorrangigem Interesse. Ein Zusammenfassung mehrerer Seiten zu einem semantischen Block erscheint sinnvoll und kann im Preprocessing vorgenommen werden.

Für die Website frankfurt-online kann eine solche Gliederung in die vier großen Bereiche Information, Shopping, Fun/Unterhaltung und Kommunikation erfolgen. Abbildung 13 zeigt

ein solches schematisches Mapping von einzelnen Dateien in die vier genannten Themenbereiche.

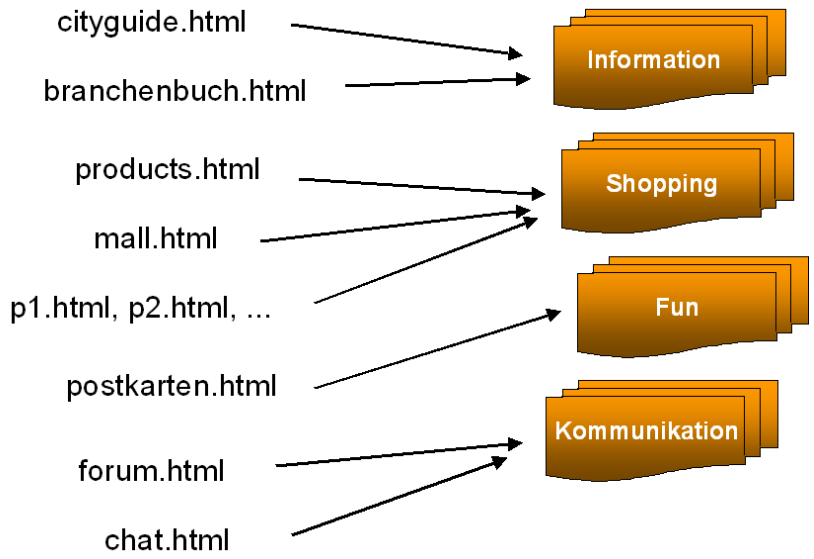

Abbildung 13 - Mapping und Gruppierung von Seiten

Vorteil an einem solchen Mapping ist eine geringere Komplexität und eine deutliche höhere Aussagekraft der mit Data Mining gefundenen Regeln, da bei der Verwendung vieler einzelner, unterschiedlicher Seiten nur Regeln gefunden werden, die eine gewisse Bedeutung für sehr kleine Teilbereiche aber nur eine geringe Bedeutung für die gesamte Website haben.

4.3 Datenanalyse

Nachdem die Daten in eine für die eingesetzten Data Mining Verfahren verwendbare Form konvertiert wurden, können die Methoden schließlich zum Einsatz kommen. Bisher waren die Vorgänge des Preprocessing noch recht allgemeingültig und noch einigermaßen frei von Fragestellungen, sieht man einmal von den Aufgaben der Datenanreicherung ab.

Im Folgenden werden einige konkrete Fragestellungen erläutert, wie sie vom Betreiber einer Kommunikations- und Informationssite mit angeschlossenem Online-Shop gestellt werden könnten. Damit werden sowohl Interessen bezüglich der Optimierung des Informationsangebots als auch eine Betrachtung der Benutzerschaft abgedeckt.

4.3.1 Fragestellungen und entsprechende Analyseverfahren

Bei den gestellten Fragen werden verschiedene Verfahren des Data Mining eingesetzt. Ein gewisser Querschnitt an Verfahren kommt also für das Web Log Mining zum Einsatz. Die hier betrachteten Fragen sind

- Welche Seiten werden von Besuchern zusammen besucht?
- Welche Wege nehmen Besucher, um zu Informationen zu gelangen?
- Gibt es Gruppen von Benutzern ähnlichen Verhaltens und wie sehen diese Gruppen aus?
- Wie sehen die Unterschiede zwischen Käufern und Nicht-Käufern aus?

Zu jeder Frage werden die einzusetzenden Verfahren aufgezeigt und beschrieben sowie eine erste Interpretation der Ergebnisse gegeben. Eine allgemeinere Interpretation erfolgt weiter unten im nächsten Hauptabschnitt. Aber zunächst zu den konkreten Fragen.

Welche Seiten werden von Besuchern zusammen besucht?

Besucher einer Website navigieren in einer Sitzung – gesteuert durch Hyperlinks in den Dokumenten – verschiedene Seiten an. In der Menge dieser Seiten kann man komplementäre (oder abgeschwächter als verwandt bezeichnete) Informationsangebote sehen, die für den Besucher von Interesse sind.

Der Nutzen einer Analyse dieser Abhängigkeitsbeziehungen ist, dass aufgedeckt wird, welche Häufigkeit und damit Bedeutung einzelne Paare oder Gruppen von gemeinsam besuchten Seiten haben. Sind bestimmte Übergänge von einer Seite zur anderen für den Webmaster überraschend häufig vertreten und leistet die Seitenstruktur hierfür nur geringe Unterstützung, so kann eine Veränderung der Seite dahingehend erfolgen, dass auf beiden Seiten auf die jeweils andere mit einem hervorgehobenen Hyperlink verwiesen wird und somit dem Informationsbedürfnis der Besucher entsprochen wird.

Zur Ermittlung entsprechender Regeln der Form

$$\text{A.html} \overset{0,9;0,2}{\Rightarrow} \text{B.html}$$

wird die Assoziationsanalyse verwendet. Der Algorithmus sucht dabei nach der Häufigkeit, mit der die einzelnen Assoziationen vorkommen. Für eine Regel „Konsequenz B folgt aus Prämisse A“ gibt es zwei Kennziffern, die die Häufigkeit der Regel im gesamten untersuchten Datenbestand angeben. Dabei handelt es sich um Konfidenz- und Supportfaktor.

Mit dem Supportfaktor wird angegeben, wie häufig (prozentual) die Seiten im Datenbestand gemeinsam auftauchen. Damit ist die Bedeutung einer Regel gegeben.

$$Support\left(\text{A.html,B.html}\right) = \frac{\#\text{ Sessions, die A.html und B.html enthalten}}{\#\text{ Anzahl der Sessions}}$$

Der Konfidenzfaktor einer solchen Regel „aus A folgt B“ gibt die Häufigkeit dafür an, dass B aus A folgt. Damit ist die Zuverlässigkeit der Assoziationsregel gegeben.

$$Konfidenz\left(\text{A.html} \Rightarrow \text{B.html}\right) = \frac{\#\text{ Sessions, die A.html und B.html enthalten}}{\#\text{ Sessions, die A.html enthalten}}$$

Problematisch an der Assoziationsanalyse ist, dass uninteressante oder fehlerhafte Daten nicht vom Algorithmus aussortiert werden und das Ergebnis schnell verfälschen können. Beispielsweise traten bei konkreten Anwendungen viele Regeln mit inhaltslosen Navigationsseiten auf, die an sich keinen Inhalt enthielten und der Benutzer niemals direkt angesteuert hat. Auch bergen Sites mit sehr vielen unterschiedlichen Dokumenten die Gefahr, dass der Algorithmus keine Regeln findet, weil die Vergleiche des *requeststring*-Feldes aus den Logdaten zu heterogen sind.

Hier schafft ein weitreichenderes Preprocessing mit mehr Datenreduktion und Redundanzvermeidung inkl. dem Mapping des *requeststring*-Feldes auf eine geringere Zahl von Bezeichnern und dem entsprechenden Zusammenfassen ähnlicher Seiten Abhilfe (siehe Abschnitt 3.2). Um dennoch seltener auftretende Assoziationen aus dem Ergebnis auszuschließen, können mit Mindestmaßen für Konfidenz- und Supportfaktor Filter vorgegeben werden, die die gefundenen Regeln erfüllen müssen.

Ein sehr gutes Beispiel für die Assoziationsanalyse ist die Präsentation weiterer Buchtitel bei einer Buchbeschreibung, wie sie beispielsweise bei amazon.com und anderen Buchhändlern anzutreffen ist. In diesem Fall werden gemeinsam gekaufte Bücher betrachtet und die stärksten Assoziationsregeln auf der Website als Links angezeigt (siehe „Abbildung 3 - Assoziationen von Büchern bei amazon.com“).

Für den Websitebetreiber bedeuten die Ergebnisse der Analyse, dass die Seiten mit den stärksten Support- und Konfidenzfaktoren deutlicher miteinander verlinkt werden können, um das Informationsbedürfnis des Besuchers besser zu befriedigen. Eine starke Assoziationsregel lässt ohnehin auf eine direkte Verlinkung der jeweiligen Seiten hindeuten,

was durch weitere Maßnahmen wie einen gezielteren Hinweis auf die komplementäre Seite intensiviert werden kann.

