TECHNICAL UNIVERSITY BERGAKADEMIE FREIBERG
TECHNISCHE UNIVERSITÄT BERGAKADEMIE FREIBERG

FACULTY OF ECONOMICS AND BUSINESS ADMINISTRATION
FAKULTÄT FÜR WIRTSCHAFTSWISSENSCHAFTEN

Carsten Felden, Heiko Bock, André Gräning,
Lana Molotowa, Jan Saat, Rebecca Schäfer,
Bernhard Schneider, Jenny Steinborn,
Jochen Voecks, Christopher Woerle

Evaluation von Algorithmen zur Textklassifikation

FREIBERG WORKING PAPERS **#10**
FREIBERGER ARBEITSPAPIERE **2006**

The Faculty of Economics and Business Administration is an institution for teaching and research at the Technische Universität Bergakademie Freiberg (Saxony). For more detailed information about research and educational activities see our homepage in the World Wide Web (WWW): http://www.wiwi.tu-freiberg.de/index.html.

Address for correspondence:

Dr. Carsten Felden
Technische Universität Bergakademie Freiberg
Fakultät für Wirtschaftswissenschaften
Lehrstuhl für Allgemeine Betriebswirtschaftslehre,
insbes. Wirtschaftsinformatik
Lessingstraße 45, D-09596 Freiberg
Tel.: ++49 / 3731 / 39 26 11
Fax: ++49 / 3731 / 39 31 17
E-mail: carsten.felden@bwl.tu-freiberg.de

ISSN 0949-9970

Coordinator: Prof. Dr. Michael Fritsch

Inhalt

Zusammenfassung

Die Informationsflut, die dem Entscheidungsträger im Internet aber auch in internen Quellen begegnet, lässt sich nur schwer bewältigen. Daher ist die verfügbare Menge der Informationen zunächst in *interessante* und *uninteressante* Informationen zu unterteilen. Dem Entscheidungsträger sind anschließend lediglich die erstgenannten zuzuführen. Dazu werden im Rahmen dieses Arbeitspapiers Algorithmen auf ihre Eignung für eine Klassifikation untersucht. Die erzielten Ergebnisse zeigen, dass die Frage nach der optimalen Kombination der Vorverarbeitungsschritte und der Algorithmen nicht allgemein gültig beantwortet werden kann. Verschiedene Kombinationen, die als viel versprechend erscheinen, müssen entsprechend der Rahmenbedingungen getestet werden. Letztlich ist für den Einzelfall die beste Kombination zu wählen.

JEL-Klassifikation: C60, C80

Schlagworte: Text Mining, Information Retrieval, Algorithmus-Evaluation.

Abstract

"Algorithm Evaluation for Text Classification"

Decision makers in enterprises cannot handle information flooding without serious problems. Due to this reason the available amount of information has to be classified into the disjoint classes of interesting and uninteresting information. Afterwards just the first ones are mentioned to the user. The implemented classification algorithm has to be benchmarked against different algorithms to enhance the classification of text documents. An evaluation environment with appropriate algorithms is developed for this purpose. Also a set of test data is provided and the implementation of text mining algorithms for classification is shown. The benchmark results will be shown in the paper.

JEL-Classification: C60, C80

Keywords: Text Mining, Information Retrieval, Evaluation of Algorithms.

1 Einleitung

Unternehmen stehen vor der Herausforderung, die für sie relevanten Informationen in immer größeren Datenbeständen zu finden. 80 bis 90 Prozent der Informationen eines Unternehmens liegen nicht in maschinell verarbeitbaren und damit strukturierten Daten vor, sondern in unstrukturierten, nicht unmittelbar maschinell verarbeitbaren Daten und somit in Dokumenten.[1] Aus dem großen Anteil der unstrukturierten Daten resultiert die Bedeutung und das erhebliche Potenzial, das durch geeigneten Umgang mit der Informationsflut zu heben ist. Textuelle Informationen sind beinahe zu jedem Thema vorhanden, allerdings vielfach in wahllos aneinander gereihten Listeneinträgen, hierarchischen Schlagwortkatalogen und Indizes. Das führt dazu, dass Recherchen häufig zeitintensiver verlaufen, als es sich ein Anwender wünscht oder unter Umständen erlauben kann. Dies widerspricht aber dem Grundgedanken modernen Informationsmanagements. Informationen sollen schnell und unkompliziert verfügbar sein. Es muss das Ziel sein, dem Entscheidungsträger sowohl unternehmensexterne als auch -interne Informationen adäquat zur Verfügung zu stellen. In diesem Kontext bietet der Einsatz von Text Mining einen betriebswirtschaftlichen Nutzen. Die vom Entscheidungsträger gesuchten Inhalte werden kontextualisiert, auf dessen Situation zugeschnitten und dementsprechend aufbereitet.

Externe Daten lassen sich über das Zukaufen von Datenströmen wie z. B. von Nachrichtendiensten oder durch die Nutzung von Internetsuchmaschinen beschaffen. Da durch solche Suchprozesse externe Informationen in heterogener Form vorliegen, sind zur Informationsnutzung geeignete Text-Mining-Systeme erforderlich. Eine weitere Schwierigkeit bei externen Quellen liegt in der teilweise unbekannten Qualität der kodierten Informationen. Im Rahmen dieses Beitrags wird unter Text Mining die maschinelle Entdeckung von Wissen in Textdokumenten verstanden, das, ausgenommen den Autoren, zuvor unbekannt war. Der Begriff Text Mining subsumiert üblicherweise die Klassifikation, das Clustering sowie das Abstracting. Im Folgenden erfolgt eine Fokussierung auf die Textklassifikation, da diese bei der Informationssuche und -zuordnung von zentraler Bedeutung ist.

[1] Vgl. Kantardzic (2003, S. 189).

In Kapitel 2 wird zunächst auf die Entwicklung der Informationseinordnung durch Textklassifikation eingegangen. Darauf aufbauend veranschaulicht Kapitel 3 die Grundlagen einer automatischen Textklassifikation. Dazu wird das Vektorraummodell (Abschnitt 3.1), das Boolesche Modell (Abschnitt 3.2) sowie das Probabilistische Modell (Abschnitt 3.3) eingeführt. Für die zu erfolgende Evaluation wird in Abschnitt 3.4 das $F_ß$-Maß von van Rijsbergen als Gütemaß eingeführt, um eine Bewertung durchführen zu können. Da eine direkte Verarbeitung von Textdokumenten auf Grund ihres unstrukturierten Charakters nicht möglich ist, existieren unterschiedliche Schritte zur Vorverarbeitung. Diese werden in Abschnitt 3.5 vorgestellt. In Kapitel 4 werden zunächst noch einmal kurz die Ergebnisse der ersten Studie präsentiert. Abschnitt 4.2 zeigt daran anschließend die Evaluation weiterer Algorithmen, um die Ergebnisse zu vervollständigen. Dieser Diskussionsbeitrag endet mit einem Fazit in Kapitel 5.

2 Informationseinordnung durch Textklassifikation

Um die Problematik der Informationseinordnung zu lösen, wurden im Literatur- und Bibliothekswesen schon früh Klassifikationsmodelle im Sinne einer Zuordnung zu einer oder mehreren bestehenden Klassen entwickelt.[2] Diese Modelle erfordern jedoch eine manuelle Kategorisierung, die einen erheblichen Aufwand mit sich bringt. Vor allem seit Anfang der 90er Jahre des letzten Jahrhunderts sind daher automatische Verfahren in den Mittelpunkt der Betrachtung geraten.

Eine Methode, die manuelle Kategorisierung zu umgehen, ist das Aufstellen von Regeln, anhand derer Texte kategorisiert werden. Werden diese Regeln von einem Anwender erarbeitet und vorgegeben, handelt es sich um ein nicht-lernendes System. Da auch diese manuelle Erstellung von Regeln sehr komplex und aufwändig sein kann, wird versucht, das Anlernen automatisch durchzuführen.[3]

Dem Katalogisieren im Bibliothekswesen entspricht das Indexieren in Dokumentenverwaltungssystemen. Den Dokumenten werden Schlagworte zugewiesen, nach denen gesucht und eine Textklassifikation ausgeführt werden kann. Auf Grund des Umfangs an Dokumenten ist

[2] Vgl. Cheeseman/Stutz (1996, S. 154).

[3] Vgl. Junker (2001, S. 3).

eine Verarbeitungsautomation notwendig. Die Indexierungssprache ist dabei entweder kontrolliert, die Begriffe stehen vorher fest, oder unkontrolliert, die Begriffe stammen direkt aus den Texten.[4]

3 Grundlagen der Klassifikation unstrukturierter Daten

Text Mining unterscheidet sich vom bekannteren Data Mining dadurch, dass die Datenaufbereitungsphase um die Merkmalsextraktion erweitert wird. Jedes Dokument besitzt spezielle Merkmale, auch Attribute genannt. Diese sind grundsätzlich die in ihm verwendeten Worte (ausgenommen Meta-Informationen), die als Terme bezeichnet werden. Dem entsprechend ist in Dokumentensammlungen die mögliche Attributzahl gewöhnlich sehr hoch. Grundsätzlich werden drei Modelle diskutiert, die im Rahmen des Text Mining eingesetzt werden können. Diese sind das Vektorraummodell, das boolesche Modell sowie das probabilistische Modell. Diese werden im Folgenden vorgestellt.

3.1 Vektorraummodell

Die übliche Darstellung von den zu klassifizierenden Dokumenten erfolgt mittels des Vektorraummodells. Ein Dokument wird dabei beschrieben, indem zu jedem Term des Gesamtvokabulars das entsprechende Gewicht gespeichert wird. Der entstehende Vektor beschreibt die Lage des Dokuments als einen Punkt im multidimensionalen Raum.[5]

Es gibt verschiedene Methoden, die Termgewichte zu bestimmen. Dabei lassen sich Verfahren unterscheiden, die einzelne Dokumente betrachten, und solche, die die Gesamtheit der Texte berücksichtigen. Auch Verfahrenskombinationen sind möglich. Eine einfache Gewichtungsart auf Textebene ist z. B. die Verwendung der absoluten Häufigkeit eines Terms in einem Text. Ein Maß auf Gesamttextebene wäre die Erscheinungshäufigkeit eines Terms in allen betrachteten Dokumenten.[6] Diese Verfahren basieren auf der Beobachtung, dass die Worthäufigkeit mit der inhaltlichen Aussage eines Textes korreliert.

[4] Vgl. Salton/McGill (1983, S. 55).

[5] Vgl. Kowalski (1999, S. 101).

[6] Vgl. Ferber (2003, S. 66 ff.).

Ein sehr häufig benutztes kombiniertes Verfahren ist die Term Frequency – Inverted Document Frequency (TF-IDF)-Gewichtung. Durch diese erhalten typische Dokumentenmerkmale ein besonderes Gewicht. Das Gewicht wird berechnet, indem die Häufigkeit des Terms in einem Dokument mit der invertierten Dokumentenhäufigkeit, also der Anzahl der Dokumente, in denen der Term auftritt, multipliziert wird.[7]

3.2 Boolesches Modell

Das Boolesche Modell wendet die bekannten logischen Operatoren an. Dabei werden und ($\wedge$), oder ($\vee$) sowie nicht ($\neg$) verwendet, um den Index eines Dokumentes d_j auf die enthaltenen Deskriptoren t_i zu prüfen, welche zuvor in einer Abfrage formuliert worden sind.[8] Hierbei stellen i und j die Laufindizes der Menge der Dokumente beziehungsweise Deskriptoren dar, welche wie folgt formal beschrieben werden können:

$$D = \{d_1,...,d_j,...,d_k\}, \quad (1)$$

$$T = \{t_1,...,t_i,...,t_k\}. \quad (2)$$

Das Boolesche Modell besitzt bezüglich der Gewichtung w_{ij} nur die binären Ausprägungen $w_{ij} \in (0, 1)$. Durch diese mengentheoretische Sichtweise werden im Ergebnis ausschließlich Dokumente mit dem Wert *wahr* repräsentiert.[9] Auf Grund der Nutzung eines invertierten Indexes werden keine aufwändigen Berechnungen vorgenommen. Diese Einfachheit macht das Verfahren sehr zeitperformant. In diesem invertierten Index sind die Deskriptoren und die Dokumente, in denen die Deskriptoren verwendet wurden, aufgelistet. Durch diese Zuordnung können die Dokumente mit diesem Deskriptor direkt ermittelt werden.[10]

Jedoch besteht keine Möglichkeit, eine Rangfolge der Ergebnismenge bezüglich des Interessantheitsgrades zu erstellen, da nur binäre Merkmalsausprägungen ermittelt werden.[11] Gege-

[7] Vgl. Ferber (2003, S. 70 f.).

