Bauhaus-Universität Weimar
Fakultät Medien
Studiengang Mediensysteme

# Automatische Erkennung von Vandalismus in Wikipedia mit Hilfe maschineller Lernverfahren

## Diplomarbeit

Robert Gerling
geb. am: 31.11.1981 in Heiligenstadt

Matrikelnummer 10165

1. Gutachter: Prof. Dr. Benno Stein
2. Gutachter: Prof. Dr. Tom Gross

Datum der Abgabe: 7. Januar 2008

# Dank

An dieser Stelle bedanke ich mich bei Herrn Prof. Dr. Benno Stein für seine Aufgeschlossenheit gegenüber meiner Idee für dieses Diplomarbeitsthema, welche die vorliegende Arbeit überhaupt erst ermöglichte.
Danke sage ich auch meinem Betreuer Herrn Dipl.-Inf. Martin Potthast, welcher mich fachliche betreut hat und seine stetige Bereitschaft, meine Fragen zu beantworten, und auch für die kritische Durchsicht dieser Niederschrift.
Ganz besonders möchte ich mich bei meinen Eltern bedanken, die mir durch ihre konsequente Unterstützung das Studium ermöglicht haben.

## Erklärung

Hiermit versichere ich, dass ich die vorliegende Diplomarbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe. Die Arbeit wurde in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt.

Heilbad Heiligenstadt, den 30. Dezember 2007

Robert Gerling

## Kurzfassung

Die freie Online-Enzyklopädie Wikipedia baut auf dem Wiki-Prinzip auf. Es erlaubt jedem Besucher Wikipedias einen Artikel des Lexikons nicht nur zu lesen, sondern auch zu editieren. Diese kollaborative Zusammenarbeit führt dazu, dass die Enzyklopädie beständig weiterentwickelt wird, indem z. B. neue Fakten eingebracht, Rechtschreib- und Grammatikfehler korrigiert, weiterführende Links ergänzt oder strukturelle Verbesserungen vorgenommen werden. Diese Besonderheit des Wiki-Prinzips wird von manchen Personen missbraucht. Sie verändern Artikel mit destruktiven Absichten und untergraben damit die konstruktive Arbeit der Gemeinschaft. Dieses Phänomen wird in Wikipedia als Vandalismus bezeichnet.

Ziel dieser Arbeit ist es, Verfahren des maschinellen Lernens für die automatische Erkennung von Vandalismus in Wikipedia einzusetzen und diesen Ansatz zu evaluieren. Die Motivation für diese Untersuchung liefern die Erfolge, die maschinelle Lernverfahren bei der ähnlichen Problematik der SPAM-Mail Identifikation erzielen konnten.

Um Vandalismus automatisch erkennen zu können, ist es notwendig, sich intensiv mit dessen Charakteristik zu beschäftigen. Dazu wurden 301 Vandalismusfälle manuell untersucht und dokumentiert. Es zeigten sich vielfältige Ausprägungen von Vandalismus, welche sich anhand der vorgenommenen Veränderung am Artikel (Einfügen, Ersetzen, Löschen) und am veränderten Inhalt (Text, Link, Medien, Formatierung) unterscheiden lassen. Darüber hinaus ließen sich charakteristische Eigenschaften von Vandalismus feststellen, die erstmalig zu 16 Features modelliert wurden, welche beim maschinellen Lernen eingesetzt werden können. Die 301 Vandalismusfälle, sowie weitere 639 Beispiele von konstruktiven Veränderungen an Artikeln, wurden zum ersten Wikipedia-Vandalismus-Korpus (kurz WEBIS-VC07-11) zusammengefasst. Auf diesem Korpus wurde das Lernverfahren trainiert und getestet.

Im Zuge der Evaluierung wurde das Lernverfahren mit zwei etablierten autonomen Anti-Vandalismus-Bots verglichen. Die Bots erkennen Vandalismus aufgrund von vordefinierten Regeln. Die Vandalismuserkennung durch das Lernverfahren erzielte eine Precision von 83 % bei einen Recall von 77 % und übertraff damit den Recall der beiden Bots, von 16 % bzw. 43 %, deutlich. Dabei lag die Verarbeitungsgeschwindigkeit auf dem Niveau der Bots.

# Inhaltsverzeichnis

# Tabellenverzeichnis

# Abbildungsverzeichnis

# 1 Einleitung

Die freie Online-Enzyklopädie Wikipedia, mit ihren aktuell 8.633.370 Artikeln[1], ist ein beeindruckendes Zeugnis dafür, was aus kollaborativer Zusammenarbeit tausender Freiwilliger im World Wide Web entstehen kann. Das Besondere an Wikipedia ist, dass dieses Nachschlagewerk nicht von einer zentralen Instanz erdacht, umgesetzt und gepflegt wird, sondern dass jeder Leser gleichzeitig auch als Autor fungieren kann. Es steht jedem Besucher von Wikipedia frei, Artikel um neue Fakten zu erweitern, falsche oder fragwürdige Aussagen zu entfernen und grammatikalische, orthografische oder strukturelle Verbesserungsarbeit zu leisten. Neben textuellen Beiträgen lassen sich auch Links und verschiedene Medien-Typen (Bilder, Sounds, Filme) einbinden, um die Qualität eines Artikels zu erhöhen. Sogar das Anlegen von neuen Artikeln zu einem noch nicht vorhandenen Thema ist möglich, ebenso wie das Löschen eines Artikels. Das Wiki-Prinzips, auf dem Wikipedia basiert, ermöglicht diese vielfältigen Formen des Editierens.

Die hohe Aktivität der großen Wikipedia-Gemeinschaft sorgt dafür, dass der komplette Artikelbestand einer ständigen Veränderung unterliegt. Wird ein Artikel editiert, werden die gemachten Änderungen sofort online sichtbar. Der Artikel liegt dann in einer neuen Revision vor. Der Übergang eines Artikels zu einer neuen Revision wird als Edit bezeichnet. Die vorherige Revision des Artikels wird archiviert.

Durch das Wiki-Prinzip kann jeder Einzelne sich und sein Wissen einbringen und dadurch zur Integrität und Qualität von Wikipedia als Enzyklopädie beitragen. Es gibt jedoch Benutzer, die das unkomplizierte Editieren von Artikeln missbrauchen. Einige „schwarze Schafe“ randalieren regelrecht innerhalb des Nachschlagewerkes, indem sie mit destruktiven Absichten Veränderungen an Artikeln vornehmen. Dieses Phänomen wird in Wikipedia als Vandalismus bezeichnet.