Angewendet auf Protokolldaten im Beispiel frankfurt-online lieferte die Assoziationsanalyse von IBM Intelligent Miner for Data unter anderem die beiden folgenden Regeln, die eine Übersichtsseite von Tourismusangeboten und eine Liste von Hotels sowie einen Stadtplan zueinander in Bezug setzen.

[tourismus/tour] ⇒ [tourismus/hotels] 0,0163;0,29

[tourismus/tour] ⇒ [stadtplan.vhtml] 0,0161;0,286

Mit einem Supportfaktor von aufgerundet 2% (die erste der beiden Werteangaben der Regeln) und einem Konfidenzfaktor von ungefähr 30% (die zweite Werteangabe), stellen die Seiten komplementäre Informationsangebote dar, bei denen der Betreiber der Website prüfen könnte, die Verbindung von allgemeinen Tourismusangeboten, der Hotelübersicht und dem entsprechenden Stadtplan zu intensivieren. Beispielsweise durch auffallendere Links oder die direkte Einblendung von Hotelangeboten zu Tourismusangeboten.

Welche Wege nehmen Besucher, um zu Informationen zu gelangen?

Bei der Betrachtung der zusammenbesuchten Seiten ging es darum, häufig zusammenbesuchte Informationsangebote ausfindig zu machen. Über die zeitliche Reihenfolge der Informationssuchaktivitäten wurde dabei nichts ausgesagt. Wie steht es aber um das Klickverhalten der Besucher?

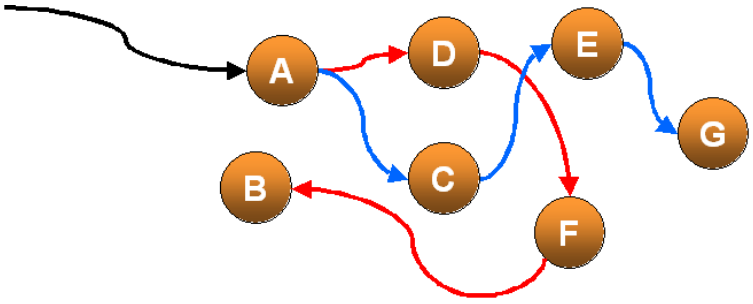

Abbildung 14 - Schematische Darstellung von Clickstreams eines Benutzers

Mit einem Clickstream wird die Abfolge von Seiten bezeichnet, die ein Benutzer durchläuft, um an das gesuchte Informationsangebot zu gelangen. Interessant ist dabei an sich jeder einzelne Clickstream eines Benutzers, der dem Webdesigner wertvolle Informationen über mögliche fehlplatzierte Hyperlinks und Zusammenhänge zwischen den dargebotenen Inhalten preisgibt und damit Optimierungsansätze für die Seitengestaltung liefert. Data Mining liefert jedoch Verfahren, um die Häufigkeit einzelner Pfade zu ermitteln und damit Regeln über das Besucherverhalten zu liefern.

Abbildung 15 - Visualisierung eines in frankfurt-online gefundenen Clickstreams

Mit der Sequenzanalyse werden – wie schon bei der Assoziationsanalyse – Regeln aus dem Datenbestand extrahiert, die eine Abfolge von Seitenaufrufen mit der entsprechenden Häufigkeit im Gesamtdatenbestand beschreiben.

$$\text{A.html} \Rightarrow \text{D.html} \Rightarrow \text{F.html} \Rightarrow \text{B.html} \quad 0{,}05$$

Wie in der obigen Regeln angegeben, werden die Seiten A.html, D.html, F.html und B.html nacheinander angesteuert und zwar mit einer Wahrscheinlichkeit von 5%.

Problematisch wie schon bei der Assoziationsanalyse ist die Vielzahl an unterschiedlichen Seiten, aus denen große Websites bestehen. Assoziations- und Sequenzalgorithmen liefern dann sehr wenige verwertbare Regeln zurück, da die meisten Regeln nur sehr geringe Support- und Konfidenzfaktoren haben. Gesucht sind aber Regeln, die allgemeine Gültigkeit für die betrachtete Website haben. Sinnvoller und nötig, um die Verwertbarkeit der gefundenen Regeln zu verbessern, ist es daher, die Seiten in eine geringe Menge von Sektionen zu unterteilen und jede einzelne Seite einer solchen Sektion zuzuordnen (siehe Abschnitt 4.2.1 und der Abbildung 13 - Mapping und Gruppierung von Seiten).

Am Beispiel frankfurt-online gibt es Diskussionsforen zu verschiedenen Themen und einen Shoppingbereich, die beide nicht optimal von den Besuchern in Anspruch genommen werden. Der Betreiber von frankfurt-online könnte beispielsweise daran interessant sein, die Anzahl der Beiträge, die Benutzer in den Diskussionsforen verfassen zu steigern, um so mehr Inhalte von den Besuchern der Site zu generieren und die Gemeinschaft lebhafter werden zu lassen. Auch der Shoppingbereich könnte deutlich häufiger angesteuert werden, um letztlich mehr Umsatz für die einzelnen Shops zu gewinnen.

Die Vielzahl der Seiten von frankfurt-online sei in die vier Kategorien Information (Branchenbuch, Nachrichten), Kommunikation (Chat, Diskussionsforen), Shopping (Shops und Produkte) und Fun (Postkarten, Spiele) aufgeteilt, wie dies in Abbildung 13 angedeutet wurde. Dabei ist anzumerken, dass die folgenden Regeln fiktiv sind und nicht unbedingt dem laufenden System entsprechen.

$$\text{Fun} \Leftrightarrow \text{Kommunikation} \Rightarrow \text{Shopping} \Rightarrow \text{Kommunikation} \quad 0{,}35$$

$$\text{Info} \Leftrightarrow \text{Kommunikation} \Rightarrow \text{Info} \quad 0{,}15$$

$$\text{Info} \Leftrightarrow \text{Fun} \Rightarrow \text{Shopping} \quad 0{,}002$$

Mit der Betrachtung der Clickstreams werden einige Wege von Benutzern aufgezeigt, die bei Diskussionsforen („Kommunikation“) und im Shoppingbereich („Shopping“) gelandet sind. Dabei handelt es sich um eine Abfolge verschiedener Seiten, die mit einer gewissen Wahrscheinlichkeit ausgestattet sind. Für den Betreiber wäre es jetzt denkbar, diese Wege näher zu betrachten, die ja mit einer angegebenen Häufigkeit bei den Diskussionsforen oder im Shoppingbereich vorbeiführen. Bei den anderen Seiten dieser Wege – wie in den Regeln oben gezeigt – wäre es beispielsweise denkbar, gesonderte Hinweise wie Texte, Buttons oder Bilder auf Diskussionsforen oder Shoppingbereich zu platzieren, um die Wahrscheinlichkeit zu steigern, dass Besucher diese Ziele ansteuern. Betrachtet man die erste der drei Regeln, führen Wege auch über Kommunikation zu Shopping. Hier wäre es denkbar, in den Diskussionsforen oder im Chat durch einen Moderator ab und an Hinweise auf Produkte zu positionieren, die Besucher in den Shoppingbereich bringen. Interessant ist auch die Häufigkeitsverteilung der drei angegeben Regeln, bei der die erste und zweite recht hohe Wahrscheinlichkeiten haben, also häufig betretenen Pfaden entsprechen während die dritte Regel im Vergleich zu den beiden ersten eher selten durchlaufen wird, also weniger Gewicht hat und daher nicht unbedingt in die Entscheidungen des Betreibers einfließen sollte.

Gibt es Gruppen von Benutzern ähnlichen Verhaltens? Wie sehen diese Gruppen aus?

Die ersten beiden Fragen haben auf das generelle Surfverhalten der Besucher einer Website abgezielt. Erweitert man den Betrachtungswinkel auf registrierte Benutzer und ihr Verhalten

auf der Site, lassen sich weitergehende Fragen über die Benutzerschaft beantworten, da die Anonymität durch das Vorhandensein von Personendaten wie Alter, Interessen und Wohnort schwindet.