[8] Vgl. Zarnekow (1999, S. 126).

[9] Vgl. Salton (1989, S. 232).

[10] Vgl. Tauritz (1996, S. 5).

[11] Vgl. Zarnekow (1999, S. 125).

benenfalls variiert die Anzahl der Ergebnisse, in Abhängigkeit von der Abfrage, stark, wodurch der Nutzer zu viele oder zu wenige Dokumente erhalten kann.[12] Wegen dieser Schwäche wird das Boolesches Modell in der Praxis kaum verwendet.

3.3 Probabilistisches Modell

Das Probabilistische Modell integriert die Beziehungen der Deskriptoren in die Bewertung und geht nicht von der Annahme der Unabhängigkeit zwischen den Deskriptoren aus.[13] Im Ergebnis werden Wahrscheinlichkeiten ermittelt, welche die Relevanz von Dokumenten für den Nutzer aufzeigen.[14] Um Aussagen über die Wahrscheinlichkeit treffen zu können, ist zumindest für eine Teilmenge der Dokumente die Relevanz zu bestimmen.[15] Dieses kann mittels *Relevance Feedback* realisiert werden. Durch die Bewertung einzelner Dokumente aus der Ergebnismenge durch den Anwender werden Deskriptoren aus diesen Dokumenten extrahiert. Zum einen kann dadurch der Suchstring bei der nächsten Anfrage um diese Deskriptoren erweitert werden. Zum anderen lässt sich diesen Deskriptoren eine höhere Priorität zuordnen, so dass Deskriptoren gemäß deren Rang in Abfragen behandelt werden.[16] Die Alternativen sind vom jeweils angewendeten Modell abhängig. Die einzelnen Methoden in diesem Abschnitt sind miteinander kombinierbar und aufeinander abzustimmen, um problembezogen die besten Ergebnisse zu erzielen.

Diese Bewertung wird anschließend genutzt, um die Ergebnisqualität einer Suchanfrage zu verbessern. Durch die Bewertungen lassen sich die charakteristischen Merkmale geeigneter und ungeeigneter Dokumente erfassen und Wahrscheinlichkeiten für deren Auftreten in bestimmten Textdokumenten bestimmen.

Jedoch werden die Deskriptoren nur mit ihren binären Ausprägungen (vorhanden beziehungsweise nicht-vorhanden) bezüglich der Gewichtung berücksichtigt.[17] Zu beachten ist

[12] Vgl. Salton (1989, S. 236).

[13] Vgl. Zarnekow (1999, S. 126).

[14] Vgl. Ferber (2003, S. 185).

[15] Vgl. Baeza-Yates/Ribeiro-Neto (1999, S. 31).

[16] Vgl. Harman (1992, S. 241).

[17] Vgl. Baeza-Yates/Ribeiro-Neto (1999, S. 34).

auch, dass zu Gunsten guter Ergebnisse viele Relevanzbewertungen von Dokumenten erforderlich sind. Hierdurch gehen jedoch individuelle Präferenzen von Anwendern verloren. Aus diesem Grund eignet sich dieses Modell nicht für die nutzerspezifische Filterung von Informationen, sondern lediglich für eine gruppenspezifische Filterung, wie sie beispielsweise im Internet häufig vorgenommen wird, wo sich Nutzer im Rahmen einer Profilierung einer bestimmten Kategorie (z. B. Kategorie Börse) zuordnen müssen und dem entsprechend mit Informationen versorgt werden.

3.4 Gütekriterien des Information Retrieval

Bedeutend für die Durchführung einer Klassifikation ist der Vergleich der Klassifikationsgüte verschiedener Algorithmen bei der Auswertung unstrukturierter Daten. Die Güte wird üblicherweise mit den Parametern Precision und Recall gemessen. Dabei gibt die Precision *p* die Genauigkeit an, mit der eine Anfrage beantwortet wird. Der Recall *r* betrachtet die Vollständigkeit der Antwort, d. h. wie viele der relevanten Dokumente gefunden wurden.[18]

Gewünscht sind in beiden Fällen möglichst hohe Werte. Die beiden Maße stehen jedoch in einem negativ korrelierten Zusammenhang. Ein allgemein gehaltenes Indexierungsvokabular wird gute Recall-Ergebnisse liefern, jedoch schlechtere Precision-Ergebnisse und vice versa. Daher bietet es sich an, die beiden Maße zu kombinieren, um einen aussagekräftigen Gesamtwert zu erhalten. Dies liefert das $F_ß$-Maß:

$$F_\beta = \frac{(\beta^2 + 1) p \cdot r}{\beta^2 p + r},$$

wobei *ß* ein Gewichtungsfaktor ist und den Einfluss von Precision und Recall steuert. Üblicherweise wird für *ß* der Wert 1 gewählt, so dass beide Maße die gleiche Bedeutung für das Gesamtmaß haben. Ein Phänomen, das zu schlechten Ergebnissen führen kann, ist das so genannte *Overfitting*. Dies entsteht, wenn sich Algorithmen zu speziell an den Trainingsdokumentenbestand anpassen. In einem solchen Fall werden zwar gute Ergebnisse auf den Trainingsdokumenten erzeugt, jedoch auf neuen Dokumenten schlechte Ergebnisse generiert.[19]

[18] Vgl. Salton (1989, S. 277 f.).

[19] Vgl. Ferber (2003, S. 129).

3.5 Preprocessing zur Reduktion zu analysierender Terme

Natürliche Sprachen zeichnen sich durch verschiedene Wortformen aus. Dies ist auf die Deklinationen bei Substantiven, das Anhängen von Suffixen wie z. B. *-es*, das Voranstellen und Abtrennen von Präfixen wie z. B. *er ging hinaus* bzw. *hinausgehen*, das Einfügen von Umlauten und das durch die Rechtschreibreform erforderliche Ersetzen von *ß* durch *ss* zurückzuführen. Da auch Texte in alter Rechtschreibung durchsucht werden, ist diese Regelung weiterhin zu beachten. Diese Vielfalt führt dazu, dass jeder dieser Terme, auch wenn er auf das gleiche Wort zurückgeht, separat zu betrachten ist.

3.5.1 Stemming

Eine Möglichkeit, dieses Problem zu reduzieren, besteht im so genannten Stemming. Darunter wird die Wortherabsetzung auf ihre grammatikalische Grundform verstanden. Diese besteht bei Substantiven aus dem Nominativ Singular und bei Verben aus dem Infinitiv.[20] Stemming reduziert die Komplexität der Dokumentbeschreibungen, da weniger Terme aufgeführt werden. Gleichzeitig wird das Ziel, bei der Suche nach Dokumenten mehr geeignete Treffer zu erzielen, erreicht, da verschiedene Wortformen auf einen Term zusammengeführt sind. Problematisch für eine zweifelsfreie Wortkategorisierung ist die Mehrdeutigkeit von Wörtern und dass einige Wörter synonym gebraucht werden können. Eine häufig unterschätzte Problematik liegt im Umgang mit Mehrwort-Begriffen begründet. Werden diese Mehrwort-Begriffe nicht erkannt und die einzelnen Bestandteile separat betrachtet, kann dies zu verfälschten Ergebnissen führen.

3.5.2 Thesauren

Sehr häufig oder auch sehr selten auftretende Wörter eignen sich nicht zur Indexierung von Dokumenten. Ein Ansatz, vor allem die niederig frequentierten Wörter für die Klassifikation nutzbar zu machen, sind Thesauren. Dabei werden mit Hilfe von Wörterbüchern geeignete allgemeinere, spezifischere oder verwandte Begriffe zur Dokumentenbeschreibung gewählt.

[20] Vgl. Ferber (2003, S. 40 f.).

Ein Thesaurus stellt für die verwendeten Terme eine Gruppierungs- oder Klassifizierungsanleitung dar. Die Klassenbezeichner können anstelle der ursprünglichen Terme oder zusätzlich gespeichert werden und sollen den Recall erhöhen. Es ist wichtig, dass Begriffe gleicher Häufigkeit zusammengefasst werden, da ansonsten Anfragen nur allgemeine Ergebnisse liefern. Durch die Zusammenfassung hat der Klassenbezeichner insgesamt eine mittlere Auftretenshäufigkeit und kann daher für die Klassenbeschreibung als geeignetes Attribut angesehen werden. Um häufig vorkommende Terme für die Klassifikation zu nutzen, werden zusammengesetzte Begriffe bestehend aus mehreren Termen gebildet. Dadurch werden sie spezifischer und erzielen bessere Ergebnisse.[21]

Unter der Prämisse, dass die Wörter über die Dokumente gleich verteilt sind, lässt sich schlussfolgern, dass sehr häufig auftretende Worte keinerlei Eignung zur Klassifizierung von Dokumenten besitzen. Dies liegt daran, dass sie in nahezu jedem Dokument auftreten, was eine Unterscheidung anhand dieser Worte in der Folge unmöglich macht. Gleichzeitig treten einige Worte so selten auf, dass ihnen keine Trennfähigkeit bescheinigt wird. Diesen Überlegungen folgend bleiben nur solche Wörter für eine Klassifikation übrig, die mit mittlerer Häufigkeit auftreten.[22] Problematisch ist, die richtigen Schwellenwerte zur Abgrenzung zu finden.

3.5.3 Termeliminierung

Auch die Eliminierung von so genannten Stoppworten verfolgt das Ziel einer Dimensionsreduzierung. Dabei handelt es sich um Begriffe, die meist häufig vorkommen, jedoch themenneutral sind.[23] Im Deutschen sind dies hauptsächlich Artikel, Pronomina und einige Adjektive. Solche Stoppworte decken ca. 40 bis 50 Prozent der Textwörter eines Dokuments ab. Die meisten Stoppwortlisten in englischer oder deutscher Sprache umfassen ungefähr 600 Terme. Die Auswirkungen einer Stoppwortliste sind häufig vergleichbar mit der Einführung einer oberen Schranke für die Häufigkeit berücksichtigter Terme, da die Stoppworte zu den am häufigsten vorkommenden Termen gehören.

[21] Vgl. Salton/McGill (1983, S. 75 ff.).

[22] Vgl. Salton (1988, S. 45 ff.).

[23] Vgl. Sebastiani (2002, S. 15).

4 Algorithmen und Evaluation zur Textklassifikation

Beim Anlernen der verschiedenen Algorithmen werden zwei generelle Verfahren unterschieden. Zum einen ist dies das überwachte Lernen. Bei diesem sind für die Trainingsmenge die klassifizierenden Attribute bekannt. Die Ergebnisse werden anhand einer Testmenge überprüft, die aus Dokumenten besteht, die nicht zum Lernen herangezogen wurden, für die aber ebenfalls die Klassifikation gegeben ist. Problematisch ist jedoch, dass für die komplette Trainings- und Testmenge die Klassifikation bekannt sein muss, womit diese Methode sehr aufwändig ist.[24] Zum anderen handelt es sich um das unüberwachte Lernen. Dabei sind die Werte der vorherzusagenden Attribute nicht im Vorhinein bekannt, so dass der Algorithmus eine Einteilung finden muss.[25] Die Ergebnisse sind zu bewerten, um eine Aussage über die Güte der Klassifizierung treffen zu können. In der ersten Evaluation wurden die folgenden, in der Literatur teilweise intensiv diskutiert Algorithmen betrachtet. Dabei handelt es sich um die Algorithmen K-Nächster-Nachbarn (K-NN) und HyperPipes, Entscheidungsbäume, Naive-Bayes, Multilayer Perceptron (MLP), AdaBoost.M1, Support Vector Machines (SVM), Rocchio, Voted Perceptron sowie Simple-Logistics.[26] In dieser zweiten Studie werden weitere Algorithmen, selten diskutierte Varianten der aufgeführten Algorithmen auf ihre Eignung zur Klassifikation untersucht.

4.1 Ergebnisse der ersten Evaluation

Im Rahmen einer Evaluation der oben aufgeführten Algorithmen für ein Marktdateninformationssystem (MAIS) im Energiehandel wurde eine eigene Dokumentensammlung aufgebaut. Das MAIS erfasst unternehmensinterne Daten sowie unternehmensexterne Internetdokumente, um Energiehändlern Informationen über Marktentwicklungen zur Verfügung zu stellen. Viele andere Autoren verwenden für einen Algorithmusvergleich den Reuters-Text-Korpus oder ähnlich stark normalisierte und standardisierte Dokumentenbestände. Da jedoch die Eignung der Algorithmen unter möglichst realitätsnahen Bedingungen getestet werden soll, wurden 990 Quellen aus dem Internet gewählt. Diese repräsentative Kollektion entspricht zum einen den Bemühungen von Unternehmen, Informationen aus dem Internet zu filtern, und ist

[24] Vgl. Ferber (2003, S. 114).