[1] Stand: 1. Dezember 2007, http://en.wikipedia.org/wiki/Wikipedia:Multilingual_statistics#Rankings, Letzter Zugriff: 06.12.2007

Viégas und Wattenberg [31] ermittelten bei ihren Untersuchungen eine Zeitspanne von 2,8 Minuten, im Median, bis eine bestimmte Form von Vandalismus entdeckt und behoben wurde. Nach Priedhorsky et al. [22] werden 42 % Prozent des Vandalismus beim ersten Aufruf des Artikels nach der Tat endeckt und beseitigt. Bei der Größe und Dynamik von Wikipedia, mit ca. 280.000 Edits pro Tag[2], wird deutlich, dass das Entdecken und Beseitigen von Vandalismus nicht nur ein Ärgernis ist, sondern einer Sisyphosarbeit gleicht. Deshalb sind viele aktive Mitglieder der Gemeinschaft damit beschäftigt, den Schaden der durch Vandalismus entstanden ist zu reparieren, anstatt selbst konstruktiv am Wikipedia-Projekt mitzuarbeiten. Kittur et al. beschreiben diesen Misstand in [15]. Die Möglichkeit Vandalismus automatisch zu erkennen, kann diese Anstrengungen unterstützen und die Gemeinschaft entlasten.

Die Problematik der automatischen Erkennung von Vandalismus in Wikipedia ist dem Problem der SPAM-Mail Filterung bei der Emailkommunikation sehr ähnlich. Es gilt, SPAM-Mails von seriösen Mails zu unterscheiden und diese auszusortieren. Der erfolgreiche Einsatz von maschinellen Lernverfahren zur Email-SPAM Bekämpfung war und ist Gegenstand aktiver Forschung z. B. in [23], [3] und [12], weshalb es nahe liegt, diese Technologien auf die Vandalismusproblematik zu übertragen.
Gegenstand dieser Diplomarbeit ist es, (*i*) zu untersuchen, inwieweit eine automatische Erkennung von Vandalismus in Wikipedia mittels maschineller Lernverfahren möglich ist, und (*ii*) diesen Ansatz mit bereits existierenden Verfahren zu vergleichen.

Die Ausarbeitung ist wie folgt organisiert. Das zweite Kapitel bietet einen Einstieg in die Problemstellung. Das Vandalismusphänomen wird anhand der Wikipedia internen Richtlinie zu Vandalismus, einer Studie über Vandalismus, sowie wissenschaftlichen Beiträgen und eigenen Untersuchungen aufgearbeitet. Die bisher übliche Form der Kategorisierung von Vandalismus in Wikipedia wird kritisch erörtert und eine alternative Einteilungsmöglichkeit vorgestellt. Diese neue Form der Kategorisierung wird auf den ersten Wikipedia-Vandalismus-Korpus, der im Zuge dieser Arbeit entstand, angewendet und diskutiert. Zum Abschluss des zweiten Kapitels wird aufgezeigt, wie dem Vandalismusproblem in Wikipedia aktuell entgegen getreten wird und wie die bisherigen Ansätze zur automatischen Erkennung funktionieren. Das dritte Kapitel beschreibt, wie überwachtes Lernen für die automatische Erkennung von Vandalismus eingesetzt wird. Es werden Grundkenntnisse über das statistische Verfahren der Regression vermittelt, welches eine der möglichen mathematischen Grundlagen für überwachtes Lernen darstellt. Des Wei-

[2] Englisches Wikipedia im Zeitraum 10. April bis 11. März 2007 [22].

teren wird erläutert, welche für einen Vandalismus-Edit charakteristischen Eigenschaften identifiziert wurden und wie diese quantifiziert werden können, um sie als Features dem Lernverfahren zuzuführen. Die Auswertung der Leistungsfähigkeit des entwickelten Ansatzes im Vergleich zu existierenden Erkennungsmöglichkeiten ist Gegenstand des vierten Kapitels. Im abschließenden fünften Kapitel wird die vorliegende Arbeit und die erzielten Ergebnisse nochmals zusammengefasst und einige Anregungen für zukünftige Arbeiten auf diesem Gebiet zur Diskussion gestellt.

# 2 Wikipedia und Vandalismus

Dieses Kapitel beginnt mit einer kurzen Einführung in das Wiki-Prinzip und stellt die darauf aufbauende Online-Enzyklopädie Wikipedia vor. Anschließend wird der Begriff Vandalismus im Zusammenhang mit Wikipedia erörtert. Als Grundlage dafür dienen zum einen eine Richtlinie Wikipedias und eine Studie zum Thema, zum anderen zwei wissenschaftliche Arbeiten, die sich teilweise mit dem Vandalismusphänomen befassen. Aufbauend auf diesen Beiträgen und unter Berücksichtigung eigener Erfahrung wird eine neue Form der Kategorisierung von Vandalismus in Wikipedia vorgestellt. Diese alternative Einteilungsmöglichkeit wird auf den ersten Wikipedia-Vandalismus-Korpus angewendet und diskutiert. Einige ausgewählte Beispiele werden präsentiert und sollen der Veranschaulichung dienen. Im Anschluss wird ein Überblick darüber gegeben, wie momentan mit dem Vandalismusproblem umgegangen wird und welche Tools existieren, die den engagierten Benutzer Wikipedias dabei unterstützen, dem Problem entgegenzutreten. Den Abschluss dieses Kapitels bildet eine Aufarbeitung der bereits existierenden Ansätze zur automatischen Erkennung von Vandalismus, welche auf von Menschen vorgegebenen festen Regeln basieren.

## 2.1 Ein Lexikon auf Basis der Wiki-Software

Ein Wiki, mitunter auch als WikiWiki oder WikiWikiWeb bezeichnet, ist eine Ansammlung von verlinkten Webseiten, bei der es möglich ist, jede einzelne Webseite[1] online zu editieren. Das Wort Wiki stammt aus dem Hawaiianischen und bedeutet soviel wie „schnell“. Ward Cunningham [18] begann 1994 mit der Entwicklung eines solchen Systems, mit dem Ziel die Erstellung, Verlinkung und die Veränderung von Webseiten so einfach wie möglich zu gestalten.

[1] Im Kontext eines Wiki-Systems spricht man im Allgemeinen auch von einer Wikiseite

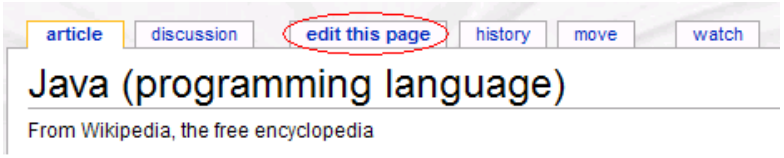


Abbildung 2.1: Über die Tabs lassen sich verschiedene Wiki-Funktionalitäten erreichen. Der hervorgehobene Reiter führt zur Bearbeitungsansicht des Artikels.