Eine grundlegende Fragestellung an die Benutzerdatenbank ist daher, wie die einzelnen nicht-anonymen Besucher der Website aussehen und wie sie sich in Gruppen ähnlichen Verhaltens einteilen lassen, um die Zielgruppe besser kennenzulernen. Um diese Frage mit Hilfe von Data Mining beantworten zu können, ist eine Datenbank von Benutzerdatensätzen nötig. Damit lässt sich die Besucherschaft in Gruppen segmentieren. Spannender wird es noch, wenn auch die Logdaten mit einbezogen und den einzelnen Benutzeraccounts zugeordnet werden, da dann auch das Surfverhalten mit in die Gruppenbildung einfließt und beispielsweise Unterschiede zwischen „eher informierenden“ und „eher kommunizierenden“ Benutzern ausgemacht werden können. Ein solcher, angereicherter Benutzerdatensatz könnte folgende Gestalt haben

Feldname	**Exemplarischer Wert**
Besuchername	Max Mustermann
Alter	28
Geschlecht	m
Region des Wohnorts	5
Jahreseinkommen	75
Besuche	33
Käufe	6
Seitenanzeigen Info	143
Seitenanzeigen Shop	110
Seitenanzeigen Fun	5
Seitenanzeigen Komm	54

Mit diesem Benutzerdatensatz wird also ein männlicher Benutzer mittleren Einkommens beschrieben, der in Region 5 lebt und die Website insgesamt 33 Mal besucht hat und dabei Seiten aus dem in den vorangegangenen Abschnitten verwendeten Bereichen Info, Shop, Fun und Kommunikation besucht hat. Neben dieser Form der Zuordnung von Protokolldaten zu einem Benutzerdatensatz besteht auch die Alternative, jedem Datensatz der Serverlogdaten Informationen aus dem entsprechenden Benutzerdatensatz zuzuordnen. Die Sicht ist dann stärker auf Log- als auf Besucherdaten ausgerichtet und bietet Potential für Mining-Aktivitäten zur Gruppierung von Websitebereichen. Hier soll die erste Version verwendet und eine Segmentierung der Besucherdatensätze vorgenommen werden.

Zur Suche nach Gruppen von Benutzern ähnlichen Verhaltens kommt die *Clusteranalyse* zum Einsatz. Ziel dieses Verfahrens ist die Unterteilung der Benutzerdatensätze in eine Anzahl Cluster, die innerhalb der Cluster eine hohe Homogenität aufweisen, zwischen den einzelnen Clustern aber möglichst heterogen sind. Der Vergleich der Datensätze in der Analyse erfolgt mit Hilfe eines *Proximitätsmaßes* wie der *Euklidischen Norm*, wodurch den Attributsausprägungen eines Datensatzes wie Alter von 28, Besuche von 33, usw. ein Vergleichsmaß zugeordnet wird.

Die *Clusteranalyse* liefert automatisch eine Anzahl Cluster von Besuchern zurück, die bestimmte Ausprägungen in ihren Attributen gemeinsam haben. Jeder Cluster enthält eine Menge von Benutzerdatensätzen und repräsentiert so einen gewissen Prozentsatz der gesamten Nutzerschaft. Die Clusteranalyse über Benutzer- und Logdaten mit IBM Intelligent Miner for Data lieferte für einen Datenbestand beispielsweise eine Menge von Clustern, von

denen einer 11,52% aller Datensätze ausmacht (siehe unter „Data Mining in Practise – with the IBM Intelligent Miner for Data“, Quelle [7] im Literaturverzeichnis). Die Ergebnisse sind exemplarisch und würden in ähnlicher Weise auch für das bereits mehrfach aufgeführte Beispiel frankfurt-online auftreten, das eine ähnliche Struktur zu dem in der Quelle untersuchten hat.

In der folgenden Grafik ist der angesprochene Cluster mit seinen Attributsausprägungen aufgeschlüsselt. Jedes der Diagramme entspricht der Ausprägung eines Attributs, wobei der jeweils schwach gezeichnete Teil die Verteilung der Werte über dem gesamten Datenbestand entspricht und die hervorgehoben gezeichneten Elemente der Verteilung im Cluster entsprechen. In diesem Cluster sind also mit 98% Männer vertreten, gegenüber knapp 60% im Gesamtdatenbestand (Attribut *gender*). Ferner zeichnet sich der Cluster durch einen hohen Anteil von Bewohnern von Region 4 (mit 41%) aus, die älter als der Durchschnitt sind (Attribut *age*), häufiger als der Durchschnitt Käufe tätigen (Attribut *shopping*) und über ein hohes Einkommen verfügen (Attribut *revenue*). Damit lässt sich als Regel ableiten, dass ältere, männliche Besucher aus Region 4 mit gehobenem Einkommen gerne Online-Shopping betreiben.

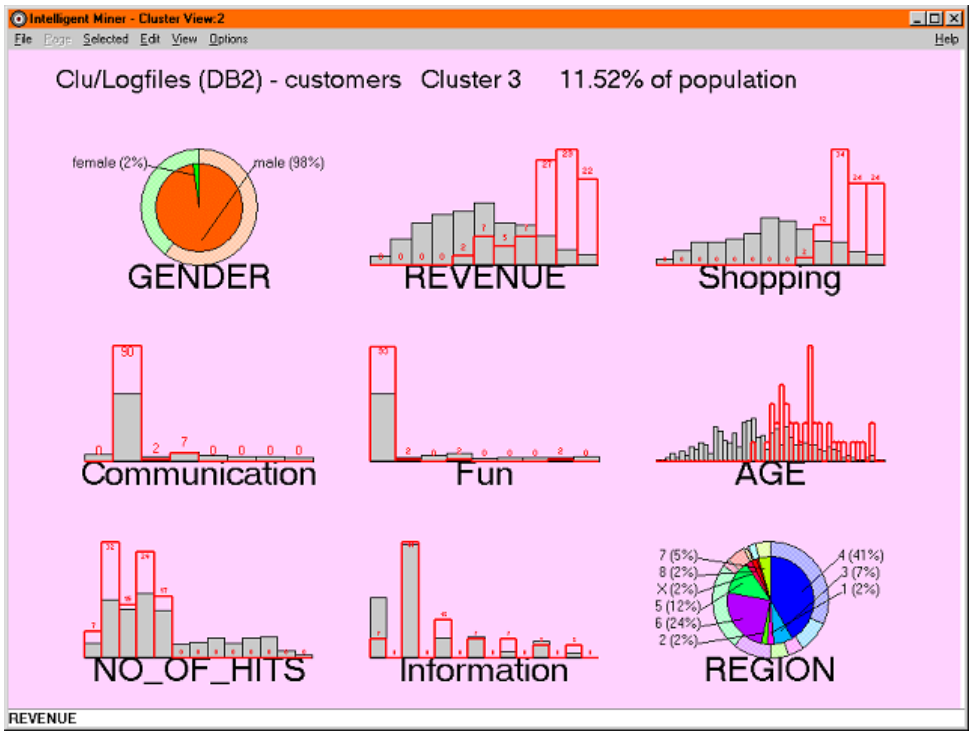

Abbildung 16 - Visualisierung eines Clusters durch IBM Intelligent Miner for Data

Dieser Cluster repräsentiert also am Online-Shopping interessierte Benutzer, während andere Cluster Personengruppen darstellen könnten, die an Kommunikation oder Unterhaltung interessiert sind. Für den Betreiber der Website wird es durch eine solche Unterteilung der Besucherschaft möglich, zielgerichteter bei der Präsentation der Informationsangebote oder Produkte vorzugehen, zielgruppengerichtete Werbung zu schalten und die einzelnen Mitglieder der Cluster durch unterschiedliche Aktionen anzusprechen.

Vorteilhaft daran ist, dass die direkte und zielgerichtetere im Gegensatz zur allgemeinen Ansprache von Besuchern ein deutlich höheres Maß an Zufriedenheit auf Seiten des Benutzers verspricht. Dabei muss es sich nicht notwendigerweise um zufriedenere Online-Shopping-Kunden drehen, die ihre Bedürfnisse nach interessanten Produkten gestillt bekommen und weitere Produkte kaufen, sondern generell um Besucher einer Website, die aufgrund ihrer Clusterzugehörigkeit abgestimmte Informationen erhalten.

Neben der Unterscheidung von Benutzern liefert eine Segmentierung von Benutzerschaft und Surfverhalten möglicherweise auch Aufschluss darüber, welche Gruppen überhaupt die betrachtete Website besuchen und ob diese Gruppen und ihr Verhalten den Vorstellungen des Betreibers entsprechen. Liegt keine Übereinstimmung vor, kann mit den Ergebnissen der Analyse eine Änderung der Geschäftspolitiken und Werbemaßnahmen erfolgen.

Wie sehen die Unterschiede zwischen Käufern und Nicht-Käufern aus?

Die Zerlegung der Gesamtheit an Benutzerdaten in einzelne Gruppen wie sie im vorangegangenen Abschnitt vorgestellt wurde liefert ein Bild vom Surfverhalten der Besucher in Abhängigkeit von ihren individuellen Ausprägungen. Dabei treten Cluster von Benutzern auf, die aufgrund ihrer Eigenschaften wie Alter, Einkommen oder Geschlecht bestimmte Bereiche der Website bevorzugt und häufig ansteuern – beispielsweise einen Online-Shopping-Bereich wie im vorherigen Abschnitt gezeigt.