[25] Vgl. Küppers (1999, S. 55).

[26] Vgl. Sebastiani (2002, S. 20 ff.).

zum anderen vergleichbar mit den gering oder gar nicht standardisierten Quellen, wie sie den Unternehmen intern in Gestalt von Mails, Gesprächsnotizen oder sonstigen Dokumenten vorliegen. Die gefundenen Dokumente wurden manuell klassifiziert und in eine Trainings- und in eine Testmenge unterteilt. Im Anschluss an die Auswahl der Dokumente wurden diese dem zuvor beschriebenen Preprocessing unterzogen. Abschließend wurden die Resultate zu einer Gesamtwortliste kombiniert, die alle aufgetretenen Wörter beinhaltet. Die Verarbeitungsschritte der allgemeinen Datenaufbereitung führen zu nahezu einer Halbierung dieser Termzahl. Durch die speziellen Datenaufbereitungsschritte wird sie weiter reduziert. Anhand der jeweiligen Gesamtwortlisten für die Versuche wurden für die einzelnen Dokumente beschreibende Vektoren erzeugt. Auf dieser Basis erfolgte dann der Test der verschiedenen Algorithmen. Im Ergebnis führten diese Vorverarbeitungsschritte zu neun unterschiedlichen Eingabedateien, die zwischen 10.343 (vollständige Vorverarbeitung) und 33.776 (keine spezielle Vorverarbeitung) Terme enthielten.

Die Ergebnisse für das $F_ß$-Maß für die durchgeführten neun Versuche bei den verschiedenen Vorverarbeitungsschritten gibt die folgende Abbildung 1 an. Das $F_ß$-Maß liegt zwischen 0 und 1.

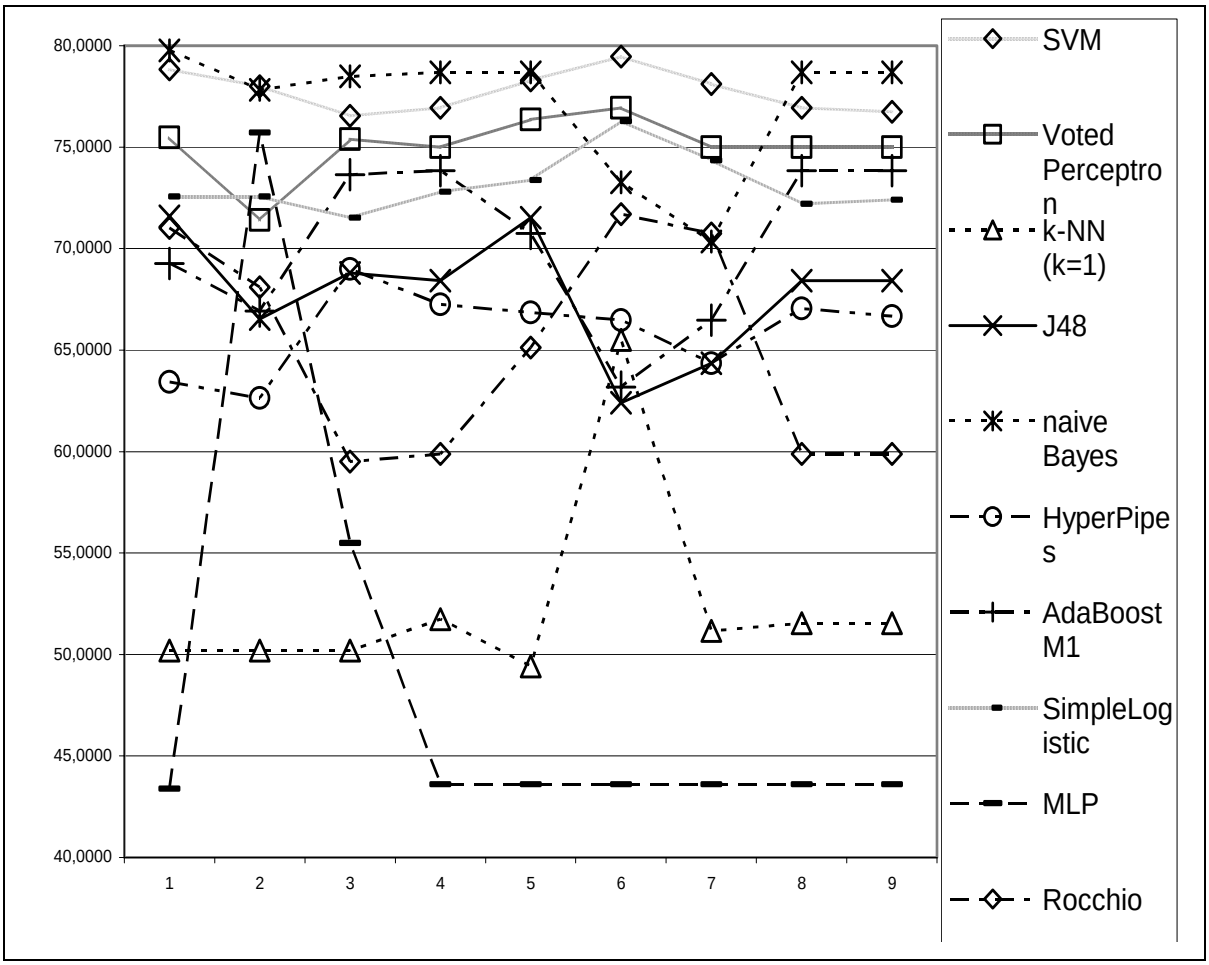

Abbildung 1: Testergebnisse anhand des $F_ß$-Maßes

Den höchsten $F_ß$-Wert erzielten Support Vector Machines (SVM) im Versuch 6 mit einem Wert von 79,73 Prozent. Insgesamt lässt sich festhalten, dass die Testergebnisse von Versuch zu Versuch Schwankungen unterliegen, die nicht bei allen Algorithmen parallel verlaufen.

4.2 Ergebnisse der zweiten Evaluation

Um eine Einheitlichkeit sowie Vergleichbarkeit der Ergebnisse zu gewährleisten, wurde derselbe Trainings- und Testdatenbestand verwendet. Der Aufbau dieses Datenbestandes wird in der Versuchsbeschreibung in Abschnitt 4.2.1 erläutert. Abschnitt 4.2.2 stellt die betrachteten Algorithmen vor, deren Evaluationsergebnisse in Abschnitt 4.2.3 aufgeführt werden.

4.2.1 Versuchsbeschreibung

In einem ersten Schritt werden Suchanfragen simuliert, indem zufällig ausgewählte branchenübliche Begriffe aus dem Data Dictionary an eine Internet-Suchmaschine übergeben werden.

Die Ergebnisse der Suche werden als Textdokumente gespeichert und in den Datenbestand aufgenommen. Anschließend werden die Dokumente manuell den Klassen *interessant* und *uninteressant* zugeordnet. Der Testdatenbestand ist über einen Zeitraum von zwei Monaten aufgebaut worden und umfasst circa 1.300 Dokumente. Die Anzahl ist zwar weitaus geringer als bei vergleichbaren Testkollektionen, jedoch erscheint dies unproblematisch, da nur eine binäre Klassifikation durchgeführt wird und bei repräsentativen Suchanfragen des MAIS jeweils nur eine geringe Anzahl von Dokumenten klassifiziert wird. Die für die Anwendung der Algorithmen notwendige Aufteilung in Trainings-, Test- und Validierungsdatenbestand erfolgt im relativen Verhältnis 50 : 20 : 30.

Problematisch ist die Wahl der Parameter der einzelnen Algorithmen. Während verfahrensspezifische Parameter frei gewählt werden können, wird die Einteilung in Trainings- und Testmenge für alle Verfahren beibehalten, um die Vergleichbarkeit zu wahren. Dies kann bedeuten, dass die Verfahren im Einzelfall bessere Ergebnisse liefern können als in dieser vergleichenden Testumgebung. Eine Teilmenge der Trainingsmenge, die Validierungsmenge, dient dazu, die Parameter der Algorithmen optimal einzustellen. Die einmal getroffene Parameterwahl ist beizubehalten, um sicherzustellen, dass die Ergebnisse reproduzierbar sind.

Erwartungsgemäß bildet das Preprocessing der Daten den größten Aufwand der Evaluation. Sämtliche Schritte des Integrationsprozesses werden manuell ausgeführt. Dadurch lassen sich alle Zwischenergebnisse überwachen und bewerten. Dies führt jedoch auch dazu, dass identische Dokumente nicht über deren Eigenschaften identifiziert und Duplikate eliminiert werden können. Es erfolgt vielmehr eine gemäß der Suchbegriffe redundante Speicherung, die das Auffinden der Textdokumente ermöglicht. An den Eigenschaften einer herkömmlichen Internetsuche mittels einer Suchmaschine soll sich nichts ändern. Die Suche selbst hat über mehrere Tage verteilt stattgefunden, um auch die Reaktion der Klassifikationsalgorithmen untersuchen zu können.

Aus den circa 1.300 Textdokumenten sind ungefähr 67.000 unterschiedliche Wörter bestimmt worden. Dies beinhaltete die folgenden Schritte, die auch grundsätzlich für die weiteren Variationen gelten: Groß- und Kleinschreibung werden angepasst und entsprechend zu einem Wert kumuliert. Bei der Konversion von HTML-Dokumenten in TXT-Dokumente werden automatisch Sonderzeichen produziert. Dies ist aus dem Grunde bedeutsam, da beispielsweise

=??? vom System als Formel interpretiert werden kann. Die zugelassenen Zeichen werden auf a-z, A-Z, äüö und ÄÜÖ beschränkt. Wörter, die Sonderzeichen enthalten, werden automatisch eliminiert. Dies führte auch automatisch zu einer Löschung von Begriffen, die als nicht englisch- oder deutschsprachig zu identifizieren sind. *‚ss'* und *‚ß'* werden gleichbehandelt. Bindestriche zwischen Wörtern werden gelöscht und die verbleibenden Begriffe jeweils als Individuen begriffen und gegebenenfalls kumuliert. Wörter, die eine Häufigkeit von 1 aufweisen, werden aus der Gesamtliste eliminiert. Diese Ursprungswortliste bildet die Basis für die weitere Bearbeitung.

Auch wenn in Diskussionen oftmals eine gegenteilige Ansicht vertreten wird, lässt sich grundsätzlich im Vornhinein nicht bestimmen, welche weiteren Schritte der Vorverarbeitung ausgeführt werden müssen. Um feststellen zu können, welche Vorverarbeitung die besten Ergebnisse garantiert, werden unterschiedliche Wortlisten erstellt. Die Variationen bestehen in den folgenden Möglichkeiten:

a) Anwendung einer manuell aufgebauten Stoppwortliste;
b) Eliminierung beispielsweise von skandinavischen Zeichen und Beschränkung auf Zeichen, die in der deutschen Sprache üblich sind;
c) Eliminierung von Begriffen, die eine geringere Wortlänge als drei Zeichen besitzen;
d) Eliminierung der Begriffe, die nur einmal in einem Text vorkommen;
e) Generierung von Wortstämmen und Zuordnung der Begriffe aus der Ursprungsliste;
f) Eliminierung der Begriffe, welche die oberen fünf Prozent der Verteilungskurve in einem Textdokument ausmachen.

Tabelle 1 veranschaulicht die Variationen.

Tabelle 1: Übersicht der Variationen zum Klassifikationsdatenbestand

Anwendung einer Stoppwortliste	Zulässigkeit beschränkt auf deutsche Zeichen	Elimination bei einer Wortlänge < 3	Elimination bei Termfrequenz #1 pro Text	Anwendung von Wortstämmen	Eliminierung der oberen 5 Prozent der Verteilungskurve	Anzahl der verbleibenden Terme	Nr.
■	■	■	■	■		10.511	1
■	■	■	■	■	■	10.343	2
			■			15.676	3
	■					31.602	4
		■				33.247	5
					■	33.392	6
					10 Prozent	32.854	7
	Sonderzeichen					33.602	8
						33.776	9

Die dunkel hinterlegten Felder zeigen an, welche Preprocessingschritte für den Aufbau des jeweiligen Klassifikationsdatenbestandes angewendet worden sind. Die Angabe *10 Prozent* besagt, dass nicht die oberen fünf Prozent, sondern die oberen zehn Prozent der Verteilungskurve beachtet werden. Analog gilt für die Angabe *Sonderzeichen*, dass lediglich die in der deutschen Sprache unüblichen Sonderzeichen aus dem Datenbestand entfernt worden sind. Die rechte Spalte zeigt, wie viele Begriffe in der Wortliste verbleiben. Die so erzeugten Listen bilden die Wortvektoren, die wiederum die Eingabedaten für die ausgewählten Klassifikationstools darstellen.