Um die Seiten des Wikis editieren zu können, stellt die Wiki-Software einen Bearbeitungsmodus zur Verfügung, der über eine Navigationsleiste (Abbildung 2.1) erreichbar ist.

Die bearbeitete Wikiseite ist nach dem Speichern sofort online und die gemachten Veränderungen sind unmittelbar für alle Besucher sichtbar. Darüber hinaus existiert eine einfache Auszeichnungssprache, die es erlaubt, z. B. Text hervorzuheben, Überschriften zu definieren oder Medieninhalte und Links einzubinden. Durch das kollaborative Schreiben unterliegen die einzelnen Seiten eines Wikis, je nachdem wie frequentiert sie sind, einer ständigen Veränderung. Die dabei entstehenden Versionen einer Wikiseite werden auch als Revisionen bezeichnet. Jede Revision ist über eine automatisch vergebene Identifikationsnummer (kurz ID) eindeutig referenzierbar.

In der Regel werden alle Revisionen einer Wikiseite in einem Archiv (Abbildung 2.2), der so genannten History, aufbewahrt. Dieses Archiv bietet zum einen einen kompakten Überblick über den zeitlichen Verlauf der Entstehung einer Wikiseite, zum anderen Details zu den einzelnen Revisionsübergängen. Darüber hinaus ist es bei Bedarf mit geringem Aufwand möglich, zu einer älteren Revision zurückzukehren und diese so neuerlich zum aktuellen Stand der Wikiseite zu machen. Wie in Abbildung 2.1 zu erkennen ist, ist auch das Archiv über einen Tab erreichbar.

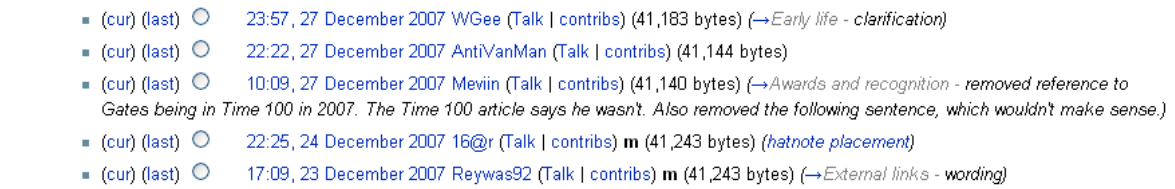


Abbildung 2.2: Auszug aus der History des Artikels über Bill Gates der englischen Wikipedia.

Meist werden Wikis dazu eingesetzt, strukturiert Wissen einer bestimmten Domäne anzulegen und gleichzeitig dieses Wissen allen Benutzern des Wikis zugänglich zu machen und es somit zu teilen. Es existieren z. B. Wikis zu den Themen Kochrezepte[2], Podcast [3] oder dem Firefox Browser[4].
Eine einzelne Wikiseite behandelt dann in der Regel einen in sich geschlossenen Teilaspekt dieser Wissensdomäne. Die Seiten innerhalb des Wikis können verlinkt werden. Man spricht dabei von internen Links. Im Gegensatz dazu steht der externe Link der auf eine Seite außerhalb des Wikis verweist. Das Wiki lässt sich nach Schlagworten durchsuchen, um so schnell zu den gewünschten Informationen zu gelangen.

Im Januar 2001 startete das Wikipedia[5]-Projekt in seiner englischen Version. Zielsetzung ist es, aufbauend auf der Wiki-Technologie, eine freie Online-Enzyklopädie zu erstellen. Im Kontext eines Lexikon werden die Wikiseiten dann als Artikel bezeichnet, da sie als ein Lexikoneintrag zu einem bestimmten Thema gedacht sind. Die internen Links werden dazu benutzt, Terme innerhalb eines Artikels auf die entsprechende Seite des korrespondierenden Artikels in Wikipedia zu verlinken. Das Projekt wächst stetig und hat bereits beeindruckende Ausmaße erreicht[6]. Allein die englische Wikipedia enthält momentan 2.060.180 Artikel [11] und belegt Platz 8 in der Rangliste der am meisten frequentierten Seiten im gesamten Internet [1]. Darüber hinaus existieren zur Zeit Derivate Wikipedias in 252 [9] verschiedenen Sprachen.

## 2.2 Vandalismus in Wikipedia

Schon immer wurde das Internet, seine Anonymität, sowie die Leichtigkeit ungestraft Schaden anzurichten, von wenigen Benutzern missbraucht. Wikipedia bildet hier keine Ausnahme. Das am häufigsten auftretende Phänomen ist als Vandalismus bekannt.

Die Benutzergemeinschaft Wikipedias definiert in einer ihrer Richtlinien[7] Vandalismus folgendermaßen:

[2] http://www.rezeptewiki.org/wiki/Hauptseite, Letzter Zugriff: 25.10.2007
[3] http://wiki.podcast.de/Hauptseite, Letzter Zugriff: 25.10.2007
[4] http://www.firefox-browser.de/wiki/Hauptseite, Letzter Zugriff: 25.10.2007
[5] Eine Wortschöpfung aus den beiden Begriffen „Wiki“ und „Encyclopedia“
[6] Alle folgenden Angaben sind Stand vom 25.10.2007
[7] http://en.wikipedia.org/wiki/Wikipedia:List_of_policies, Letzter Zugriff: 02.11.2007

> „Vandalism is any addition, removal, or change of content made in a deliberate attempt to compromise the integrity of Wikipedia“[8].

Diese Definition ist sehr allgemein gehalten, da offen gelassen wird, wodurch genau die Integrität von Wikipedia gefährdet würde. Deshalb werden zusätzlich Charakteristiken beschrieben, die verschiedene Vandalismusdelikte genauer umreißen sollen und für eine Kategorisierung von Vandalismus in Wikipedia herangezogen werden. Dabei wird immer wieder die entscheidende Bedeutung der (vermuteten) Intention des Benutzers bei der Beurteilung eines Edits herausgestrichen. Sollte die Veränderung destruktiv wirken oder ist sie durch Unwissenheit oder Unerfahrenheit zustande gekommen? Mit der Antwort kommt man unweigerlich wieder zurück auf die einleitende Definition von Vandalismus.

### 2.2.1 Charakterisierung und Einteilung von Vandalismus

Weitere Arbeiten über Vandalismus in Wikipedia bedienen sich ebenfalls typischer Charakteristiken, um Vandalismus zu kategorisieren. Teilweise werden dafür dieselben bzw. ähnliche Eigenschaften herangezogen wie in der Richtlinie Wikipedias, teilweise werden neue Eigenschaften beschrieben. In [22] und [11] werden die vorgeschlagenen Kategorisierungen durch manuelle Analysen von Vandalismusbeispielen quantifiziert.
Tabelle 2.1 fasst die Untersuchungen zusammen und stellt sie einander gegenüber.