Für Betreiber von Online-Shops dürfte es darauf aufbauend eine äußerst interessante Fragestellung sein, wie Nicht-Käufer zum Kauf von Produkten animiert werden. Ein Lösungsansatz dazu ist die Betrachtung, wie sich Käufer und Nicht-Käufern unterschieden. Das dafür zum Einsatz kommende Data Mining Verfahren ist das der Entscheidungsbäume.

Entscheidungsbaumverfahren generieren Klassifikationsregeln, die Zusammenhänge zwischen den Attributen der Datensätze und ihrer Gruppenzugehörigkeit beschreiben. Jeder Knoten entspricht der Abfrage eines Attributes wie größer oder kleiner einem Schwellwert, gemäß der die Klassifikation eines Datensatzes vorgenommen wird. So lässt sich der Datensatz eines Benutzers durch einen mit einem Trainingsdatenbestand aufgebauten Entscheidungsbaum „sickern“ und sobald ein Blatt des Baumes erreicht wird, liegt die Klassifikation des Datensatzes vor. Geht man davon aus, dass sich das Verhalten von Benutzern mit gleichen Merkmalen wie Alter, Einkommen und Wohnort ähnelt, so kann die mit Hilfe des Entscheidungsbaumes vorgenommene Klassifizierung als Prognose für zukünftiges Verhalten herangezogen werden, wie Kauf oder Nicht-Kauf in der Zukunft.

Beim Aufbau von Entscheidungsbäumen wird der vorliegende Datenbestand nach möglichst starken Attributsausprägungen durchsucht, die eine zuverlässige Klassifikation erlauben. Dabei tauchen besonders starke Ausprägungen weiter oben im Baum auf, schwächere weiter unten. Ein einfacher Entscheidungsbaum mit nur einem inneren Knoten ist in der folgenden Abbildung dargestellt. Kunden unter und im Alter von 38 Jahren sind Käufer, ältere Kunden nicht. Bei der Generierung des Entscheidungsbaumes wurde hier das Attribut Alter als besonders klassifizierend für das Kaufverhalten identifiziert.

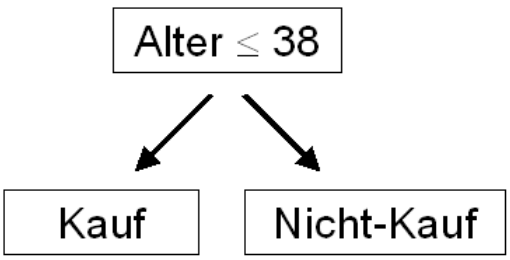

Abbildung 17 - Ein einfacher Entscheidungsbaum

Eine Prognose über das Benutzerverhalten auf einer Website ausgehend nur von einem Attribut wie dem Alter ist in der Praxis aber zu schwach, insbesondere bei sehr vielen unterschiedlichen Benutzerdatensätzen.

Worin unterscheiden sich aber jetzt Käufer und Nicht-Käufer eines Online-Shops? Wendet man ein Entscheidungsbaumverfahren auf die um Logdaten angereicherte Benutzerdatenbank an wie sie im vorangegangenen Abschnitt bereits beschrieben wurde, erhält man einen Entscheidungsbaum, der in seinen Blättern die Klassifikation Kauf oder Nicht-Kauf enthält. In den inneren Knoten werden Abfragen an die Attribute aus der

Benutzerdatenbank gestellt. Ein konkretes Beispiel wird in Abbildung 18 dargestellt. Es stammt aus Quelle [7] und könnte vom Aufbau her auch einer Untersuchung der Daten von frankfurt-online entstammen.

Auch hier gilt die Unterteilung der Seiten in Informations-, Unterhaltungs-, Kommunikations- und Shoppingangebote. Jeder Benutzerdatensatz enthält Angaben darüber, wie oft der Benutzer die entsprechenden Seiten angesteuert hat und persönliche Angaben über den Besucher, wie dies im vorangegangenen Abschnitt bereits erläutert wurde. Einen Weg durch den Entscheidungsbaum könnte dann ein Benutzerdatensatz nehmen, der einen Benutzer beschreibt, der bisher weder eine Informations- noch Unterhaltungsseite besucht hat und häufiger als 4 mal eine Kommunikationsseite angesteuert hat. Er wird dann als Käufer klassifiziert. Ein anderer Benutzer könnte Informationsseiten mindestens einmal angesteuert haben, Shoppingseiten häufiger als 8 mal, Kommunikationsangebote noch nie und Unterhaltungsseiten häufiger als 10 mal. Auch er würde als Käufer klassifiziert werden.

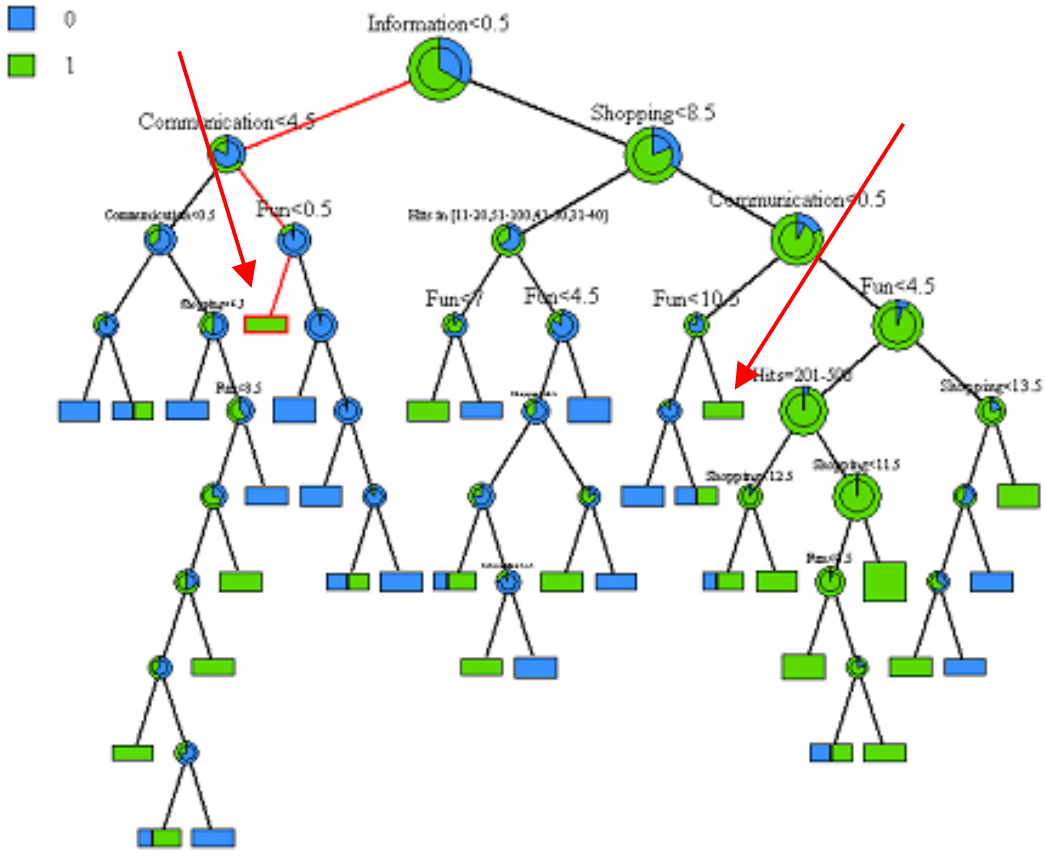

Abbildung 18 – Ein mit Intelligent Miner erstellter Entscheidungsbaum für Kaufverhalten

Für den Betreiber der Website ist mit Entscheidungsbaumverfahren als Teil des Web Log Mining also eine ungefähre Klassifikation der Besucher in Käufer und Nicht-Käufer möglich. Da das Verhalten von Benutzern mit gleichen Merkmalsausprägungen wahrscheinlich ähnlich ist, ist eine Prognose über das zukünftige Kaufverhalten von Besuchern möglich. Ein mit den Daten aus dieser Analyse gesteuertes Content-Management-System könnte dann bei Eintreten der Klassifikation „Kauf“ für den Benutzer eine entsprechende Behandlung durch angepasste Inhalte vorsehen, genauso wie mit „Nicht-Kauf“ klassifizierte Besucher durch spezialisierten Inhalt zum Kaufen animiert werden könnten. Die Unterscheidung zwischen den potentiellen Käufern und Nicht-Käufern wird mit diesem Verfahren also vom Verhalten des Sitebesuchers abhängig gemacht und bietet dem Betreiber Anhaltspunkte für Optimierungen.