4.2.2 Eingesetzte Klassifikationsalgorithmen

Die hier eingesetzten Klassifikationsalgorithmen sind eine Implementierung des Werkzeugs *Weka* der *University of Waikato* in Neuseeland. Da im Rahmen dieses Diskussionsbeitrages die Ergebnisse der Algorithmusevaluation im Vordergrund stehen und nicht die Entwicklung einzelner Klassifikationsalgorithmen, wird an dieser Stelle grundsätzlich auf die Literatur Witten/Frank (2000) verwiesen, in der die einzelnen Algorithmen beschrieben sind. Bei eini-

gen Algorithmen wird zusätzlich an weiterführende Literatur relegiert. An dieser Stelle wird lediglich kurz aufgeführt, wie die Algorithmusklasse implementiert wurde. Dazu wird zunächst die Klassenbezeichnung aufgeführt, die auch später in der Ergebnisstabelle zu sehen ist, gefolgt von einer pragmatischen Erläuterung.

- ComplementNaiveBayes

  Hier wird eine komplementäre Klasse Naive-Bayes-Klassifikatoren genutzt. Kennzeichnend ist die Normalisierung der Termfrequenz und der invertierten Termfrequenz.[27]

- NaiveBayesMultinomial

  Klasse zur Bildung eines multinomialen Naive-Bayes-Klassifikators.[28]

- NaiveBayesUpdateable

  Klasse für Naive-Bayes-Klassifikatoren durch Anwendung von Schätzklassen. Dies ist die Update-Version des Naive Bayes. Es wird eine Standardpräzision von 0,1 für numerische Attribute verwendet, insofern keine Trainingsinstanzen aufgerufen werden.[29]

- SMO

  SMO ist die Implementierung der schrittweisen Optimierung (hinsichtlich der Fehlerminimierung) von John Platt zum Trainieren eines Support Vector Klassifikators. Die Implementierung ersetzt alle fehlenden Werte und transformiert nominale in binäre Attribute. Zusätzlich werden alle Attribute per Default normalisiert. In diesem Fall basieren alle Ausgabekoeffizienten auf den normalisierten Daten, nicht den Originaldaten.[30]

---

[27] ICML-2003 "Tackling the poor assumptions of Naive Bayes Text Classifiers", 2003.

[28] Mccallum, A.; Nigam, K.: "A Comparison of Event Models for Naive Bayes Text Classification", 1998.

[29] John, G. H.; Langley, P.: "Estimating Continuous Distributions in Bayesian Classifiers". Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. 1998, S. 338 - 345. Morgan Kaufmann, San Mateo.

[30] Platt, J.: "Fast Training of Support Vector Machines using Sequential Minimal Optimization". In: Scholkopf, B.; Burges, C.; Smola, A. (eds.): "Advances in Kernel Methods - Support Vector Learning, MIT Press, 2002 und Keerthi, S. S.; Shevade, S. K.; Bhattacharyya, C.; Murthy, K. R. K.: „Improvements to Platt's SMO Algorithm for SVM Classifier Design". Neural Computation, 13(3), 2001, S. 637 - 649.

- RBFNetwork

  Diese Klasse implementiert eine kreisförmige Netzwerkfunktion. Angewendet wird der K-Means Cluster-Algorithmus als Basisfunktion. Darauf setzt entweder eine logistische Regression zu Gunsten diskreter Problemstellungen auf, oder eine lineare Regression bei numerischen Problemstellungen.

- bayes.AODE

  AODE erreicht recht gute Klassifikationsergebnisse durch ein kleines, einem Naive-Bayes ähnliches Modell, das schwächere Unabhängigkeitsannahmen hat, als ein tatsächliches Naive-Bayes-Modell.[31]

- BayesNet

  Das Bayesche-Netzwerk wendet unterschiedliche Suchalgorithmen und Qualitätsmaßstäbe an.

- bayes.NaiveBayesSimple

  Klasse zur Bildung eines einfachen Naive-Bayes-Klassifikators. Numerische Attribute werden als Normalverteilung modelliert.[32]

- LeastMedSq

  Dies implementiert eine Medianbestimmung als quadratische lineare Regression. Die Einzelwerte werden aus zufällig bestimmten Beispieldaten ermittelt. Die quadratische Regression mit dem niedrigsten quadratischen Medianfehler wird abschließend als Modell verwendet.[33]

- LinearRegression

  Klasse zur Nutzung einer linearen Regression. Diese verwendet das Akaike-Kriterium zur Modellbestimmung.

[31] Webb, G.; Boughton, J.; Wang, Z.: “Not So Naive Bayes.” In: “Machine Learning”, 2002, und Webb, G.; Boughton, J.; Wang, Z.: “Averaged One-Dependence Estimators: Preliminary Results”. AI2002 Data Mining Workshop, Canberra, 2002.

[32] Duda, R.; Hart, P.: “Pattern Classification and Scene Analysis”. Wiley, New York, 1973.

[33] Rousseeuw, P. J.; Leroy, A. M.: “Robust regression and outlier detection”, 1987.

– Pace Regression

  Mit dieser Klasse wird ein lineares Regressionsmodell aufgebaut. Unter regulären Umständen ist die Pace Regression optimal, sofern die Koeffizienten zur Unendlichkeit tendieren. Nicht beachtet werden fehlende Werte und non-binäre nominale Attribute, insofern weniger als 20 Koeffizienten vorliegen.[34]

– Simple Linear Regression

  Hier wird ein einfaches lineares Regressionsmodell aufgebaut. Es werden diejenigen Attribute ausgewählt, welche den niedrigsten quadrierten Fehler aufweisen. Zu beachten ist, dass fehlende Werte nicht zugelassen sind und das lediglich numerische Attribute zu verarbeiten sind.

– SMO Regression

  Implementiert Alex Smola und Bernhard Scholkopf's sequentielle Optimierung (hinsichtlich der Fehlerminimierung), um ein Support-Vector-Regressions-Modell zu trainieren. Diese Implementierung ersetzt alle fehlenden Werte und transformiert nominale in binäre Attribute. Dabei werden alle Attribute per Default normalisiert, so dass auch die Ausgabe auf den normalisierten Daten basiert, nicht den Originaldaten.[35]

– Winnow

  Implementiert den Winnow sowie den Balanzierten-Winnow-Algorithmus von Littelstone. Die Klassifikation wird auf nominalen Attributen ausgeführt. Binäre Attribute werden umgewandelt.[36]

[34] Wang, Y.: “A new approach to fitting linear models in high dimensional spaces. PhD Thesis. Department of Computer Science, University of Waikato, New Zealand und Wang, Y.; Witten, I. H.: “Modeling for optimal probability prediction”. In: “Proceedings of ICML'2002“. Sydney, 2002.

[35] Smola, A. J.; Scholkopf, B.: „A Tutorial on Support Vector Regression“. In: „NeuroCOLT2 Technical Report Series - NC2-TR-1998-030.“, und Shevade, S. K.; Keerthi, S. S.; Bhattacharyya, C.; Murthy, K. R. K.: “Improvements to SMO Algorithm for SVM Regression”. In: “Technical Report CD-99-16”, Control Division Dept of Mechanical and Production Engineering, National University of Singapore.

[36] Littlestone, N.: “Learning quickly when irrelevant attributes are abound: A new linear threshold algorithm”. Machine Learning 2, 1998, S. 285 - 318 und Littlestone, N.: “Mistake bounds and logarithmic linear-threshold learning algorithms”. Technical report UCSC-CRL-89-11, University of California, Santa Cruz, 1989.

– ADTree

  Klasse zur Erstellung eines alternierenden Entscheidungsbaumes. Dabei werden in dieser Version nur Zwei-Klassen-Probleme unterstützt. Die Anzahl der Iterationen ist manuell zu bestimmen, um das best mögliche Ergebnisse zu erhalten. Die Induktion des Baumes ist durch den Anwender zu optimieren und heuristische Suchmethoden sind einzuführen, um das Trainieren zu beschleunigen.[37]

– Id3

  Klasse zur Konstruktion eines nicht-beschnittenen (unpruned) Entscheidungsbaumes, der auf dem ID3-Algorithmus basiert. Zu beachten ist, dass dieser Algorithmus lediglich mit nominalen Attributen arbeiten kann und fehlende Werte nicht gestattet sind. Hintergrund dafür ist, dass leere Blätter zu unklassifizierten Instanzen führen können.[38]

– LMT

  Klassifikator, um Bäume als logistisches Modell zu konstruieren. Dabei handelt es sich um Klassifikationsbäume, die logistische Funktionen in den Blättern beinhalten. Dieser Algorithmus kann mit binären und Multi-Klassen-Variablen, numerischen und nominalen Attributen und fehlenden Werten arbeiten.[39]

– M5P

  Dieser Baum enthält Regressionsalgorithmen in den Blättern.

– NBTree

  Klasse zur Generierung eines Entscheidungsbaumes mit einem Naive-Bayes-Klassifikator in den Blättern.[40]

---

[37] Freund, Y.; Mason, L.: “The alternating decision tree learning algorithm”. In: “Proceeding of the Sixteenth International Conference on Machine Learning”, Bled, Slovenia, 1999, S. 124 - 133.

[38] Quinlan, R.: “Induction of decision trees”. Machine Learning. Vol.1, No.1, 1986, S. 81 - 106.

[39] Landwehr, L.; Hall, M.; Frank, E.: “Logistic Model Trees” (ECML 2003), 2003.

[40] Kohavi, R.: “Scaling up the accuracy of naive-Bayes classifiers: a decision tree hybrid.” Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 1986.

– Random Forest

  Klasse zur Schaffung eines Waldes aus einer unbestimmten Anzahl an Entscheidungsbäumen.[41]

– RepTree

  In dieser Implementierung wird ein Entscheidungs-/Regressionsbaum durch die Nutzung von Informationen über die Varianz aufgebaut. Die Werte numerischer Attribute werden nur einmalig sortiert. Fehlende Werte werden durch die Aufteilung zwischen korrespondierenden Instanzen, analog zum C4.5, gehandhabt.

– User Classifier

  Hierbei erfolgt eine interaktive Klassifikation mittels der Visualisierung auf der Benutzeroberfläche. Der Anwender bekommt einen Graphen, gemäß den gewählten Attributen, dargestellt. Dazu wird analog ein Entscheidungsbaum aufgebaut und visualisiert. Der Anwender kann damit eine binäre Trennung zwischen Attributen durchführen, indem Daten eingekreist oder andere Klassifikatoren in den Entscheidungsbaum eingefügt werden.

– Conjunctive Rule

  Diese Klasse implementiert einen konjunktiven Regellerner, der mit numerischen und nominalen Bezeichnungen arbeiten kann. Eine Regel besteht aus so genannten Antizendenten, die miteinander verknüpft werden, sowie dem Resultat der angewendeten Klassifikation beziehungsweise Regression. In diesem Fall ist das Resultat die Verteilung der verfügbaren Klassen innerhalb der Datensammlung. Wird die Testinstanz durch die Verteilung nicht gedeckt, wird eine Defaultklasse verwendet, die für solche Fälle im Trainingsdatensatz enthalten ist. Zu Gunsten einer Klassifikation ist die Information eines Antizendenten der gewichtete Durchschnitt der Entropie der Daten, die nicht durch das in den Daten enthaltene Regelwerk gedeckt sein muss. Zu Gunsten einer Regression ist die Information der gewichtete Durchschnitt des durchschnittlichen quadrierten Fehlers der Daten, die ebenso nicht durch Regeln in den Daten ge-

[41] Breiman, L.: “Random Forests”. Machine Learning 45 (1), 2001, S. 5 - 32.

deckt sein müssen. Beim Beschneiden des Baumes (pruning) wird der gewichtete Durchschnitt der Klassifikationsergebnisse auf den Pruningdaten für die Klassifikation verwendet. Bei einer Regression wird der gewichtete Durchschnitt des quadrierten Fehlers auf den Pruningdaten verwendet.