Das Problem bei der Einteilung von Vandalismus in Wikipedia anhand der beschriebenen Eigenschaften ist, dass eine solche Kategorisierung nicht absolut ist. Sowohl die Beschreibung der Eigenschaften, als auch die Interpretation im konkreten Vandalismusfall ist subjektiv. Priedhorsky et al. [22] ließen die 308 Vandalismusbeispiele von drei verschiedenen Personen beurteilen, wobei eine mehrfache Zuordnung von Eigenschaften möglich war. Dies ist auch der Grund, warum die Angaben der ersten Spalte in Tabelle 2.1 in der Summe keine 100 % ergeben. Die dort aufgeführten Ergebnisse beziehen sich allein auf die einstimmigen (drei zu null) Entscheidungen der drei Personen bei der Zuordnung der jeweiligen Eigenschaft zu den Vandalismusbeispielen.
Es ist also nicht möglich, Vandalismus in Wikipedia allein durch beschreibende Charakteristiken zu kategorisieren.

[8] Quelle: [10], Frei deutsche Übersetzung: Vandalismus ist das Hinzufügen, Löschen, Verändern jeglichen Inhaltes mit dem Ziel die Integrität Wikipedias zu untergraben.

| | **Priedhorsky et al. [22]** | **Study 1 [11]** | **Viégas und Wattenberg [31]** | **Richtlinie Wikipedias [10]** |
|---|---|---|---|---|
| Vandalismushäufigkeit | 5 %* | 4,6 % | - | - |
| manuell analysierte Vandalismusfälle | 308 | 31 | - | - |
| gemeinsame Vandalismus Charakteristiken | | | | |
| Nonsens | 53 % | 83,87 % | - | x |
| Löschen | 23 % | 6,45 % | x | x |
| vulgärer Inhalt | 28 % | - | x | - |
| Verfälschen von Inhalten | 20 % | 0 % | - | x |
| Spam | 9 % | 9,68 % | - | x |
| Umlenken von Redirects | - | - | x | x |
| Meinung/Wertung | - | 0 % | - | x |
| individuelle Vandalismus Charakteristiken | Sonstiges 5 % | | themenfremd | Lobbyismus<br>Bilder<br>Urheberrechtsverletzungen |

Tabelle 2.1: Gegenüberstellung verschiedener Arbeiten zum Thema Vandalismus in Wikipedia. Ein „x“ kennzeichnet Charakteristiken, für die keine Häufigkeitsangaben vorliegen. Ein „-“ markiert Charakteristiken, die in der jeweiligen Arbeit unberücksichtigt blieben.
*Schätzwert nach imperfektem automatischen Analyseverfahren

Die Schwierigkeiten bei der Einteilung von Vandalismus wurden bereits von Kittur et al. in [15] beschrieben, ohne jedoch einen Lösungsansatz zu bieten. Im Zuge dieser Arbeit wurde eine alternative Einteilungsmöglichkeit konzipiert, die auf objektiven, eindeutig bestimmbaren Kriterien basiert. Diese Kategorisierung wird im nächsten Abschnitt vorgestellt.

## 2.3 Alternative Kategorisierung von Vandalismus

Die in diesem Abschnitt vorgestellte Einteilung von Vandalismus in Wikipedia nähert sich dem Problem der Kategorisierung aus einem anderen Blickwinkel. Der Vandalismus wird entlang der beiden eindeutig bestimmbaren Dimensionen „Manipulationsform“ und „veränderter Inhalt“ organisiert.

1. *Manipulationsform*: Die erste Dimension gibt an, wie eine Stelle im Artikel editiert wurde. Inhalte können eingefügt, ersetzt oder gelöscht werden.

2. *veränderter Inhalt*: Die zweite Dimension spiegelt den Typ des editierten Inhaltes

wieder. Wir unterscheiden dabei in Text, Link, Medien (Bilder, Sounds, Filme) und Formatierung.

Ein Edit kann Manipulationen an verschiedenen Stellen im Artikel beinhalten und somit mehr als einen Vandalismusdelikt aufweisen. Den Editierungsstellen ordnen wir die Kategorien in der vorgestellten Form [Manipulationsform, veränderter Inhalt] zu. Die Editierungsstellen müssen nicht zwangsläufig derselben Manipulationsform angehören. So könnte innerhalb eines einzigen Edits z. B. irrelevanter Text eingefügt und ein Link aus der Referenzenliste gelöscht werden. Dieser Edit als Ganzes würde sich dann in zwei Kategorien widerspiegeln, nämlich [Einfügen, Text] und [Löschen, Link].
Den häufigsten Formen von Vandalismus ordnen wir Charakteristiken zu. Sie sind teilweise an die Charakteristiken angelehnt, die bereits in der Literatur diskutiert wurden (Vgl. Tabelle 2.1). Darüber hinaus wurden neue Eigenschaften eingeführt, wie z. B. die plakative Formatierung.
Für die Kategorie [Einfügen, Text] werden folgende Charakteristiken verwendet:

- *Anstößig*: Vulgäre, obszöne Worte und Formulierungen. Die Auffassung dessen, was als vulgär eingestuft werden kann, ist im Kontext des Artikelthemas zu sehen.

- *Themenfremd*: Inhalt ohne thematischen Bezug zum Artikel.

- *Nonsens*: Grotesk, absurder Inhalt mit oder ohne Bezug zum Artikelthema.

- *Kauderwelsch*: Graffitiartige Zeichenfolgen, die gemeinhin nicht mehr als natürlichsprachliches Wort wahrgenommen werden. Zeichenfolgen wie sie z. B. entstehen, wenn man wahllos auf der Tastatur herumtippt.

- *Persönliche Meinung/Wertung*: Aussagen die vom neutral, fachlich beschreibenden Stil eines Lexikonartikels abweichen und eventuell die eigene Meinung des Autors oder einer gesellschaftlichen Gruppe widerspiegeln.

- *Textduplikate*: n-fache Wiederholung eines Zeichens, eines Wortes oder einer Wortgruppe.

Selbige Charakteristiken lassen sich in der Kategorie [Ersetzen, Text] anwenden, um den Text zu beschreiben, der durch das Ersetzen neu in den Artikel eingebracht wurde.
Vandalismusdelikte der Kategorie [Einfügen, Formatierung] lassen sich wie folgt charakterisieren:

- *Plakativ/Unangebracht*: überwiegend in Großbuchstaben oder fette, kursive, durchgestrichene oder ähnliche Formatierung.