4.4 Interpretation der Ergebnisse

In den ersten beiden Schritten des Web Log Mining (siehe Abbildung 10) sind Datenaufbereitung und Analyse mit den Methoden des Data Mining betrieben worden. Es stehen noch die beiden letzten Schritte aus, die sich mit der Interpretation der Analyseergebnisse und ihrer Integration in die bestehende Website befassen. An dieser Stelle soll noch einmal auf den Nutzen von Web Log Mining eingegangen werden.

4.4.1 Nutzen und Anwendungsmöglichkeiten für Websitebetreiber

Web Log Mining bietet eine tiefere Einsicht in die Unmengen von Log- und Personendaten, die von Websites und den zugrundeliegenden Webservern produziert werden. Für den Betreiber einer Website bietet Web Log Mining Optimierungsmöglichkeiten an der Struktur und den Inhalten der Website, die Betrachtung der Besucher und eine zielgruppengerechte Ausrichtung des Systems sowie allgemein entscheidungsrelevante Informationen.

Mit Assoziationsanalyse und Betrachtung der Clickstreams werden Regeln geliefert, wie die Inhalte einer Website besser miteinander verknüpft werden können, um den Besucher besser ansprechen zu können. Die Betrachtung der Benutzerdatenbank liefert konkretere Informationen zu den Besuchern der Website und bietet Potential für die direkte Ansprache von Benutzergruppen. Werden Emailnewsletter an die registrierten Benutzer verschickt, die allgemeine Informationen enthalten, werden möglicherweise nur sehr wenige Benutzer davon angesprochen. Durch die Clusterung der Benutzerdatensätze kann jedoch für jeden Cluster ein angepassten Newsletter erstellt werden, der potentiell mehr Interesse beim Leser weckt und daher eine bessere Wirkung erzielt wie dem häufigeren Besuch der Website. Auch Inhalte auf der Website können so dynamisch geschaltet werden, um den Merkmalen der einzelnen Cluster zu entsprechen. Solche dynamischen Inhalte könnten auch Werbebanner sein, die auf die im Cluster repräsentierte Zielgruppe ausgerichtet sind und so Streuverluste vermeiden helfen.

Eine weitere Anwendung des Web Log Mining ist die Überprüfung von Zielen, die hinter einer Website stehen. So kann überprüft werden, ob die angepeilte Zielgruppe tatsächlich das angestrebte Surfverhalten an den Tag legt und ob die Besucher tatsächlich die Pfade auf der Site einschlagen, die der Webdesigner vorgesehen hat. Generell kann auch untersucht werden, wie einzelne Angebote wie Informationsseiten oder Produkte angenommen werden und ob es sich bei den angetroffenen Besuchern um die anvisierte Zielgruppe handelt.

Geht man nicht mehr von Clustern aus, sondern von einzelnen Benutzerdatensätzen, so ist mit dieser personalisierten Besucheransprache eine sehr gezielte Kommunikation mit dem Anwender realisierbar, was mit Techniken außerhalb des Internet bisher kaum oder nur unter hohem Aufwand möglich war. All diese Aktionen sind auf die Bindung von Websitebesuchern ausgerichtet und bei Online-Shops auf die Animierung zum Kauf. Also ist primär von Interesse, den Kunden vor dem Kauf oder den Besucher vor dem Erlangen der Information auf das für ihn angenommen richtige Angebot zu lenken. Aber auch nach dem Verkauf sind durch Web Log Mining Informationen gegeben, die einen besseren Kundenservice ermöglichen, da der Benutzer und sein Verhalten identifiziert und klassifiziert und so seine Bedürfnisse möglicherweise besser zu verstehen und vorherzusagen sind.

4.5 Integration in das laufende System

Die mit Web Log Mining gewonnenen Informationen können gemäß der Betrachtungen im vorherigen Abschnitt, die auf den Nutzen und die Interpretation der Analyseergebnisse abzielten, in das bestehende System der Website integriert werden. Hier kommen aufgrund der unterschiedlichen Fragestellungen verschiedene Aufgaben in Betracht, die sowohl technischer als auch betriebswirtschaftlich-organisatorischer Natur sind. Wesentlich ist aber, dass sie genauso wie das Preprocessing und wesentlich stärker als die eigentliche Datenanalyse sehr stark von der betrachteten Website abhängen. Veränderungen an der Website müssen also abhängig von der Beschaffenheit des lokalen Systems erfolgen.

Interessant ist dabei, Web Log Mining als kontinuierlichen Prozess zu betrachten, der wiederholt und nach jeder Umgestaltung des Systems mit den Ergebnissen der vorangegangenen Analyse angewendet wird und damit zu einer kontinuierlichen Optimierung der Website beitragen kann. Damit schließt sich die Prozesskette der vier Schritte Preprocessing, Analyse, Interpretation und Integration wie sie zu Beginn dieses Kapitels vorgestellt wurde und Web Log Mining kann im Kreislauf betrieben werden.

4.6 Return on Investment

Bei all den Betrachtungen wurde deutlich, dass Web Log Mining kein standardisierter Prozess ist. Vielmehr müssen die jeweiligen Eigenheiten und die Struktur der zu untersuchenden Website und der angeschlossenen Datenbanken intensiv in den Prozess einbezogen werden. Web Log Mining ist daher aufwändig, da die verwendete Software kein Massenprodukt und individuelle Anpassung nötig ist. Vor allem muß ein vielfältiges Wissen bei den Betreibern der Analyse gegeben sein, das sich aus technischen und betriebswirtschaftlichen Fähigkeiten gepaart mit dem Wissen um die Besonderheiten der betrachteten Website zusammensetzt.

Diesem Aufwand steht der Nutzen gegenüber, der durch die Optimierung von Website und Geschäftsmodellen erreicht wird. Problematisch ist aber, diesen Nutzen oder auch Return on Investment zu messen. Denn möglicherweise kommt es nach einer Anpassung der Inhalte für einzelne Zielgruppen und Newslettermailingaktionen zu einer stärkeren Nutzung der Website und im Fall eines Online-Shops zu mehr Käufen. Eine exakte Zuordnung der Optimierungsmaßnahmen und dem potentiell größeren Erfolg ist aber nicht eindeutig und nur sehr bedingt in Zahlen möglich, da auch andere Einflüsse wie Mundpropaganda, begleitende Werbemaßnahmen oder Wegbrechen von Konkurrenz – also den allgemein nicht einfach zu messenden Faktoren – eine Veränderung des Benutzerverhaltens bedingen können.

Die Frage nach dem Return on Investment ist also nicht eindeutig mit Zahlen zu beantworten. Das zufriedenere Besucher bei richtiger Interpretation und Umsetzung der Web Log Mining Ergebnisse zu auf die eine oder andere Weise geartetem Mehr-Erfolg führen, ist aber anzunehmen und Web Log Mining an sich als sinnvoll einzustufen.

5 Ausblick

Da Web Log Mining eine sehr junge Anwendung von Data Mining ist und die Technik für Betreiber von Websites von großem Interesse für Optimierungsfragen und Erfolgsmessung ist, dürften sich in Zukunft weitere Entwicklungen auf diesem Gebiet abspielen. Für Betreiber größerer Websites, die ihr System nicht sich selbst überlassen wollen, wird es in Zukunft wohl unerlässlich sein, intensivere Analysen des Benutzungsverhaltens zu betreiben – falls dies nicht sogar bereits der Fall ist. Web Log Mining dürfte sich als eine dieser Analyseformen etablieren.

Der Anwender benötigt jedoch komfortable Hilfsmittel, um häufig und erfolgsversprechend Analysen durchführen zu können. Sowohl die Integration von Web Log Mining Features in vorhandene Webserver- und eCommerce-Software als auch die Weiterentwicklung von Auswertungssoftware für Logdaten um Web Log Mining Fähigkeiten werden daher interessant sein. Gleichzeitig könnten durch die vorangehende Standardisierung allgemeine Datenformate beispielsweise für Benutzerdatenbanken entstehen, was zu einer deutlichen Vereinfachung für Web Log Mining führen würde. Hier hängt viel von den gegenwärtigen Internetstrategien der großen Softwarehersteller ab.

5.1 Integration in Webserver- und eCommerce-Software

Bereits bei Datenbanksoftware ist ein Trend zu erkennen, dass Business Intelligence Features (zu denen auch Data Mining zu zählen ist) fest in die Software integriert werden. Alle drei großen Hersteller von Datenbanken – IBM, Oracle und Microsoft – bieten mittlerweile entsprechende Funktionen an, wenn auch in unterschiedlicher Ausprägung.