- Decision Table

  Klasse zum Aufbau und Nutzung einer einfachen Entscheidungstabelle.[42]

- Jrip

  Diese Klasse implementiert einen proportionalen Regellerner (Repeated Incremental Pruning to Produce Error Reduction (RIPPER)).[43]

- M5 Rules

  Diese Klasse generiert eine Entscheidungsliste für Regressionsprobleme nach dem Prinzip Teile-und-Herrsche. Bei jeder Iteration wird ein Baum durch Anwendung des M5-Algorithmus gebaut und der Weg zum besten Blatt zu einer Regel, die für eine Klassifikation anzuwenden ist.[44]

- Nnge

  Hierbei handelt es sich um einen, dem Nächsten-Nachbarn ähnlichen Algorithmus, der non-nested generalisierte Exemplare nutzt. Diese so genannten Hyper-Rechtecke lassen sich als Wenn-Dann-Regeln betrachten.[45]

---

[42] Kohavi, R.: “The Power of Decision Tables.” In: “Proc European Conference on Machine Learning”, 1996.

[43] Cohen, W. W.: “Fast Effective Rule Induction”. In: “Machine Learning”, Proceedings of the Twelfth International Conference' (ML95), 1995.

[44] Hall, M; Holmes, G.; Frank, E.; “Generating Rule Sets from Model Trees”. Proceedings of the Twelfth Australian Joint Conference on Artificial Intelligence, Sydney, Australia. Springer-Verlag, 1999, S. 1 - 12.

[45] Brent, M.: “Instance-Based learning : Nearest Neighbour With Generalization”, Master Thesis, University of Waikato, Hamilton, New Zealand, 1995, und Sylvain, R.: “Nearest Neighbour With Generalization”, Unpublished, University of Canterbury, Christchurch, New Zealand, 2002.

– OneR

  Klasse zur Bildung und Benutzung eines 1R-Klassifikators. Dabei wird das Attribut verwendet, welches den kleinsten Fehler aufweist, um eine Vorhersage und Diskretisierung numerische Attribute durchzuführen.[46]

– PART

  Klasse zur Erstellung einer PART-Entscheidungliste durch Anwendung des Teile-und-Herrsche-Prinzips. Dabei wird ein partieller C4.5-Entscheidungsbaum in jeder Iteration aufgebaut, von dem der Weg zum besten Blatt als Regel zu verwenden ist.[47]

– Prism

  Klasse zum Aufbau und Anwendung einer Menge von PRISM-Regeln. Dabei ist zu beachten, dass lediglich mit nominalen Werten gearbeitet werden kann und fehlende Werte nicht zulässig sind.[48]

– Ridor

  Die Implementierung eines RIpple-Down-Rule Trainingsalgorithmus generiert zunächst eine Default-Regel mit der zuletzt berechneten und gewichteten Fehlerrate eines vorherigen Durchlaufs. Daraufhin werden die besten Erwartungswerte für jeden Durchgang bestimmt. Hieraus wird eine Baumstruktur generiert, welche die Erwartungswerte erfasst. Die Erwartungswerte bilden dann eine Menge von Regeln, die verwendet werden können.

– ZeroR

  Klasse zum Aufbau und Nutzung eines 0R-Klassifikators. Dieser berechnet Erwartungswerte für den Durchschnitt zur Klassifikation einer numerischen Klasse.

---

[46] Holte, R. C.: “Very simple classification rules perform well on most commonly used datasets”. Machine Learning, Vol. 11, 1996, S. 63 - 91.

[47] Frank, E.; Witten, I. H.: “Generating Accurate Rule Sets Without Global Optimization.” In: Shavlik, J. (ed.): “Machine Learning: Proceedings of the Fifteenth International Conference”, Morgan Kaufmann Publishers, 1998.

[48] Cendrowska, J.: “PRISM: An algorithm for inducing modular rules”. International Journal of Man-Machine Studies. Vol.27, No.4, 1987, S. 349 - 370.

– IB1

  Hierbei handelt es sich um den Nächsten-Nachbar-Algorithmus. Dieser nutzt die normalisierte Euklidische Distanz zur Bestimmung der Ähnlichkeit der Trainings- zur Testinstanz. Weisen mehrere Instanzen die gleiche (kleinste) Distanz auf, wird die zuerst gefundene verwendet.[49]

– IBk

  Diese Implementierung ist der K-Nächste-Nachbar-Algorithmus. In dieser werden normalisierte Attribute als Default-Wert gesetzt. Dem Anwender ist die Bestimmung des entsprechenden K-Wertes überlassen, um die Anzahl der zu prüfenden Nachbarn zu bestimmen.[50]

– KStar

  KStar ist ein instanzenbasierter Klassifikator, der eine Klasse der Instanz, basierend auf einer Trainingsinstanz, verwendet. Deren Ähnlichkeit zu den anderen Klassen im Trainingsdatensatz wurde zuvor bestimmt. Der zentrale Unterschied zu anderen instanzenbasierten Lernverfahren liegt darin, dass eine Entropie-basierte Distanzfunktion Anwendung findet.[51]

– LBR

  Lazy Bayesian Rules Classifier ersetzt die Annahme der Unabhängigkeit der Attribute, die vom Naive-Bayes-Klassifikator zu Grunde gelegt wird. Für eine kleine Anzahl an zu klassifizierenden Objekten verbessert dies oftmals das Ergebnis.

– LWL

  Klasse zur Umsetzung des *locally weighted learning*. Die Klassifikation wird dabei entweder durch Naive-Bayes oder einer linearen Regression durchgeführt.[52]

---

[49] Aha, D.; Kibler, D.: “Instance-based learning algorithms”. In: Machine Learning, Vol. 6, 1991, S. 37 - 66.

[50] Aha, D.; Kibler, D.: “Instance-based learning algorithms”. In: Machine Learning, Vol. 6, 1991, S. 37 - 66.

[51] John, G. C.; Leonard, E. T.: “K*: An Instance- based Learner Using an Entropic Distance Measure”, Proceedings of the 12th International Conference on Machine learning, 1996, S. 108 - 114.

[52] Frank, E.; Hall, M.; Pfahringer, B.: “Locally Weighted Naive Bayes”. Conference on Uncertainty in AI. Zusätzlich: Atkeson, C.; Moore, A.; Schaal, S.: “Locally weighted learning" AI Reviews”, 1996.

– Additive Regression

  Die Additive Regression ist ein Meta-Klassifikator, der eine einfache Regression erweitert. Jede Iteration verbessert das aufgebaute Modell gemäß des verbleibenden Rests der vorherigen Iteration. Die Gesamtvorhersage zur Klassifikation besteht aus der Summe der Teilvorhersagen jeden Durchlaufs.[53]

– AttributeSelectClassifier

  Dieser Klassifikator reduziert die Anzahl der Dimensionen der Trainings- und Testdaten durch Attributauswahl. Erst danach werden diese an den Klassifikator übergeben. Dies verbessert die Laufzeit, da ein geringerer Datenumfang betrachtet wird.

– Bagging

  Diese Klasse führt ein Bagging der Klassifikation durch, um Varianzen zu reduzieren.[54]

– Classification via Regression

  Durch diese Klasse wird eine Klassifikation durch Anwendung von Regressionsmethoden ausgeführt. Die Klasse arbeitet lediglich binär und das Klassifikationsmodell erfasst sämtliche Werte einer Klasse.[55]

– Cost SensitiveClassifier

  Diese Meta-Klassifikator versucht die angewendeten Basis-Klassifikatoren kostenreduzierend (im Sinne der Laufzeit und Speichernutzung) auszuführen. Dazu können zwei Methoden ausgeführt werden. Möglich ist eine erneute Gewichtung der Trainingsinstanzen gemäß der Kosten einer Klasse. Eine andere Umsetzung ist die Bestimmung der Klasse, die wahrscheinlich die geringste Fehlklassifikation erzeugt. Performancekosten können reduziert werden, indem die Wahrscheinlichkeitsbestimmungen durch Bagging verbessert werden.

---

[53] J. H.: “Stochastic Gradient Boosting”. Technical Report Stanford University. http://www-stat.stanford.edu/~jhf/ftp/stobst.ps, 1999.

[54] Breiman, L.: “Bagging predictors”. Machine Learning, 24(2): 1996, S. 123 - 140.

[55] Frank, E.; Wang, Y.; Inglis, S.; Holmes, G.; Witten, I. H.: “Using model trees for classification”, Machine Learning, Vol.32, No.1, 1998, S. 63 - 76.

- CVParameterSelection

  Diese Klasse realisiert eine Parameterauswahl durch Cross-Validierung aller Klassifikatoren.[56]

- Decorate

  Decorate ist ein Meta-Lernverfahren, der diverse Klassifikatoreinheiten durch künstliche Trainingsbeispiele aufbaut.[57]

- Filtered Classifier

  Es wird ein Filter aufgebaut, der vollständig auf Trainingsdaten basiert und in seiner Struktur durch andere Strukturen in den Testdaten nicht verändert wird.

- Grading

  Die Basisklassifikatoren bekommen einen Gütegrad zugewiesen.[58]

- Logit Boost

  Diese Klasse führt eine additive logistische Regression aus. Als Basis wird ein Regressionsschema genutzt.[59]

- MetaCost

  Dieser Klassifikator realisiert identische Ergebnisse wie das Bagging. Jedoch hat dieser Vorteile durch eine schnellere Klassifikation in der Basisklassifikation, da die Anzahl der Iterationen niedriger ist.[60]

---

[56] Kohavi, R.: "Wrappers for Performance Enhancement and Oblivious Decision Graphs". PhD Thesis. Department of Computer Science, Stanford University, 1995.

[57] Melville, P.; Mooney, R. J.: "Constructing diverse classifier ensembles using artificial training examples (IJCAI 2003)" und Melville, P.; Mooney, R. J.: "Creating diversity in ensembles using artificial data.", 2003.

[58] Seewald, A. K.; Fuernkranz, J.: "An Evaluation of Grading Classifiers", in Hoffmann, F. et al. (eds.), Advances in Intelligent Data Analysis, 4th International Conference, IDA 2001, Proceedings, Springer, Berlin/Heidelberg/New York/Tokyo, 2001, S. 115 - 124.

[59] Friedman, J.; Hastie, T.; Tibshirani, R.: "Additive Logistic Regression: a Statistical View of Boosting". Technical report. Stanford University, 1998.

[60] Domingos, P.: "MetaCost: A general method for making classifiers cost-sensitive", Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining, 1999, S. 155 - 164.

- MultiBoostAB

  MultiBoosting ist eine Erweiterung der AdaBoost-Technik. Dabei wird C4.5 als Basis-Lernalgorithmus verwendet, um einen geringeren Fehler als AdaBoost zu erzielen. Zusätzlich lassen sich Prozessschritte parallel ausführen, so dass die Klassifikationsgeschwindigkeit verbessert wird.[61]

- MultiClassClassifier

  Dieser Meta-Klassifikator kann eine Multiklassenklassifikation mit 2-Klassen-Klassifikatoren ausführen. Zusätzlich wird bei der Ausgabe eine Fehlerkorrektur durchgeführt.

- MultiScheme

  Diese Klasse bestimmt einen Klassifikator durch Cross-Validierung der Ergebnisse auf einem Trainingsdatenbestand.

- OrdinalClassClassifier

  Dieser Meta-Klassifikator erlaubt eine Standardklassifikation bei ordinalen Klassenproblemen.[62]

- RacedIncrementalLogitBoost

  Der Klassifikator führt inkrementelles Lernen auf großen Datenbeständen durch ein so genanntes logistisches Boosting aus.

- RandomCommittee

  Klasse zum Aufbau einer Sammlung zufallsorientierter Klassifikatoren. Jeder der Klassifikatoren nimmt eine zufällige Anzahl an Daten als Trainingsbasis. Die abschließende Aussage ist der Durchschnitt aller Einzelergebnisse.

---

[61] Webb, G. I.: "MultiBoosting: A Technique for Combining Boosting and Wagging". Machine Learning, 40(2): 2000, S. 159-196, Kluwer Academic Publishers, Boston.

[62] Frank, E.; Hall, M.: "A simple approach to ordinal prediction. 12th European Conference on Machine Learning." Freiburg, Germany.

- RegressionByDiscretization

  Dieses Regressionsschema arbeitet mit diskretisierten Attributen. Die Aussage ist der Erwartungswert eines Klassenwertes eines diskretisierten Intervalls (basierend auf den vorhergesagten Wahrscheinlichkeitswerten eines jeden Intervalls).

- Stacking

  Diese Implementation kombiniert unterschiedliche Klassifikatoren durch Nutzung der Stacking-Methode. Dabei kann sowohl eine Klassifikation als auch eine Regression ausgeführt werden.[63]

- StackingC

  StackingC ist eine effizientere Implementierung des herkömmlichen Stacking.[64]

- ThresholdSelector

  Hierbei handelt es sich um einen Meta-Klassifikator, der einen Mittelwert für den wahrscheinlichen Ausgabewert bestimmt. Dieser Mittelwert ist von Anfang an definiert, so dass anhand dessen eine Optimierung ausführbar ist. Hier ist dies das $F_ß$-Maß. Die Ausgabewahrscheinlichkeiten liegen zwischen 0 und 1, was eine adäquate Bandbreite darstellt.