Im folgenden Abschnitt wird der Korpus von manuell überprüften und gekennzeichneten Edits an Wikipedia-Artikeln vorgestellt, der im Zuge dieser Arbeit erstellt wurde. Die 301 im Korpus enthaltenen Vandalismusfälle werden entsprechend der eingeführten Kategorisierung eingeteilt und diskutiert.

### 2.3.1 Korpus von Edits an Wikipedia-Artikeln

Damit die verschieden Erkennungsansätze miteinander verglichen werden können, ist ein Korpus von Edits an Wikipedia-Artikeln notwendig, bei dem für jeden Edit manuell geprüft und gekennzeichnet wurde, ob es sich um Vandalismus handelt oder nicht. Da es bislang keinen Korpus gibt, der die gewünschten Anforderungen erfüllt, wurde im Rahmen dieser Arbeit ein Korpus für diesen Zweck erstellt. Der Korpus wurde WEBIS-VC07-11 getauft und ist frei zugänglich [32].
Die Basis des Korpus bilden die Daten der *Study 1* [11]. In dieser Studie ist Vandalismus mit 31 von 669 Edits nur sehr schwach vertreten. Damit Vandalismus stärker im Korpus repräsentiert wird, wurden weitere 270 Fälle von Vandalismus manuell überprüft

| **Edit Typ** | **# Edits** | **# Edititerungsstellen** | | |
|---|---|---|---|---|
| | | Einfügen | Ersetzen | Löschen |
| konstruktiv | 639 | 349 | 327 | 182 |
| Vandalismus | 301 | 158 | 146 | 98 |
| Summe | 940 | 507 | 473 | 280 |

Tabelle 2.2: Aufteilung des Korpus. Die zweite Spalte enthält die Angaben bezogen auf einen Edit. Die Spalten drei bis fünf hingegen beziehen sich auf einzelne Editierungsstellen, aufgeschlüsselt nach den Manipulationsformen. Ein Edit kann mehrere Editierungsstellen beinhalten (Vgl. 2.3).

| **Form der Manipulation** | **veränderter Inhalt** | | | | |
|---|---|---|---|---|---|
| | Text | Link intern | Link extern | Media | Formatierung |
| Einfügen | 43,9 % (83,5 %) Charakteristik: Anstößig, themenfremd, Nonsens, bewertend, Duplikate, Kauderwelsch | 5,3 % (10,1 %) | 6,9 % (13,3 %) | 0,7 % (1,3 %) | 14,6 % (27,8 %) Charakteristik: Formatierung, Hervorhebung |
| Ersetzen | 45,8 % (94 %) | 14,3 % (29,5 %) | 4,7 % (9,6 %) | 2 % (4,1 %) | 15,5 % (31,5 %) |
| Löschen | 31,6 % (96,9 %) | 28,2 % (86,7 %) | 22,9 % (70,4 %) | 19,4 % (61,2 %) | 20,3 % (62,2 %) |

Tabelle 2.3: Diese Matrix organisiert Vandalismus entlang der beiden Dimensionen „Manipulationsform“ und „veränderter Inhalt“. Für die im Korpus enthaltenen Vandalimus Edits ist die prozentuale Verteilung bezogen auf die Gesamtheit aller Vandalismus-Edits (schwarz) und bezogen auf die Vandalismus-Edits für die jeweiligen Manipulationsformen (grau) aufgeführt.

und hinzugefügt. Die Tabelle 2.2 zeigt die Verteilung von konstruktiven und destruktiven Edits im Korpus und die Präsenz der Manipulationsformen Einfügen, Ersetzen und Löschen.
Details zur praktischen Organisation des Korpus sind in Anhang A.1 zu finden.

Die Einteilung der 301 im Korpus enthaltenen Vandalismusfälle anhand der beiden Kriterien „Manipulationsform“ und „veränderter Inhalt“ zeigt Tabelle 2.3.
Die Vandalismus-Matrix (Tabelle 2.3) gibt Hinweise, wie Vandalismus in Wikipedia begangen wird. Wie zu erwarten, ist der textuelle Inhalt eines Artikels am häufigsten von Vandalismus betroffen und dies unabhängig von der Manipulationsform, wie die grau gehaltenen Prozentangaben der „Text“ Spalte belegen. Deshalb fokussiert der in dieser Arbeit vorgestellte Ansatz für die automatischen Erkennung von Vandalismus auf textuellen Inhalt. Vandalismus, der primär Medieninhalte und Links betrifft, bleibt unberücksichtigt.
Warum entfällt ein Großteil des Vandalismus auf die „Text“ Kategorie? Diese Tatsache ist damit zu begründen, dass es sich um Lexikonartikel handelt und diese überwiegend aus textuellem Inhalt bestehen. Darüber hinaus stellt die Manipulation des Textes aber auch die geringsten Anforderungen an den Vandalen und führt somit am schnellsten zum Ziel. So wird der Text analog zum Umgang mit bekannten Texteditoren bearbeitet und das Ergebnis ist nach dem Speichern sofort online sichtbar.
Eine weitere Auffälligkeit zeigt sich bei der Betrachtung der einzelnen Manipulationsformen. So ist die Dominanz von textuellem Inhalt als Gegenstand von Vandalismus

für das Löschen weniger stark ausgeprägt, als beim Einfügen und Ersetzen. Es liegt die Vermutung nahe, dass das Löschen größerer Teile des Artikels die vorrangige Form von Löschvandalismus ist. Dabei ist nämlich die Wahrscheinlichkeit größer auch nicht textuelle Inhalte, wie z. B. Bilder, mit zu löschen, die nur ein Minimum der Gesamtsubstanz eines Artikels ausmachen. Eine, innerhalb von Wikipedia bekannte, Form von Löschvandalismus ist das so genannte Page-Blanking. Dabei wird der komplette Inhalt eines Artikels gelöscht und gegebenenfalls durch ein kurzes Statement, z. B. beleidigende Äußerungen, ersetzt. Definiert man Page-Blanking als die Reduktion um mindestens 90 % des Inhaltes, so macht diese Form des Vandalismus ca. 42 % des Löschvandalismus innerhalb des Korpus aus. In Kombination mit der Tatsache, dass in ca. 62 % der Fälle von Löschvandalismus Formatierung und Medieninhalte (mit)betroffen sind, lässt sich folgern, dass bei ca. 20 % des Löschvandalismus ein größerer Abschnitt des Artikels (z. B. ein Absatz) gelöscht wird, wovon natürlich auch häufig Bilder und sonstiges Material betroffen sind.

### 2.3.2 Beispiele von Vandalismus

Zur Veranschaulichung werden abschließend einige Beispiele von Vandalismus präsentiert. Alle aufgeführten Beispiele sind Bestandteil des erstellten Korpus. Eine Liste mit den zugehörigen URL's ist im Anhang A.2 zu finden.