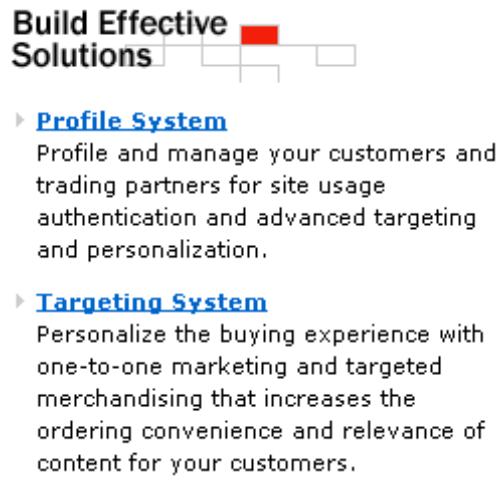

Abbildung 19 - Featurebeschreibung einer Online-Shop-Software von Microsoft

Daher ist anzunehmen, dass auch in Webserversoftware und Software für Online-Shops entsprechende Verfahren zum Web Log Mining implementiert werden. Inwieweit das aufgrund der doch recht unterschiedlichen Anforderungen beim Web Log Mining funktionieren wird, bleibt abzuwarten. Hersteller wie Microsoft werben aber bereits jetzt für Serverprodukte, die die automatische Speicherung von Benutzerverhalten und Personalisierung ermöglichen (siehe „Microsoft Commerce Server“, Quelle [8]). Dabei hat man fast das Gefühl, die Bedienung der Software werde auch für Nicht-Fachleute einfach sein, was aber aufgrund der Komplexität von Websites erstaunlich wäre und abzuwarten bleibt.

5.2 Weiterentwicklung von Auswertungssoftware

Parallel zur Integration von Web Log Mining Features in Serversoftware werden auch Analyseprogramme weiterentwickelt werden. Dazu zählen natürlich die allgemein einsetzbaren Statistikprogramme und Data Mining-Pakete aber auch speziell für die Auswertung von Logdaten entwickelte Software. Gerade in letzterer Gattung ist mittlerweile die einfache Auswertung von Weblogs perfektioniert und eine Visualisierung auf verschiedene Weise möglich. Das Zählen von Seitenabrufen (PageImpressions) und Besuchersitzungen (Visits oder Sessions) gehört so zum Standardumfang dieser speziellen Tools zur Logdatenauswertung. Die Implementierung von Web Log Mining Features dürfte als nächster logischer Schritt folgen, da die Hersteller dieser Softwareprodukte nach neuen Fähigkeiten für ihre Produkte Ausschau halten werden.

5.3 Grenzen

Die Anwendungsmöglichkeiten von Web Log Mining sind verlockend und entlocken den fortwährend generierten Protokolldaten von Webservern Nutzen. Grenzen hat die Technik jedoch dann, wenn die Daten nicht sauber vorliegen und sich einer Untersuchung verschließen oder wenn der Datenbestand aufgrund der Komplexität der Website so heterogen ist, dass es schwierig ist, allgemein verwendbare Regeln abzuleiten. Auch die beschränkte Verfügbarkeit von Rechenleistung kann bei den doch teilweise erheblich leistungszehrenden Data Mining Verfahren ein Hindernis für die allgemeine Anwendung von Web Log Mining sein.

Ein Problem ganz anderer Natur stellen Datenschutzbestimmungen dar, die die Verarbeitung und Speicherung von Personendaten einschränken oder gar verbieten. Einzelne Länder haben hier unterschiedliche Bestimmungen. In angelsächsischen Ländern beispielsweise werden Personendaten weniger sensibel behandelt als in Kontinentaleuropa, wo schon das Sammeln von Benutzerdaten problematisch sein kann. Web Log Mining findet hier also Grenzen gesetzlicher Natur.

5.4 Fazit

Trotz oder vielleicht auch gerade wegen der aufwändigen, nicht standardisierten Vorgänge beim Web Log Mining bietet die Technik interessante Möglichkeiten, Erkenntnisse aus Webserverdaten zu ziehen, die mit Methoden der rein hypothesengestützten Entdeckung von Informationen kaum möglich wären. Für den ernsthaften Betrieb von Websites dürfte Web Log Mining egal in welchem Umfang angewendet eine notwendige Technik sein, um die Vorgänge auf der Site und das Surfverhalten der Benutzer kennenzulernen und zu verstehen und damit den Erfolg eines Webprojektes erkennen zu können.

6 Anhang

6.1 Quellenverzeichnis

[1] "Data Mining und E-Commerce"

Jesus Mena, Verlag: Symposium Publishing 2000, Düsseldorf, ISBN 3-933814-12-X

[2] "Web Log Mining"

Diplomarbeit von Ernst Biesalski, bereitgestellt von Dr. Christoph Schommer (schommer@de.ibm.com)

[3] "Business Intelligence"

Martin Grothe, Peter Gentsch, Verlag: Addison-Wesley 2000, München, ISBN 3-8273-1591-3

[4] „Übersicht über Data Mining-Methoden"

Seminararbeit von Qin Sun, WS 00/01, bei PD Dr. Rüdiger Brause, Uni Frankfurt , sun@informatik.uni-frankfurt.de, zu finden unter http://www.informatik.uni-frankfurt.de/asa/SemWS00/Sun.pdf

[5] „Algorithmisches Lernen"

Skript zur Vorlesung „Algorithmisches Lernen", SS 2001, gehalten von Prof. Dr. Schnittger, Uni Frankfurt, zu finden unter http://www.thi.informatik.uni-frankfurt.de/AL/start.html

[6] „Dynamic Queries"

Seminararbeit von Martin Klossek, SS 2000, bei Prof. Dr. Krömker, Uni Frankfurt , martin@klossek3000.de, zu finden unter http://www.agc.fhg.de/uniGoethe/lehre/SS00/unterlagen/ausarbeitung_gruppe_06.pdf

[7] „Data Mining in Practise – with the IBM Intelligent Miner for Data"

Folienpräsentation von Christoph Schommer (schommer@de.ibm.com), Februar 2000

[8] „Microsoft Commerce Server"

Microsoft, Informationsangebot unter http://www.microsoft.com/commerceserver/, Stand Juni 2001

[9] „Web Mining"

Ralf Walther, in Informatik Spektrum, Band 24, Heft 1, Februar 2001, Seite 16 f., Verlag: Springer, Heidelberg

[10] „Das Ende der Logfiles"

Achim Wagenknecht, in Internet Professional, Ausgabe 6/2001, Seite 54f., Verlag: VNU Business Publications Deutschland GmbH

[11] „Dynamische Websites mit ASP"