- Vote

  Diese Klasse kombiniert Klassifikatoren durch Bestimmung des ungewichteten Durchschnitts der Wahrscheinlichkeiten bei einer Klassifikation oder einer numerischen Vorhersage bei einer Regression.

[63] Wolpert, D. H.: “Stacked generalization”. Neural Networks, 5, 1972, S. 241 - 259, Pergamon Press.

[64] Seewald, A. K.: “How to Make Stacking Better and Faster While Also Taking Care of an Unknown Weakness”. In: Sammut, C., Hoffmann, A. (eds.): “Proceedings of the Nineteenth International Conference on Machine Learning (ICML 2002)”, Morgan Kaufmann Publishers, 2002, S. 554 - 561.

4.2.3 Ergebnisse

Die Ergebnisse für das $F_ß$-Maß für die durchgeführten neun Versuche bei den verschiedenen Vorverarbeitungsschritten gibt die folgende Tabelle 2 an. Die oben aufgeführten Algorithmen, die nicht in der Tabelle enthalten sind, konnten die Textdokumente nach Vorverarbeitung nicht verarbeiten. Es wurden numerische Attribute verlangt, die Vorverarbeitung jedoch nominale Attribute erzeugt.

Tabelle 2: Klassifikationsergebnis

Test-Nr. / Algorithmus	1	2	3	4	5	6	7	8	9
Complement Naive Bayes	**0,8005**	0,7805	0,7765	0,7865	0,7905	0,7355	0,7105	0,7850	0,7845
Naive Bayes Multinomial	**0,7950**	0,7770	0,7810	0,7870	0,7870	0,7315	0,6995	0,7865	0,7865
Naive Bayes Updateable	0,7590	0,7130	0,7355	0,7645	0,7565	0,7375	0,7555	0,7650	**0,7670**
SMO	0,7390	0,7430	0,7450	0,7515	0,7575	**0,7875**	0,7685	0,7460	0,7480
RBF Network	0,5390	0,5600	0,5640	0,5615	0,5635	**0,5715**	0,5585	0,5575	0,5595
AD Tree	0,7060	0,5275	**0,7460**	0,7010	0,7240	0,7050	0,6400	0,7010	0,7010
Rep Tree	0,6580	0,6265	**0,7135**	0,6685	0,6625	0,6530	0,6295	0,6685	0,6685
User Classifier	0,3025	**0,5100**	0,3035	0,3035	0,3035	0,3035	0,3035	0,3035	0,3035
Conjunctive Rule	0,5190	0,5100	0,5720	0,6250	**0,6265**	0,5625	0,4440	0,6250	0,6250
Jrip	0,7045	0,6615	0,7170	0,7155	0,7065	0,6835	0,5985	**0,7300**	0,6900
OneR	0,5275	0,5750	0,6001	0,6060	0,5640	**0,6110**	0,5065	0,5995	0,6050
PART	**0,7195**	0,6445	0,6925	0,7065	0,7045	0,6870	0,6645	0,7065	0,7065
Ridor	0,6725	0,6510	0,6550	0,6980	**0,7030**	0,5930	0,5480	0,6980	0,6980
ZeroR	0,3025	0,3025	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**
IB1	0,4270	0,4250	0,4245	0,4515	0,4140	**0,6395**	0,4425	0,4485	-
KStar	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
Bagging	0,7315	0,7290	**0,7615**	0,7550	0,7495	-	0,7150	0,7550	0,7550
Classification via Regression	0,6945	0,6455	0,7035	**0,7190**	0,6790	0,6860	0,6410	**0,7190**	0,7140
CV Parameter Selection	0,3025	0,3025	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**
Filtered Classifier	0,6865	0,6535	**0,6910**	0,6620	0,6545	0,6570	0,6625	0,6620	0,6620
Grading	0,3025	0,3025	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**
Logit Boost	0,6945	0,6745	0,7305	0,7365	0,7310	0,6725	0,6155	0,7365	**0,7400**
Multi Boost AB	0,6920	0,6685	**0,7360**	0,7320	0,7055	0,6275	0,6645	0,7320	0,7320
Multi Class Classifier	0,3025	0,3025	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**
Multi Scheme	0,3025	0,3025	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**
Ordinal Class Classifier	0,3025	0,3025	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**
Raced Increm. Logit Boost	0,3025	0,3025	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**
Stacking	0,3025	0,3025	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**
StackingC	0,3025	0,3025	0,3035	0,3035	0,3035	0,3035	0,3035	**0,3570**	0,3035
Vote	0,3025	0,3025	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**	**0,3035**

Die Tabelle zeigt mit dem *Complement-Naiven-Bayes-Algorithmus* das beste Klassifikationsergebnis mit einem $F_ß$-Maß von 0,8005 in Test-Nr. 1. Dies bedeutet, dass circa 80 Prozent der zur Verfügung gestellten Dokumente korrekt im Sinne der Vorgaben klassifiziert werden konnte. Der Multinomiale-Naive-Bayes-Algorithmus erzielt im gleichen Versuch ein Ergebnis von 0,7950. Die vorliegende Implementierung der Support Vector Machines (SMO) erzielt ein $F_ß$-Maß von 0,7875 in Test-Nr. 6 und ist damit nur unwesentlich schlechter. Zu be-

achten ist an dieser Stelle jedoch, dass Versuch 6 ein geringeres Preprocessing erlebt hat, als Versuch 1. Damit ist unter zeitlichen Aspekten erwähnenswert, dass ein geringeres Preprocssing eine Reduktion der gesamten Klassifikationszeit bedeutet, die nicht vollständig durch eine verlängerte Klassifikationszeit (bedingt durch die höhere Anzahl der Attribute) eliminiert wird. Ausgenommen den Jrip-Algorithmus, der in Versuch 7 einbricht, dies in Versuch 8 jedoch wieder behebt, verlaufen alle Klassifikationsergebnisse stabil. Diejenigen Algorithmen, die über sämtliche Versuche einen Wert von 0,3035 erzielen, kennzeichnen damit, dass sie nicht in der Lage waren, Muster zu erkennen und zu lernen, um eine Klassifikation adäquat ausgeführt werden konnte. Es wurde im Ergebnis ohne ein Lernen klassifiziert. Zellen, die einen Strich enthalten, symbolisieren, dass eine Klassifikation nicht möglich war. Diese Fortführung zum bereits erschienenen Arbeitsbericht zeigt noch einmal, dass eine grundsätzliche Festlegung auf einen Klassifikationsalgorithmus ohne Beachtung der zu Grunde liegenden Rahmenbedingungen nicht sinnvoll ist. Regelmäßige Evaluationen und fortschreitendes Lernen der Algorithmen ist eine notwendige Voraussetzung, sinnvolle Ergebnisse für den praktischen Einsatz zu erzielen.

5 Fazit

Im Rahmen dieses Arbeitspapiers wurden zunächst die Notwendigkeit für Informationsfilterung und im Anschluss daran die theoretischen Grundlagen der Textklassifikation vorgestellt. Danach wurde das Konzept von zehn Algorithmen, anhand derer Dokumente klassifiziert werden können, dargestellt. Anhand eines Beispiels wurde die Eignung der erläuterten Verfahren überprüft.

In bisherigen Untersuchungen wurden häufig verschiedene Algorithmen als die zur Klassifizierung von Dokumenten am besten geeigneten dargestellt.[65] Diese Unsicherheit über den besten Algorithmus bestätigt der vorliegende Beitrag insofern, als dass teilweise der Naive-Bayes-Algorithmus und in anderen Versuchen die SVM's die besten Ergebnisse erzielten und damit kein Algorithmus durchgängig zu den besten Resultaten führte. Darüber hinaus ist das Resultat nach den hier vorliegenden Ergebnissen von der Art der Vorverarbeitung abhängig.

[65] Die Widersprüchlichkeit einiger Testergebnisse wird auch von Junker festgestellt. Vgl. Junker (2001, S. 20 f.) Weitere Untersuchungsergebnisse finden sich bei Joachims (1998, S. 140 ff.), Sebastiani (2002, S. 38 ff.) und Cohen (1995, S. 131).

Die Ergebnisse schwankten zwischen den einzelnen Versuchen, jedoch verlaufen diese Ausschläge nicht für alle Algorithmen gleich, so dass eine pauschale Aussage für verschiedene Datenbestände nicht möglich erscheint. Damit ist der Nachweis erbracht, dass die Ergebnisse, die auf Basis stark standardisierter Daten erreicht wurden, auch auf unstrukturierte und unstandardisierte Daten übertragen werden können.

Das Hauptaugenmerk wird in der Praxis auf den Kennzahlen des $F_ß$-Maß liegen. Jedoch sollten auch anderen Aspekte berücksichtigt werden. Bei ähnlicher Qualität ist der Klassifikator zu nutzen, der das beste Laufzeitverhalten hat. Für die Überlegung, ob ein Algorithmus bei den geplanten Vorverarbeitungsschritten sinnvoll einzusetzen ist, ist auch das Merkmal des Laufzeitverhaltens in Relation zu steigender Termzahl relevant. Ein stark überproportionaler Anstieg der Laufzeiten spricht gegen den Einsatz dieses Algorithmus, wenn wenig Vorverarbeitung und damit viele Terme genutzt werden sollen.
Auch ohne spezielle Datenaufbereitung lassen sich ähnliche Ergebnisse erzielen, wie sie für Versuche mit weiteren Datenaufbereitungsschritten erreicht werden können. Daraus lässt sich die provokante These ableiten, dass sich Datenaufbereitung nicht lohnt, da die Vorverarbeitungsschritte mit Zeit- und Rechenaufwand verbunden sind. Jedoch werden bei dieser vereinfachenden These zwei Aspekte außer Acht gelassen. Zum einen führt ein höheres Maß an Datenaufbereitung in der Regel zu kleineren Dokumentenvektoren und damit zu einem schnelleren Ablauf der eigentlichen Anlernphase des Algorithmus. Auch die Klassifizierung neuer Dokumente wird beschleunigt. Zum anderen wird der Entscheidungsträger in der Regel schon geringen Qualitätsverbesserungen ein hohes Gewicht beimessen, da diese bedeuten, dass er mit weniger *uninteressanten* Informationen konfrontiert wird. Ebenso folgt daraus, dass er nur eine geringere Anzahl von *interessanten* Dokumenten übersieht, was häufig ein Wettbewerbs-vorteil ist. Ist die Genauigkeit eines automatisierten Informationssystems nicht hoch genug, wird der Entscheidungsträger dieses nicht akzeptieren, weil das Risiko zu groß ist, entscheidungsrelevante Informationen zu übersehen. Eine endgültige Entscheidung für oder gegen einen Algorithmus und eine spezielle Form der Datenaufbereitung muss auf Basis aller vorgestellten Aspekte fallen und ist sehr stark vom jeweiligen Einzelfall abhängig.

Im Rahmen dieses Beitrages wurde davon ausgegangen, dass die relevanten Dokumente bereits im Unternehmen vorhanden sind oder mittels geeigneter Verfahren aus dem Internet extrahiert werden.[66] Weitere Forschungsfelder liegen in der Untersuchung, ob die Kombination unterschiedlicher Algorithmen, die Klassifikationsgüte erhöht. Mit dem gleichen Ziel wird analysiert, wie das Preprocessing modifiziert werden kann, z. B. durch einen umfangreichen Trainingsdatensatz oder Nutzung der Entropie anstelle der invertierten Termfrequenz, so dass die Fehlklassifikation reduziert wird.

[66] Vgl. Markellos et al. (2002, S. 25 ff.).

Literaturverzeichnis

Baeza-Yates, R.; Ribeiro-Neto, B. (1999): Modern Information Retrieval. Addisson-Wesley, New York.

Cheeseman, P.; Stutz, J. (1996): Bayesian Classification (AutoClass): Theory and Results. In: Fayyad, U.; Piatetsky-Shapiro, G.; Smyth, P.; Uthurusamy, R. (Hrsg.): Advances in Knowledge Discovery and Data Mining. aaai-press, Menlo Park et al., S. 153-180.

Cohen, W. (1995): Text Categorization and Relational Learning. In: Priedetis, A.: *Machine learning: Proceedings of the twelfth International Conference on Machine Learning*; Tahoe City, California, July 9 - 12, 1995. San Francisco, S. 124 – 132.