*Artikel*: Banana
*Kategorie*: [Ersetzen, Text]
*Charakteristik*: Kauderwelsch

- *Vorher*: „Banana is the common name used for herbaceou plants of the genus Musa, and is also the name given to the fruit of these plants. ...“

- *Nachher*: ”n cjmdftujrhdvgfuifjvdr tghjukidtm’ is the common name used for herbaceou plants of the genus Musa, and is also the name given to the fruit of these plants. ...‘“

*Artikel*: African American
*Kategorie*: [Ersetzen, Text] und [Ersetzen, Formatierung]

*Charakteristik*: anstößig, Formatierung

- *Vorher*: „... Überschrift: Early history ...“
- *Nachher*: „... Überschrift: NIGGER SCUM: THE LFIE AS A SLAVE ...“

*Artikel*: Google
*Kategorie*: [Einfügen, Text]
*Charakteristik*: Nonsens, Textduplikate

- *Vorher*: „... Google was co-founded by Larry Page and Sergey Brin while they were students at Stanford University, and the company was first incorporated as a privately held company on September 7 1998. ...“
- *Nachher*: „... Google was co-founded by Sam Harris from hitchin hertfordshire is rely rely rely rely rely rely rely good looking the best lookingest person on the world Larry Page and Sergey Brin while they were students at Stanford University, and the company was first incorporated as a privately held company on September 7 1998. ...“

*Artikel*: Dog
*Kategorie*: [Einfügen, Text]
*Charakteristik*: Nonsens

- *Vorher*: „...and Coat (dog) coats can be anything from very short to several centimeters long, from coarse hair to something akin to wool, straight or curly, or smooth. ...“
- *Nachher*: „... and Coat (dog) coats can be anything from very short to several centimeters long, from coarse hair to something akin to wool, straight or curly, or smooth. Many people believe that Michael Lee eats them....Its true ...“

*Artikel*: Borat
*Kategorie*: [Einfügen, Text]
*Charakteristik*: persönliche Meinung / Wertung

- *Vorher*: „... Over the course of the film, Borat falls in love with Pamela Anderson after watching a re-run of "Baywatch,änd vows to make her his wife. ...“

- *Nachher*: „... Over the course of the film, Borat falls in love with Pamela Anderson after watching a re-run of "Baywatch,änd vows to make her his wife. Although exposing the show to millions of viewers, the movie lacks the subtle intellectual humour of the the UK and HBO Television sketches. ...“

*Artikel*: Klondike (solitaire)
*Kategorie*: [Einfügen, Text]
*Charakteristik*: themenfremd

- *Vorher*: „... Klondike is a solitaire card game often known purely by the name of Solitaire. It is probably the most well known solo card game. ...“

- *Nachher*: „... Welcome to this article. You can read it if you want. If you do not want it, look at some other article. Klondike is a solitaire card game often known purely by the name of Solitaire. It is probably the most well known solo card game. ...“

*Artikel*: Bill Gates
*Kategorie*: [Löschen, (Text, Formatierung, Link, Media)], [Einfügen, (Text, Formatierung)]
*Charakteristik*: anstößig, plakativ

- Der kompletter Artikel wurde ersetzt durch: „CUNT“

Das letzte Beispiel belegt, dass Vandalismus durchaus auch in Mischformen auftreten kann und die verschiedenen Unterteilungen sich nicht gegenseitig ausschließen.

## 2.4 Aufspüren und Beseitigen von Vandalismus

Wird ein Fall von Vandalismus durch einen Benutzer Wikipedias entdeckt, ist es für diesen ein Leichtes, den Schaden zu beheben. Er braucht nur in der Artikel History eine Revision zu suchen, auf die er den Artikel zurücksetzen möchte und macht diese Revision durch einen einzigen Knopfdruck wieder zur aktuellen Version des Artikels. Die Gemeinschaft schlägt für einen solchen Fall vor, dass im Kommentar des Edits einen Vermerk, wie etwa „revert vandalism“ (kurz rvv), hinterlassen werden sollte. Dadurch wird auf die Beseitigung von Vandalismus hingewiesen und es entsteht nicht der fälschliche Verdacht eines Vandalimusaktes. Des Weiteren sollte dem Vandalen, sofern es sich um einen registrierten Benutzer handelt, eine Mahnung auf dessen Benutzerseite hinterlassen werden. Sollte der Benutzer weiterhin als Vandale auffällig sein, wird geraten, ihn bei den Administratoren von Wikipedia zu melden, was zur Sperrung seines Accounts und/oder seiner verwendeten IP führen kann.
Neben den Mechanismen, die durch die Wiki-Software selbst zur Verfügung gestellt werden, gibt es weitere Werkzeuge, die den Benutzer im Kampf gegen Vandalismus unterstützen. Im Folgenden werden einige dieser Tools vorgestellt.

**Tools zum Überwachen der Recent-Changes** Schnell auf Vandalismus aufmerksam zu werden ist die Grundvorraussetzung, um diesen rasch beheben zu können und so den Schaden für das Lexikon in Grenzen zu halten. Das Wiki-System selbst bietet dem registrierten Benutzer z. B. die Möglichkeit, einen Artikel auf seine persönliche Beobachtungsseite zu setzen. Auf dieser Seite werden alle Edits der eingetragenen Artikel aufgelistet, so dass der Benutzer alle Veränderungen an für ihn interessanten Artikeln auf einen Blick erfassen kann. Darüber hinaus gibt es sozusagen eine globale Beobachtungsseite, die Recent-Changes (kurz RC)[9], auf der die letzten 500 getätigten Edits des gesamten Wikis aufgeführt werden.
Es existieren diverse Werkzeuge, welche die beiden Punkte, 1. Vandalismus aufspüren durch Überwachen der RC's und 2. Vandalismus beseitigen durch Zurücksetzen, bündeln und vereinfachen. Diese Tools unterstützen den engagierten Benutzer beim Überwachen und Zurücksetzen, sind jedoch nicht in der Lage, selbstständig Vandalismus zu entdecken. Tabelle 2.4 zeigt eine Auswahl dieser Programme.

[9] http://en.wikipedia.org/wiki/Special:Recentchanges, Letzter Zugriff: 11.12.2007

| Tool | URL |
|---|---|
| VandalProof | http://en.wikipedia.org/wiki/User:AmiDaniel/VandalProof |
| VandalFigther | http://en.wikipedia.org/wiki/User:Henna/VF |
| Lupin's Anti-Vandal Tool | http://en.wikipedia.org/wiki/User:Lupin/Anti-vandal_tool |
| WikiMonitor | http://meta.wikimedia.org/wiki/WikiMonitor |
| Vandal Sniper | http://en.wikipedia.org/wiki/WP:SNIPE |
| WikiGuard | http://en.wikipedia.org/wiki/WP:WikiGuard |
| ShadowTool | http://en.wikipedia.org/wiki/User:Shadow1/ShadowTool |
| Mike's Wiki Tool | http://en.wikipedia.org/wiki/Wikipedia:MWT |
| Wikipedia Vandalism Watch | http://en.wikipedia.org/wiki/WP:WVW |

Tabelle 2.4: Übersicht über Tools, die den Benutzer Wikipedias beim Überwachen der RC's und beim Zurücksetzens von entdecktem Vandalismus unterstützen (Stand: 25.10.2007). Detaillierte Information sind in [8] zu finden.