Uwe Bünning, Verlag: Data Becker 2000, Düsseldorf, ISBN 3-8158-2019-7

6.2 Quellcode Transaktionsableitung

Hier ist der vollständige Quellcode des in Abschnitt 4.2.1 vorgestellten Skripts zur Transaktionsableitung, das in Visual Basic for Applications (VBA) verfasst wurde und auf eine per ODBC angebundene DB2-Datenbank zugreift.

```
Option Compare Database

' Einige Parameter zum Logfile Processing
Dim sRequestsTableName As String
Dim sLogFileName As String

' Die Basisdomäne des untersuchten Servers
Dim sBaseUrl As String

' Datenbankobjekte
Dim objConn As ADODB.Connection
Dim objRS As ADODB.Recordset
Dim objCommand As ADODB.Command

' Zähler für Datensatz- und Sessionidentifier und
Dim iMaxSid As Integer
Dim iMaxRid As Integer

' Initialisiert einige Variablen und die Datenbankverbindung
Sub initMethods()

    ' setzen der Werte für die globalen Variablen (okay, ue-technisch
    ' kann man das natürlich drastisch optimieren :)
    sRequestsTableName = "requests3"
    ' sLogFileName = "c:\temp\ac17.log"
    sLogFileName = "c:\temp\2001-06-08_fo_logfiles_unzipped\a15"
    sBaseUrl = LCase("http://www.frankfurt-online.de")

    ' Verbindung zur DB2 Datenbank aufbauen, in der die Requestdaten gespeichert werden
    Set objConn = CreateObject("ADODB.Connection")
    objConn.ConnectionString = "LOGDB1"
    objConn.Open "LOGDB1", "webadmin", "webibm"

    ' und ein frisches Recordset - Objekt erzeugen
    Set objRS = CreateObject("ADODB.Recordset")
    objRS.CursorLocation = adUseClient

    ' Schreiben von Datensätzen vorbereiten (benötigt wird ein Command - Objekt)
    Set objCommand = CreateObject("ADODB.Command")
    objCommand.ActiveConnection = objConn

End Sub

' Diese Methode versucht, einzelne Requests aus dem Logfile in Sessions
' zu gruppieren. Dabei werden die Daten aus der Access-Tabelle gelesen,
' verglichen und anschließend die relevanten Werte in die db2-Tabelle
' geschrieben.
Sub makeSessions()

    ' Initialisierungen ausführen
    initMethods

    ' holt die Anzahl an Sessions sowie den höchsten Wert einer Sessionid (sid)
    ' und der Requestid (rid) in der verwendeten Logdatentabelle zurück
    sSQL = "SELECT COUNT(DISTINCT sid) AS countses, MAX(sid) AS maxsid, MAX(rid) " & _
        " AS maxrid FROM " & sRequestsTableName
    objRS.Open sSQL, objConn, adOpenDynamic, adLockPessimistic

    ' Auswerten des soeben angeforderten querys
    iNumberOfSessions = ""
    iMaxSid = 0
    iMaxRid = 0
    If Not objRS.EOF Then
        iNumberOfSessions = objRS("countses")
        If objRS("maxsid") > 0 Then iMaxSid = objRS("maxsid") Else iMaxSid = 1000
        If objRS("maxrid") > 0 Then iMaxRid = objRS("maxrid") Else iMaxRid = 1000
    End If
    objRS.Close
    Debug.Print "Anzahl Sessions in Tabelle = " & iNumberOfSessions & _
          ", Nächste Sessionid = " & (iMaxSid + 1) ' vbCr &
```

```
    ' mit grep bereinigtes Logfile sLogFileName öffnen, um es in die Datenbank
    ' einzufügen und Sessions zu bilden
    Dim fLogFile As File, fs As Scripting.FileSystemObject, tsLogFile As TextStream
    Set fs = CreateObject("Scripting.FileSystemObject")
    Set fLogFile = fs.GetFile(sLogFileName)
    Set tsLogFile = fLogFile.OpenAsTextStream(ForReading, TristateFalse)

    ' Alle Zeilen durchlaufen und parsen lassen
    iNumberOfRequests = 0
    Debug.Print "Startzeit = " & Date$ & " " & Time$
    While tsLogFile.AtEndOfStream = False ' And iNumberOfRequests < 100

        ' Hole aktuelle Zeile aus Logfile und rufe parsemethode auf
        sLine = tsLogFile.ReadLine
        processLogLine sLine

        iNumberOfRequests = iNumberOfRequests + 1
        If iNumberOfRequests Mod 100 = 0 Then Debug.Print ("- " & _
          iNumberOfRequests & " Requests verarbeitet")

    Wend
    Debug.Print "Endzeit = " & Date$ & " " & Time$
    tsLogFile.Close
    Debug.Print "Einlesen des Logfiles abgeschlossen (" & iNumberOfRequests & " Requests)"
    Debug.Print "Stand des SessionId-Zählers = " & iMaxSid

End Sub

Sub processLogLine(ByVal sLine As String)

    ' wir parsen eine Logzeile im Prinzip eines Automaten für reguläre Ausdrücke
    '
    ' hier ein Beispiel für eine Logzeile
    '
    ' isdn96.hrz.tu-chemnitz.de - - [17/Apr/2001:00:00:22 +0200]
    '    "GET /cityguide/restaurants/mexikanisch/ HTTP/1.1" 200 60390
    '    "http://www.frankfurt-online.de/cityguide/restaurants/"
    '    "Mozilla/4.0 (compatible; MSIE 6.0b; Windows NT 5.0)"
    '

    ' Initialisieren einiger Variablen
    sLine = Trim(sLine)
    iLineLen = Len(sLine)
    iCurrentStrPos = 1
    iState = 0 ' Starten in Zustand 0 -> Suche des ersten Leerzeichens
    bParsing = True

    ' Der Parsingloop für eine Zeile (im Prinzip ein tokenizing)
    Dim sOutputLine As String
    While bParsing = True And iState <> -1

        Select Case iState
            ' Suche nach dem ersten space im String (damit fangen wir den remotehost ein!)
            Case 0
                iSpacePosition = InStr(iCurrentStrPos, sLine, " ", vbBinaryCompare)
                If (iSpacePosition >= iCurrentStrPos) Then
                    iSpacePosition = iSpacePosition - iCurrentStrPos
                    sRemoteHost = Mid(sLine, iCurrentStrPos, iSpacePosition)
                    iCurrentStrPos = iCurrentStrPos + iSpacePosition + 1
                    iState = 1

                    sOutputLine = sOutputLine & "'" & sRemoteHost & "', "
                Else
                    iState = -1
                End If

            ' Suche nach dem zweiten space im String (damit bekommen wir das reservierte Feld)
            Case 1
                iSpacePosition = InStr(iCurrentStrPos, sLine, " ", vbBinaryCompare)
                If (iSpacePosition >= iCurrentStrPos) Then
                    iSpacePosition = iSpacePosition - iCurrentStrPos
                    sReserved = Mid(sLine, iCurrentStrPos, iSpacePosition)
                    iCurrentStrPos = iCurrentStrPos + iSpacePosition + 1
                    iState = 2

                    sOutputLine = sOutputLine & "'" & sReserved & "', "
                Else
```

```
            iState = -1
        End If

    ' Suche nach dem dritten space im String (damit bekommen wir den Usernamen,
    ' der allerdings zumeist '-' ist)
    Case 2
        iSpacePosition = InStr(iCurrentStrPos, sLine, " ", vbBinaryCompare)
        If (iSpacePosition >= iCurrentStrPos) Then
            iSpacePosition = iSpacePosition - iCurrentStrPos
            sUserName = Mid(sLine, iCurrentStrPos, iSpacePosition)
            iCurrentStrPos = iCurrentStrPos + iSpacePosition + 1
            iState = 3

            sOutputLine = sOutputLine & "'" & sUserName & "', "
        Else
            iState = -1
        End If

    ' Suche nach dem nächsten ] im String (damit bekommen wir das Datum des requests)
    Case 3
        iSpacePosition = InStr(iCurrentStrPos, sLine, "] ", vbBinaryCompare)
        If (iSpacePosition >= iCurrentStrPos) Then
            iSpacePosition = iSpacePosition - iCurrentStrPos + 1
            sRequestDate = Mid(sLine, iCurrentStrPos, iSpacePosition)
            iCurrentStrPos = iCurrentStrPos + iSpacePosition + 1
            iState = 4

            sOutputLine = sOutputLine & "'" & sRequestDate & "', "
        Else
            iState = -1
        End If

    ' Suche nach dem nächsten " im String (damit bekommen wir den requeststring)
    Case 4
        iSpacePosition = InStr(iCurrentStrPos, sLine, Chr(34) & " ", vbBinaryCompare)
        If (iSpacePosition >= iCurrentStrPos) Then
            iSpacePosition = iSpacePosition - iCurrentStrPos + 1
            sRequestString = Mid(sLine, iCurrentStrPos, iSpacePosition)
            iCurrentStrPos = iCurrentStrPos + iSpacePosition + 1
            iState = 5

            sOutputLine = sOutputLine & "'" & sRequestString & "', "
        Else
            iState = -1
        End If

    ' Suche nach dem nächsten space im String (damit bekommen wir den returncode)
    Case 5