Ferber, R. (2003): Information Retrieval – Suchmodelle und Data-Mining-Verfahren für Textsammlungen und das Web. Dpunkt, Heidelberg.

Harman, D. (1992): Relevance Feedback and Other Query Modification Techniques. In: Frakes, W.; Baeza-Yates, R. (Hrsg.): Information Retrieval. Data Structures & Algorithms. Addisson-Wesley, New York, S. 241 – 263.

Joachims, T. (1998): Text Categorization with Support Vector Machines: Learning with Many Relevant Features. URL: http://ranger.uta.edu/~alp/ix/readings/SVMsforText Categorization.pdf, 1998, Abruf am 2003-12-11.

Junker, M. (2001): Heuristisches Lernen von Regeln für die Textkategorisierung. dissertation.de, Berlin.

Kantardzic, M. (2003): Data Mining. Concepts, Models, Methods, and Algorithms. Wiley&Sons, Piscataway.

Kowalski, G. (1999): Information Retrieval Systems. Theory and Implementation. Springer, Boston et al.

Küppers, B. (1999): Data mining in der Praxis. Ein Ansatz zur Nutzung der Potentiale von Data mining im betrieblichen Umfeld. P. Lang, Frankfurt am Main et al.

Markellos, K.; Markellou, P.; Rigou, M.; Sirmakessis, S. (2002): Web Mining: The Past, the Present, and Future. In: Sirmakessis, S. (Hrsg.): Text Mining and its Applications. Results of the NEMIS Launch Conference. Springer, Berlin et al., S. 25 – 35.

Salton, G. (1989): Automatic Text Processing. The Transformation, Analysis, and Retrieval of Information by Computer. Addison-Welsey, Reading et al.

Salton, G. (1988): Automatic Indexing and Abstracting. In: Willett, P. (Hrsg.): Document Retrieval Systems. Morgan Kaufmann, London, S. 42-80.

Salton, G.; McGill, M. (1983): Introduction to Modern Information Retrieval. McGraw-Hill, New York et al.

Sebastiani, F. (2002): Machine Learning in Automated Text Categorization. In: *ACM Computing Surveys,* Vol. 34, No. 1, März 2002, S. 1 – 47.

Tauritz, D. (1996): Adaptive Information Filtering as a Means to Overcome Information Overload. URL: http://web.umr.edu/~tauritzd/papers/thesis.ps.gz, Abruf am 2003-12-16.

van Rijsbergen, C. (1979): Information Retrieval. 2. Auflage. Butterworths, London et al.

Witten, I. H.; Frank, E. (2000): Data Mining: Practical machine learning tools with Java implementations. Morgan Kaufmann, San Francisco.

Zarnekow, R. (1999): Softwareagenten und elektronische Kaufprozesse. Referenzmodelle zur Integration. DUV, Wiesbaden.

List of Working Papers of the Faculty of Economics and Business Administration,
Technische Universität Bergakademie Freiberg.

2000

00/1 Michael Nippa, Kerstin Petzold, Ökonomische Erklärungs- und Gestaltungsbeiträge des Realoptionen-Ansatzes, Januar.

00/2 Dieter Jacob, Aktuelle baubetriebliche Themen – Sommer 1999, Januar.

00/3 Egon P. Franck, Gegen die Mythen der Hochschulreformdiskussion – Wie Selektionsorientierung, Nonprofit-Verfassungen und klassische Professorenbeschäftigungsverhältnisse im amerikanischen Hochschulwesen zusammenpassen, erscheint in: *Zeitschrift für Betriebswirtschaft (ZfB)*, 70. (2000).

00/4 Jan Körnert, Unternehmensgeschichtliche Aspekte der Krisen des Bankhauses Barings 1890 und 1995, in: *Zeitschrift für Unternehmensgeschichte*, München, 45 (2000), 205 – 224.

00/5 Egon P. Franck, Jens Christian Müller, Die Fußball-Aktie: Zwischen strukturellen Problemen und First-Mover-Vorteilen, *Die Bank*, Heft 3/2000, 152 – 157.

00/6 Obeng Mireku, Culture and the South African Constitution: An Overview, Februar.

00/7 Gerhard Ring, Stephan Oliver Pfaff, CombiCar: Rechtliche Voraussetzungen und rechtliche Ausgestaltung eines entsprechenden Angebots für private und gewerbliche Nutzer, Februar.

00/8 Michael Nippa, Kerstin Petzold, Jamina Bartusch, Neugestaltung von Entgeltsystemen, Besondere Fragestellungen von Unternehmen in den Neuen Bundesländern – Ein Beitrag für die Praxis, Februar.

00/9 Dieter Welz, Non-Disclosure and Wrongful Birth , Avenues of Liability in Medical Malpractice Law, März.

00/10 Jan Körnert, Karl Lohmann, Zinsstrukturbasierte Margenkalkulation, Anwendungen in der Marktzinsmethode und bei der Analyse von Investitionsprojekten, März.

00/11 Michael Fritsch, Christian Schwirten, R&D cooperation between public research institutions - magnitude, motives and spatial dimension, in: Ludwig Schätzl und Javier Revilla Diez (eds.), *Technological Change and Regional Development in Europe*, Heidelberg/New York 2002: Physica, 199 – 210.

00/12 Diana Grosse, Eine Diskussion der Mitbestimmungsgesetze unter den Aspekten der Effizienz und der Gerechtigkeit, März.

00/13 Michael Fritsch, Interregional differences in R&D activities – an empirical investigation, in: *European Planning Studies*, 8 (2000), 409 – 427.

00/14 Egon Franck, Christian Opitz, Anreizsysteme für Professoren in den USA und in Deutschland – Konsequenzen für Reputationsbewirtschaftung, Talentallokation und die Aussagekraft akademischer Signale, in: *Zeitschrift Führung + Organisation (zfo)*, 69 (2000), 234 – 240.

00/15 Egon Franck, Torsten Pudack, Die Ökonomie der Zertifizierung von Managemententscheidungen durch Unternehmensberatungen, April.

00/16 Carola Jungwirth, Inkompatible, aber dennoch verzahnte Märkte: Lichtblicke im angespannten Verhältnis von Organisationswissenschaft und Praxis, Mai.

00/17 Horst Brezinski, Der Stand der wirtschaftlichen Transformation zehn Jahre nach der Wende, in: Georg Brunner (Hrsg.), *Politische und ökonomische Transformation in Osteuropa*, 3. Aufl., Berlin 2000, 153 – 180.

00/18 Jan Körnert, Die Maximalbelastungstheorie Stützels als Beitrag zur einzelwirtschaftlichen Analyse von Dominoeffekten im Bankensystem, in: Eberhart Ketzel, Stefan Prigge u. Hartmut Schmidt (Hrsg.), *Wolfgang Stützel – Moderne Konzepte für Finanzmärkte, Beschäftigung und Wirtschaftsverfassung*, Verlag J. C. B. Mohr (Paul Siebeck), Tübingen 2001, 81 – 103.

00/19 Cornelia Wolf, Probleme unterschiedlicher Organisationskulturen in organisationalen Subsystemen als mögliche Ursache des Konflikts zwischen Ingenieuren und Marketingexperten, Juli.

00/20 Egon Franck, Christian Opitz, Internet-Start-ups – Ein neuer Wettbewerber unter den „Filteranlagen“ für Humankapital, erscheint in: *Zeitschrift für Betriebswirtschaft (ZfB)*, 70 (2001).

00/21 Egon Franck, Jens Christian Müller, Zur Fernsehvermarktung von Sportligen: Ökonomische Überlegungen am Beispiel der Fußball-Bundesliga, erscheint in: Arnold Hermanns und Florian Riedmüller (Hrsg.), *Management-Handbuch Sportmarketing*, München 2001.

00/22 Michael Nippa, Kerstin Petzold, Gestaltungsansätze zur Optimierung der Mitarbeiter-Bindung in der IT-Industrie - eine differenzierende betriebswirtschaftliche Betrachtung -, September.

00/23 Egon Franck, Antje Musil, Qualitätsmanagement für ärztliche Dienstleistungen – Vom Fremd- zum Selbstmonitoring, September.

00/24 David B. Audretsch, Michael Fritsch, Growth Regimes over Time and Space, *Regional Studies*, 36 (2002), 113 – 124.

00/25 Michael Fritsch, Grit Franke, Innovation, Regional Knowledge Spillovers and R&D Cooperation, *Research Policy*, 33 (2004), 245-255.

00/26 Dieter Slaby, Kalkulation von Verrechnungspreisen und Betriebsmittelmieten für mobile Technik als Grundlage innerbetrieblicher Leistungs- und Kostenrechnung im Bergbau und in der Bauindustrie, Oktober.

00/27 Egon Franck, Warum gibt es Stars? – Drei Erklärungsansätze und ihre Anwendung auf verschiedene Segmente des Unterhaltungsmarktes, *Wirtschaftsdienst – Zeitschrift für Wirtschaftspolitik*, 81 (2001), 59 – 64.

00/28 Dieter Jacob, Christop Winter, Aktuelle baubetriebliche Themen – Winter 1999/2000, Oktober.

00/29 Michael Nippa, Stefan Dirlich, Global Markets for Resources and Energy – The 1999 Perspective - , Oktober.

00/30 Birgit Plewka, Management mobiler Gerätetechnik im Bergbau: Gestaltung von Zeitfondsgliederung und Ableitung von Kennziffern der Auslastung und Verfügbarkeit, Oktober.

00/31 Michael Nippa, Jan Hachenberger, Ein informationsökonomisch fundierter Überblick über den Einfluss des Internets auf den Schutz Intellektuellen Eigentums, Oktober.

00/32 Egon Franck, The Other Side of the League Organization – Efficiency-Aspects of Basic Organizational Structures in American Pro Team Sports, Oktober.

00/33 Jan Körnert, Cornelia Wolf, Branding on the Internet, Umbrella-Brand and Multiple-Brand Strategies of Internet Banks in Britain and Germany, erschienen in Deutsch: *Die Bank*, o. Jg. (2000), 744 – 747.

00/34 Andreas Knabe, Karl Lohmann, Ursula Walther, Kryptographie – ein Beispiel für die Anwendung mathematischer Grundlagenforschung in den Wirtschaftswissenschaften, November.

00/35 Gunther Wobser, Internetbasierte Kooperation bei der Produktentwicklung, Dezember.

00/36 Margit Enke, Anja Geigenmüller, Aktuelle Tendenzen in der Werbung, Dezember.

2001

01/1 Michael Nippa, Strategic Decision Making: Nothing Else Than Mere Decision Making? Januar.

01/2 Michael Fritsch, Measuring the Quality of Regional Innovation Systems – A Knowledge Production Function Approach, *International Regional Science Review*, 25 (2002), 86-101.

01/3 Bruno Schönfelder, Two Lectures on the Legacy of Hayek and the Economics of Transition, Januar.

01/4 Michael Fritsch, R&D-Cooperation and the Efficiency of Regional Innovation Activities, *Cambridge Journal of Economics*, 28 (2004), 829-846.

01/5 Jana Eberlein, Ursula Walther, Änderungen der Ausschüttungspolitik von Aktiengesellschaften im Lichte der Unternehmenssteuerreform, *Betriebswirtschaftliche Forschung und Praxis*, 53 (2001), 464 - 475.

01/6 Egon Franck, Christian Opitz, Karriereverläufe von Topmanagern in den USA, Frankreich und Deutschland – Elitenbildung und die Filterleistung von Hochschulsystemen, *Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung (zfbf)*, (2002).

01/7 Margit Enke, Anja Geigenmüller, Entwicklungstendenzen deutscher Unternehmensberatungen, März.

01/8 Jan Körnert, The Barings Crises of 1890 and 1995: Causes, Courses, Consequences and the Danger of Domino Effects, *Journal of International Financial Markets, Institutions & Money*, 13 (2003), 187 – 209.

01/9 Michael Nippa, David Finegold, Deriving Economic Policies Using the High-Technology Ecosystems Approach: A Study of the Biotech Sector in the United States and Germany, April.

01/10 Michael Nippa, Kerstin Petzold, Functions and roles of management consulting firms – an integrative theoretical framework, April.

01/11 Horst Brezinski, Zum Zusammenhang zwischen Transformation und Einkommensverteilung, Mai.

01/12 Michael Fritsch, Reinhold Grotz, Udo Brixy, Michael Niese, Anne Otto, Gründungen in Deutschland: Datenquellen, Niveau und räumlich-sektorale Struktur, in: Jürgen Schmude und Robert Leiner (Hrsg.), *Unternehmensgründungen - Interdisziplinäre Beiträge zum Entrepreneurship Research*, Heidelberg 2002: Physica, 1 – 31.