**Zuordnen von anonymen Edits** Eine sehr subtile Art des Vandalismus ist es, die Popularität von Wikipedia zu nutzen, um das Meinungsbild über eine bestimmte Sache zu beeinflussen. Dies geschieht häufig im Interesse einer bestimmten Gruppierung (z. B. einer Partei, einer Organisation oder eines Unternehmens), welche durch das Beschönigen bestimmter Artikel die Meinung der Leser in eine für sie vorteilhafte Richtung lenken will.

Derartigen Lobbyismus zu identifizieren ist oft sehr schwierig, da er auf eine sehr subtile Art und Weise in den Artikel eingebettet wird. Es gibt ein Tool namens Wikiscanner [13], welches helfen kann, diese Form von Vandalismus aufzudecken. Der Wikiscanner löst die IP-Adressen anonymer Edits auf und ordnet ihnen Domains und die zugehörigen Domainbesitzer zu. So lässt sich nachvollziehen, aus welcher Arbeitsumgebung bzw. aus welchem Umfeld ein vermeintlich anonymer Edit getätigt wurde. Unter Berücksichtigung dieser Information kann der Inhalt eines Edits nochmals kritisch hinterfragt werden.[10]

**Urheber einer bestimmten Textstelle finden** Ein weiteres Tool, das sich sehr gut mit dem Wikiscanner ergänzt, ist WhodunitQuery [2]. Dieses Werkzeug ermöglicht es, für eine markierte Phrase im Text automatisch herauszufinden, durch welchen Edit bzw. Benutzer diese Textstelle in den Artikel eingefügt wurde. Um dies zu realisieren, durchsucht das Tool die History des Artikels nach dem ersten Auftreten des markierten Textes. Sollte das Ergebnis dieser Suche ein anonymer Edit sein, könnte die IP wiederum als

[10] Eine Auflistung von manipulierten Beiträgen in der deutschen Wikipedia, die mit Hilfe des Wikiscanners gefunden wurden, findet man unter http://de.wikipedia.org/wiki/Wikipedia:Wikiscanner. Letzter Zugriff: 01.10.2007

Ausgangspunkt für eine Analyse mit dem Wikiscanner dienen.

## 2.5 Regelbasierte Erkennung durch Bots

Auf dem Datenbestand von Wikipedia arbeiten kleine, autonome Programme oder Scripte, so genannte Bots, welche meist wiederkehrende Fleißaufgaben übernehmen, um die Gemeinschaft der Benutzer zu entlasten und zu unterstützen. So gibt es z. B. einen Bot, der externe Links in Wikipedia-Artikeln auf deren Erreichbarkeit prüft und eine Liste mit „kaputten“ externen Links pflegt, so dass ein Benutzer diese nochmals verifizieren und gegebenenfalls entfernen kann[11]- oder Bots, die Artikel des gleichen Themas in unterschiedlichen Sprachversionen verlinken[12].
Darüber hinaus gibt es auch Bots, die autonom nach Vandalismus suchen und diesen entweder in bestimmten IRC[13]-Channels melden und/oder den entdeckten Vandalismus beheben, indem sie den Artikel auf die vorherige Revision zurücksetzen. Die Gruppe dieser Bots wird im weiteren Verlauf der Arbeit als Anti-Vandalismus-Bots bezeichnet. Die Tabelle 2.5 gibt einen Überblick über verschiedene Anti-Vandalismus-Bots.

| Bot | Aktion | Status | URL |
|---|---|---|---|
| AntiVandalBot | Zurücksetzen | inaktiv | http://en.wikipedia.org/wiki/User:AntiVandalBot |
| MartinBot | Zurücksetzen, Melden | inaktiv | http://en.wikipedia.org/wiki/User:MartinBot |
| T-850RoboticAssistant | Zurücksetzen | aktiv | http://en.wikipedia.org/wiki/User:T-850_Robotic_Assistant |
| WerdnaAntiVandalBot | Zurücksetzen | aktiv | http://en.wikipedia.org/wiki/User:WerdnaAntiVandalBot |
| Xenophon | Zurücksetzen | aktiv | http://en.wikipedia.org/wiki/User:Xenophon_(bot) |
| PgkBot | Melden | aktive | http://meta.wikimedia.org/wiki/CVN/Bots#pgkbot |
| MiszaBot | Zurücksetzen | aktiv | http://en.wikipedia.org/wiki/User:MiszaBot |
| SWMTBot | Melden | aktiv | http://meta.wikimedia.org/wiki/CVN/Bots#SWMTBot |
| ClueBot | Zurücksetzen | aktiv | http://en.wikipedia.org/wiki/User:ClueBot |
| CounterVandalismBot | Zurücksetzen | aktiv | http://en.wikipedia.org/wiki/User:CounterVandalismBot |

Tabelle 2.5: Übersicht über Anti-Vandalismus-Bots (Stand: 25. Oktober 2007). Grau gehaltene Bots arbeiten mit dem selben oder ähnlichen Erkennungsregeln bzw. Quellcode (Clone) wie der zuletzt in schwarz aufgeführte Bot. Stuft ein Bot einen Edit als Vandalismus ein, setzt er den Artikel auf die vorherige Version zurück und/oder meldet den verdächtigen Edit in einem IRC-Channel.

[11] http://en.wikipedia.org/wiki/User:Ocobot
[12] z. B. http://en.wikipedia.org/wiki/User:YurikBot, Letzter Zugriff: 13.09.2007
[13] IRC, Kurzform von Internet Relay Chat: Ein Online-Chatsystem, bei dem sich mehrere Benutzer in so genannten Channels treffen, um sich zu unterhalten.