        iSpacePosition = InStr(iCurrentStrPos, sLine, " ", vbBinaryCompare)
        If (iSpacePosition >= iCurrentStrPos) Then
            iSpacePosition = iSpacePosition - iCurrentStrPos
            sReturnCode = Mid(sLine, iCurrentStrPos, iSpacePosition)
            iCurrentStrPos = iCurrentStrPos + iSpacePosition + 1
            iState = 6

            sOutputLine = sOutputLine & "'" & sReturnCode & "', "
        Else
            iState = -1
        End If

    ' Suche nach dem nächsten space im String (dadurch Größe des requests)
    Case 6
        iSpacePosition = InStr(iCurrentStrPos, sLine, " ", vbBinaryCompare)
        If (iSpacePosition >= iCurrentStrPos) Then
            iSpacePosition = iSpacePosition - iCurrentStrPos
            sRequestSize = Mid(sLine, iCurrentStrPos, iSpacePosition)
            iCurrentStrPos = iCurrentStrPos + iSpacePosition + 1
            iState = 7

            sOutputLine = sOutputLine & "'" & sRequestSize & "', "
        Else
            iState = -1
        End If

    ' Suche nach dem nächsten " im String (damit bekommen wir den referrer)
    Case 7
        iSpacePosition = InStr(iCurrentStrPos, sLine, Chr(34) & " ", vbBinaryCompare)
        If (iSpacePosition >= iCurrentStrPos) Then
```

```
                iSpacePosition = iSpacePosition - iCurrentStrPos + 1
                sReferrer = Mid(sLine, iCurrentStrPos, iSpacePosition)
                iCurrentStrPos = iCurrentStrPos + iSpacePosition + 1
                iState = 8

                sOutputLine = sOutputLine & "'" & sReferrer & "', "
            Else
                iState = -1
            End If

        ' Suche nach dem nächsten " im String (damit bekommen wir den browserstring)
        Case 8
            iSpacePosition = InStr(iCurrentStrPos, sLine, Chr(34), vbBinaryCompare)
            If (iSpacePosition >= iCurrentStrPos) Then
                iSpacePosition = iSpacePosition - iCurrentStrPos
                sBrowser = Mid(sLine, iCurrentStrPos)
                iCurrentStrPos = iCurrentStrPos + iSpacePosition + 1
                iState = 9

                sOutputLine = sOutputLine & "'" & sBrowser & "', "
            Else
                iState = -1
            End If

        Case 9
            iState = -1

        ' Case 1 To 5    ' Zahl von 1 bis 5.
        ' Case 6, 7, 8    ' Zahl von 6 bis 8.
        ' Case Is 9 To 10    ' Zahl ist 9 oder 10.
        ' Case Else    ' Andere Werte.
    End Select

Wend
' Debug.Print sOutputLine

' Jetzt liegen die einzelnen token vor, von denen manche noch in eine
' reinere Form überführt werden müssen (das Datum, der Requeststring,
' der referrer und auch der browserstring)

' 1. Datumsfeld konvertieren
sMonthRaw = Mid(sRequestDate, 5, 3)
sMonth = "01"
Select Case sMonthRaw
    Case "Jan"
        sMonth = "01"
    Case "Feb"
        sMonth = "02"
    Case "Mar"
        sMonth = "03"
    Case "Apr"
        sMonth = "04"
    Case "May"
        sMonth = "05"
    Case "Jun"
        sMonth = "06"
    Case "Jul"
        sMonth = "07"
    Case "Aug"
        sMonth = "08"
    Case "Sep"
        sMonth = "09"
    Case "Oct"
        sMonth = "10"
    Case "Nov"
        sMonth = "11"
    Case "Dec"
        sMonth = "12"
End Select
sNowDate = Mid(sRequestDate, 9, 4) & "-" & sMonth & "-" & Mid(sRequestDate, 2, 2)
sNowTime = Mid(sRequestDate, 14, 8)

' Vergleichsdatum für Query berechnen (-25 Minuten!)
dateCompare = DateAdd("n", -25#, DateValue(sNowDate) + TimeValue(sNowTime))

sCmpDay = Day(dateCompare)
If (Len(sCmpDay) < 2) Then sCmpDay = "0" & sCmpDay
sCmpMonth = Month(dateCompare)
```

```
If (Len(sCmpMonth) < 2) Then sCmpMonth = "0" & sCmpMonth
sCmpDate = Year(dateCompare) & "-" & sCmpMonth & "-" & sCmpDay

sCmpHour = Hour(dateCompare)
If (Len(sCmpHour) < 2) Then sCmpHour = "0" & sCmpHour
sCmpMinute = Minute(dateCompare)
If (Len(sCmpMinute) < 2) Then sCmpMinute = "0" & sCmpMinute
sCmpSecond = Second(dateCompare)
If (Len(sCmpSecond) < 2) Then sCmpSecond = "0" & sCmpSecond
sCmpTime = sCmpHour & ":" & sCmpMinute & ":" & sCmpSecond

' die aktuelle Systemzeit in einem länderunabhängigen Format (!) holen
'sNowTime = Time$
'sNowDate = Date$
'sNowDate = Mid(sNowDate, 7, 4) & "-" & Mid(sNowDate, 1, 2) & "-" & Mid(sNowDate, 4, 2)

' Nehmen wir uns den Requeststring vor, der ein wenig gesäubert werden muß...
iGetPos = InStr(1, sRequestString, "GET ", vbBinaryCompare)
sUrl1 = sRequestString
If iGetPos > 0 Then
    sUrl1 = Mid(sRequestString, iGetPos + 4)
Else
    iPostPos = InStr(1, sRequestString, "POST ", vbBinaryCompare)
    If iPostPos > 0 Then
        sUrl1 = Mid(sRequestString, iPostPos + 5)
    End If
End If
iHttpPos = InStrRev(sUrl1, "HTTP/1.")
If iHttpPos > 0 Then
    sUrl = Left(sUrl1, iHttpPos - 1)
Else
    sUrl = sUrl1
End If
sUrl = Trim(LCase(sUrl))
If Len(sUrl) > 255 Then sUrl = Left(sUrl, 255)

' beim Referrer-String müssen das führende und schließende " entfernt werden
If Left(sReferrer, 1) = Chr(34) Then sReferrer = Mid(sReferrer, 2)
If Right(sReferrer, 1) = Chr(34) Then sReferrer = Left(sReferrer, Len(sReferrer) - 1)
If Len(sReferrer) > 255 Then sReferrer = Left(sReferrer, 255)
sReferrer = LCase(sReferrer)
If InStr(1, sReferrer, sBaseUrl) = 1 Then
    sReferrer = Mid(sReferrer, Len(sBaseUrl) + 1)
Else
    sReferrer = "ext"
End If

' bleibt noch der Browserstring, bei dem ebenfalls führende und schließende " entfernt
' werden die Länge des Varchar-Tabellen-Feldes für den Browserstring beträgt 64 Zeichen
If Left(sBrowser, 1) = Chr(34) Then sBrowser = Mid(sBrowser, 2)
If Right(sBrowser, 1) = Chr(34) Then sBrowser = Left(sBrowser, Len(sBrowser) - 1)
If Len(sBrowser) > 64 Then sBrowser = Left(sBrowser, 64)

' SQL-Suchstring für Sessionzuordnung generieren
' - am referrer kann man noch drehen ---> rooturl-abschneiden!
' - uhrzeit muß auch integriert werden!
sQuery = "SELECT DISTINCT sid FROM " & sRequestsTableName & _
    " WHERE remotehost='" & sRemoteHost & "' AND browser='" & sBrowser & "'" & _
    " AND requestdate>='" & sCmpDate & "'" '& _
    ' " AND requesttime>='" & sCmpTime & "'"
    ' hm, das geht so nicht, weil es da bei Tageswechseln Schwierigkeiten gibt.
    ' menno... wir brauchen doch ein Feld vom Typ Timestamp!
    ' " AND url LIKE %" & sReferrer & "%"
' Debug.Print sQuery

' Abfrage an Datenbank absetzen, um den Request einer Session
' zuzuordnen und Resultset auswerten
objRS.Open sQuery, objConn, adOpenDynamic, adLockPessimistic
If Not objRS.EOF Then
    If objRS("sid") > 0 Then
        sSessionId = objRS("sid")
    Else
        iMaxSid = iMaxSid + 1
        sSessionId = iMaxSid
    End If
Else
    iMaxSid = iMaxSid + 1
    sSessionId = iMaxSid
```

```
    End If
    objRS.Close

    ' SQL-String zum Einfügen des Requests in die Datenbank generieren
    iMaxRid = iMaxRid + 1
    sNewRecordSQL = "INSERT INTO " & sRequestsTableName
    sNewRecordSQL = sNewRecordSQL & " VALUES (" & iMaxRid & ", " & sSessionId & ", 0, "
    sNewRecordSQL = sNewRecordSQL & "'" & sRemoteHost & "', '" & sNowDate & "', "
    sNewRecordSQL = sNewRecordSQL & "'" & sNowTime & "', '" & sUrl & "', "
    sNewRecordSQL = sNewRecordSQL & "'" & sBrowser & "', '" & sReferrer & "', 460)"

    ' neuen Datensatz schreiben
    With objCommand
        .CommandText = sNewRecordSQL
        .CommandTimeout = 5
        .CommandType = adCmdText
        .Execute
    End With

End Sub
```

6.3 Definition der Logdatentabelle als SQL-Befehl

Die mit dem im vorherigen Abschnitt angegebenen Skript verarbeiteten Logdaten werden in einer Tabelle mit der folgenden Struktur im Datenbanksystem DB2 gespeichert. Hier der SQL-create-Befehl für das Erstellen einer entsprechenden Tabelle.

```
CREATE TABLE WEBADMIN.REQUESTS3 (
    RID INTEGER  NOT NULL ,
    SID INTEGER,
    UID INTEGER,
    REMOTEHOST VARCHAR (128),
    REQUESTDATE DATE,
    REQUESTTIME TIME,
    URL VARCHAR (255),
    BROWSER VARCHAR (128),
    REFERRER VARCHAR (255),
    DURATION INTEGER,
    PRIMARY KEY (RID)
) DATA CAPTURE NONE;
```

6.4 Abbildungsverzeichnis

6.5 Stichwortindex