01/13 Jan Körnert, Oliver Gaschler, Die Bankenkrisen in Nordeuropa zu Beginn der 1990er Jahre - Eine Sequenz aus Deregulierung, Krise und Staatseingriff in Norwegen, Schweden und Finnland, *Kredit und Kapital*, 35 (2002), 280 – 314.

01/14 Bruno Schönfelder, The Underworld Revisited: Looting in Transition Countries, Juli.

01/15 Gert Ziener, Die Erdölwirtschaft Russlands: Gegenwärtiger Zustand und Zukunftsaussichten, September.

01/16 Margit Enke, Michael J. Schäfer, Die Bedeutung der Determinante Zeit in Kaufentscheidungsprozessen, September.

01/17 Horst Brezinski, 10 Years of German Unification – Success or Failure? September.

01/18 Diana Grosse, Stand und Entwicklungschancen des Innovationspotentials in Sachsen in 2000/2001, September.

2002

02/1 Jan Körnert, Cornelia Wolf, Das Ombudsmannverfahren des Bundesverbandes deutscher Banken im Lichte von Kundenzufriedenheit und Kundenbindung, in: *Bank und Markt*, 31 (2002), Heft 6, 19 – 22.

02/2 Michael Nippa, The Economic Reality of the New Economy – A Fairytale by Illusionists and Opportunists, Januar.

02/3 Michael B. Hinner, Tessa Rülke, Intercultural Communication in Business Ventures Illustrated by Two Case Studies, Januar.

02/4 Michael Fritsch, Does R&D-Cooperation Behavior Differ between Regions? *Industry and Innovation*, 10 (2003), 25-39.

02/5 Michael Fritsch, How and Why does the Efficiency of Regional Innovation Systems Differ? in: Johannes Bröcker, Dirk Dohse and Rüdiger Soltwedel (eds.), *Innovation Clusters and Interregional Competition*, Berlin 2003: Springer, 79-96.

02/6 Horst Brezinski, Peter Seidelmann, Unternehmen und regionale Entwicklung im ostdeutschen Transformationsprozess: Erkenntnisse aus einer Fallstudie, März.

02/7 Diana Grosse, Ansätze zur Lösung von Arbeitskonflikten – das philosophisch und psychologisch fundierte Konzept von Mary Parker Follett, Juni.

02/8 Ursula Walther, Das Äquivalenzprinzip der Finanzmathematik, Juli.

02/9 Bastian Heinecke, Involvement of Small and Medium Sized Enterprises in the Private Realisation of Public Buildings, Juli.

02/10 Fabiana Rossaro, Der Kreditwucher in Italien – Eine ökonomische Analyse der rechtlichen Handhabung, September.

02/11 Michael Fritsch, Oliver Falck, New Firm Formation by Industry over Space and Time: A Multi-Level Analysis, Oktober.

02/12 Ursula Walther, Strategische Asset Allokation aus Sicht des privaten Kapitalanlegers, September.

02/13 Michael B. Hinner, Communication Science: An Integral Part of Business and Business Studies? Dezember.

2003

03/1 Bruno Schönfelder, Death or Survival. Post Communist Bankruptcy Law in Action. A Survey, Januar.

03/2 Christine Pieper, Kai Handel, Auf der Suche nach der nationalen Innovationskultur Deutschlands – die Etablierung der Verfahrenstechnik in der BRD/DDR seit 1950, März.

03/3 Michael Fritsch, Do Regional Systems of Innovation Matter? in: Kurt Huebner (ed.): *The New Economy in Transatlantic Perspective - Spaces of Innovation*, Abingdon 2005: Routledge, 187-203.

03/4 Michael Fritsch, Zum Zusammenhang zwischen Gründungen und Wirtschaftsentwicklung, in Michael Fritsch und Reinhold Grotz (Hrsg.), *Empirische Analysen des Gründungsgeschehens in Deutschland*, Heidelberg 2004: Physica 199-211.

03/5 Tessa Rülke, Erfolg auf dem amerikanischen Markt

03/6 Michael Fritsch, Von der innovationsorientierten Regionalförderung zur regionalisierten Innovationspolitik, in: Michael Fritsch (Hrsg.): *Marktdynamik und Innovation – Zum Gedenken an Hans-Jürgen Ewers*, Berlin 2004: Duncker & Humblot, 105-127.

03/7 Isabel Opitz, Michael B. Hinner (Editor), Good Internal Communication Increases Productivity, Juli.

03/8 Margit Enke, Martin Reimann, Kulturell bedingtes Investorenverhalten – Ausgewählte Probleme des Kommunikations- und Informationsprozesses der Investor Relations, September.

03/9 Dieter Jacob, Christoph Winter, Constanze Stuhr, PPP bei Schulbauten – Leitfaden Wirtschaftlichkeitsvergleich, Oktober.

03/10 Ulrike Pohl, Das Studium Generale an der Technischen Universität Bergakademie Freiberg im Vergleich zu Hochschulen anderer Bundesländer (Niedersachsen, Mecklenburg-Vorpommern) – Ergebnisse einer vergleichenden Studie, November.

2004

04/1 Michael Fritsch, Pamela Mueller, The Effects of New Firm Formation on Regional Development over Time, *Regional Studies*, 38 (2004), 961-975.

04/2 Michael B. Hinner, Mirjam Dreisörner, Antje Felich, Manja Otto, Business and Intercultural Communication Issues – Three Contributions to Various Aspects of Business Communication, Januar.

04/3 Michael Fritsch, Andreas Stephan, Measuring Performance Heterogeneity within Groups – A Two-Dimensional Approach, Januar.

04/4 Michael Fritsch, Udo Brixy, Oliver Falck, The Effect of Industry, Region and Time on New Business Survival – A Multi-Dimensional Analysis, Januar.

04/5 Michael Fritsch, Antje Weyh, How Large are the Direct Employment Effects of New Businesses? – An Empirical Investigation, März.

04/6 Michael Fritsch, Pamela Mueller, Regional Growth Regimes Revisited – The Case of West Germany, in: Michael Dowling, Jürgen Schmude and Dodo von Knyphausen-Aufsess (eds.): *Advances in Interdisciplinary European Entrepreneurship Research Vol. II*, Münster 2005: LIT, 251-273.

04/7 Dieter Jacob, Constanze Stuhr, Aktuelle baubetriebliche Themen – 2002/2003, Mai.

04/8 Michael Fritsch, Technologietransfer durch Unternehmensgründungen – Was man tun und realistischerweise erwarten kann, in: Michael Fritsch and Knut Koschatzky (eds.): *Den Wandel gestalten – Perspektiven des Technologietransfers im deutschen Innovationssystem*, Stuttgart 2005: Fraunhofer IRB Verlag, 21-33.

04/9 Michael Fritsch, Entrepreneurship, Entry and Performance of New Businesses – Compared in two Growth Regimes: East and West Germany, in: *Journal of Evolutionary Economics*, 14 (2004), 525-542.

04/10 Michael Fritsch, Pamela Mueller, Antje Weyh, Direct and Indirect Effects of New Business Formation on Regional Employment, Juli.

04/11 Jan Körnert, Fabiana Rossaro, Der Eigenkapitalbeitrag in der Marktzinsmethode, in: *Bank-Archiv* (ÖBA), Springer-Verlag, Berlin u. a., ISSN 1015-1516. Jg. 53 (2005), Heft 4, 269-275.

04/12 Michael Fritsch, Andreas Stephan, The Distribution and Heterogeneity of Technical Efficiency within Industries – An Empirical Assessment, August.

04/13 Michael Fritsch, Andreas Stephan, What Causes Cross-industry Differences of Technical Efficiency? – An Empirical Investigation, November.

04/14 Petra Rünger, Ursula Walther, Die Behandlung der operationellen Risiken nach Basel II - ein Anreiz zur Verbesserung des Risikomanagements? Dezember.

2005

05/1 Michael Fritsch, Pamela Mueller, The Persistence of Regional New Business Formation-Activity over Time – Assessing the Potential of Policy Promotion Programs, Januar.

05/2 Dieter Jacob, Tilo Uhlig, Constanze Stuhr, Bewertung der Immobilien von Akutkrankenhäusern der Regelversorgung unter Beachtung des neuen DRG-orientierten Vergütungssystems für stationäre Leistungen, Januar.

05/3 Alexander Eickelpasch, Michael Fritsch, Contests for Cooperation – A New Approach in German Innovation Policy, April.

05/4 Fabiana Rossaro, Jan Körnert, Bernd Nolte, Entwicklung und Perspektiven der Genossenschaftsbanken Italiens, in: *Bank-Archiv* (ÖBA), Springer-Verlag, Berlin u. a., ISSN 1015-1516, Jg. 53 (2005), Heft 7, 466-472.

05/5 Pamela Mueller, Entrepreneurship in the Region: Breeding Ground for Nascent Entrepreneurs? Mai.

05/6 Margit Enke, Larissa Greschuchna, Aufbau von Vertrauen in Dienstleistungsinteraktionen durch Instrumente der Kommunikationspolitik – dargestellt am Beispiel der Beratung kleiner und mittlerer Unternehmen, Mai.

05/7 Bruno Schönfelder, The Puzzling Underuse of Arbitration in Post-Communism – A Law and Economics Analysis. Juni.

05/8 Andreas Knabe, Ursula Walther, Zur Unterscheidung von Eigenkapital und Fremdkapital – Überlegungen zu alternativen Klassifikationsansätzen der Außenfinanzierung, Juli.

05/9 Andreas Ehrhardt, Michael Nippa, Far better than nothing at all - Towards a contingency-based evaluation of management consulting services, Juli

05/10 Loet Leydesdorff, Michael Fritsch, Measuring the Knowledge Base of Regional Innovation Systems in Germany in terms of a Triple Helix Dynamics, Juli.

05/11 Margit Enke, Steffi Poznanski, Kundenintegration bei Finanzdienstleistungen, Juli.

05/12 Olga Minuk, Fabiana Rossaro, Ursula Walther, Zur Reform der Einlagensicherung in Weißrussland - Kritische Analyse und Vergleich mit dem Deutschen Einlagensicherungssystem, August.

05/13 Brit Arnold, Larissa Greschuchna, Hochschulen als Dienstleistungsmarken – Besonderheiten beim Aufbau einer Markenidentität, August.

05/14 Bruno Schönfelder, The Impact of the War 1991 – 1995 on the Croatian Economy – A Contribution to the Analysis of War Economies, August.

05/15 Michael Fritsch, Viktor Slavtchev, The Role of Regional Knowledge Sources for Innovation – An Empirical Assessment, August.

05/16 Pamela Mueller, Exploiting Entrepreneurial Opportunities: The Impact of Entrepreneurship on Economic Growth, August.

05/17 Pamela Mueller, Exploring the Knowledge Filter: How Entrepreneurship and University-Industry Relations Drive Economic Growth, September.

05/18 Marc Rodt, Klaus Schäfer, Absicherung von Strompreisrisiken mit Futures: Theorie und Empirie, September.

05/19 Klaus Schäfer, Johannes Pohn-Weidinger, Exposures and Exposure Heding in Exchange Rate Risk Management, September.

2006

06/1 Michael Nippa, Jens Grigoleit, Corporate Governance ohne Vertrauen? Ökonomische Konsequenzen der Agency-Theorie, Januar.

06/2 Tobias Henning, Pamela Mueller, Michael Niese, Das Gründungsgeschehen in Dresden, Rostock und Karlsruhe: Eine Betrachtung des regionalen Gründungspotenzials, Januar.

06/3 Dorothea Schäfer, Dirk Schilder, Informed Capital in a Hostile Environment – The Case of Relational Investors in Germany, Januar.

06/4 Oleg Badunenko, Michael Fritsch, Andreas Stephan, Allocative Efficiency Measurement Revisited – Do We Really Need Input Prices? Januar.

06/5 Diana Grosse, Robert Ullmann, Enrico Weyh, Die Führung innovativer Teams unter Berücksichtigung rechtlicher und psychologischer Aspekte, März.

06/6 Silvia Rogler, Vergleichbarkeit von Gesamt- und Umsatzkostenverfahren – Auswirkungen auf die Jahresabschlussanalyse, März.

06/7 Michael Fritsch, Dirk Schilder, Does Venture Capital Investment Really Require Spatial Proximity? An Empirical Investigation, März.

06/8 Michael Fritsch, Viktor Slavtchev, Measuring the Efficiency of Regional Innovation Systems – An Empirical Assessment, März.

06/9 Michael Fritsch, Dirk Schilder, Is Venture Capital a Regional Business? The Role of Syndication, Mai.