Die momentan aktiven Anti-Vandalismus-Bots arbeiten mit von Menschen festgelegten Regeln, um konstruktive von destruktiven Edits zu unterscheiden. Das bedeutet, der Bot hat eine Menge von Regeln, die er überprüft und wenn eine oder mehrere Regeln zutreffen, wird der untersuchte Edit als Vandlismus eingestuft.
Eine solche Regel könnte z. B. lauten: *„Wenn die Differenz des Umfangs von alter und neuer Revision mehr als k Byte beträgt, dann ist der Edit Vandalimus“*. Eine mögliche Begründung für eine solche Regel könnte sein: *„Ein massenhaftes Löschen von Inhalt liegt vor und man kann somit von einer destruktiven Veränderung, also Vandalismus ausgehen“*. Den Schwachpunkt dieser Vorgehensweise macht das Beispiel der formulierten Regel gut deutlich: Welcher Wert sollte für $k$ gewählt werden?
Dieses Problem löst maschinelles Lernen auf elegante Art und Weise. Dort wird der zu wählende Wert $k$ aus einer Vielzahl von Beispiel-Edits gelernt. Dies ist möglich, da sowohl die Werte für $k$, als auch die Zuordnung, ob es sich um Vandalismus handelt oder nicht, für jedes Beispiel bereitgestellt wird.
Ein weitere Vorteil des maschinellen Lernens ist, dass auch komplexe Entscheidungsstrukturen gelernt werden können. Hingegen genügt es den Anti-Vandalismus-Bots, aufgrund der disjunktiven Verknüpfung der Regeln, wenn eine Regel erfüllt ist, um einen Edit als Vandalismus einzustufen.

Für den Vergleich der Erkennungsleistung von Lernverfahren und Anti-Vandalismus-Bots wurden zwei Bots stellvertretend ausgewählt: ClueBot [5] und AntiVandalBot (kurz AVB) [27].

Beide Bots nutzen vordefinierte Regeln, die darauf ausgelegt sind, verschiedene Charakteristiken von Vandalismus in Wikipedia zu erfassen. Teilweise zielen mehrere Regeln auf

| **Charakteristik** | **# AVB** | **# ClueBot** |
|---|---|---|
| Löschen kompletter Artikel | 1 | 1 |
| Löschen großer Teile des Artikels | 3 | 1 |
| Wiederholung eines Zeichens | 3 | - |
| Wiederholung einer Zeichensequenz | 1 | - |
| vulgärer Inhalt | 5 | 2 |
| Text Formatierung | 3 | - |
| Natürlichsprachlichkeit | 1 | - |
| Aufblähen des Artikels durch Leerzeichen | 1 | - |
| Inhalt des Kommentares | - | 1 |
| Summe | 18 | 4 |

Tabelle 2.6: Quantitative Auflistung der von AntiVandalBot und ClueBot verwendeten Regeln.

eine einzelne Eigenschaft von Vandalismus ab. AntiVandalBot arbeitet mit 18 Regeln, ClueBot hingegen nur mit 4. Tabelle 2.6 listet die Regeln der beiden Bots auf.

Ein Vergleich mit dem zum Zeitpunkt der Arbeit neuen CounterVandalismBot [19] konnte nicht durchgeführt werden, da dieser nicht quelloffen ist und sein Betreiber auf unsere Anfragen nicht reagierte.
Ähnlich verhält es sich mit PkgBot [21], der zwar quelloffen ist, aber intern mit einer Datenbank arbeitet. Den Bestand der Datenbank konnten wir trotz unserer Bemühungen nicht in Erfahrung bringen.

# 3 Vandalismuserkennung mit Hilfe maschineller Lernverfahren

Dieses Kapitel erläutert, wie Vandalismus in Wikipedia mit Hilfe maschinellen Lernens erkannt werden kann.
Im ersten Teil dieses Kapitels werden theoretische Grundlagen des maschinellen Lernens vermittelt, wobei speziell auf das überwachte Lernen eingegangen wird, da die Vandalismuserkennung in diesem Bereich angesiedelt ist. Genauer gesagt handelt es sich um ein so genanntes One-Class-Klassifikationsproblem, dessen Besonderheiten diskutiert werden. Abschließend wird die Regressionsanalyse, stellvertretend für verschiedene statistische Verfahren des überwachten Lernens, erörtert.
Den zweiten Teil dieses Kapitels bildet die Vorstellung von Features, die charakteristische Eigenschaften von Vandalismus quantifizieren.

## 3.1 Vandalismuserkennung als Klassifikationsproblem

Bekommt ein Mensch einen Stapel von Fotos, bei dem auf jedem Bild entweder ein Hund oder eine Katze zu sehen ist, ist er ohne weiteres in der Lage, diesen Stapel nach Hunde- und Katzenbildern zu sortieren. Diese Aufgabe wird auch als Klassifikationsproblem bezeichnet, wobei die Tierfotos als Objekte aufgefasst werden, denen die Klassen „Hundebild“ oder „Katzenbild“ zuzuweisen sind. Allgemein wird die Zuordnung von Objekten zu Klassen als Klassifikationsproblem bezeichnet.

Ein Klassifikationsproblem lässt sich wie folgt formalisieren [25]: $O$ bezeichnet eine Menge von Objekten, die durch die Zuordnung $\gamma : O \to C$ auf eine Menge von Klassen $C$ abgebildet werden. Die Zuordnung $\gamma$ entspricht der tatsächlichen Klasse der Objekte in

der realen Situation und wird auch als idealer Klassifikator für $O$ bezeichnet. Klassifizieren bedeutet die Klasse $\gamma(o) \in C$ für ein gegebenes Objekt $o \in O$ zu bestimmen.

Das einleitende Beispiel ist sicher ein Leichtes für einen Menschen. Er unterscheidet Hunde und Katzen anhand der Gestalt, der Größe, der Form der Ohren und Schnauze. In der Regel dient nicht eine einzelne dieser Eigenschaften als Entscheidungskriterium, sondern eine Kombination all dieser und weiterer beobachtbarer Eigenschaften. Um automatisch Objekte klassifizieren zu können, müssen die beobachtbaren Eigenschaften quantifiziert und sinnvoll miteinander kombiniert werden. Der nächste Abschnitt erläutert, wie dafür vorgegangen wird.

### 3.1.1 Überwachtes Lernen

Nach Mitchell [20] lernt ein Programm, wenn es anhand von Beispielen Regeln extrahieren kann, die anschließend auf zuvor ungesehene Objekte angewendet werden können, um sie zu klassifizieren.
Anders ausgedrückt: Ziel des überwachten Lernens ist es, ein Programm in die Lage zu versetzen, anhand von beobachtbaren Eigenschaften eines Objektes eine unbeobachtbare Eigenschaft, seine Klasse, vorherzusagen und zwar aufgrund des zuvor erlernten Zusammenhanges von beobachtbaren und unbeobachtbarer Eigenschaft.

Da die tatsächliche Zuordnung $\gamma : O \to C$ in einer realen Situation unbekannt ist, wird beim überwachten Lernen versucht diese Funktion zu approximieren. Dazu wird ein Modell der realen Situation erstellt. In diesem Modell werden die Objekte $o \in O$ nicht in ihrer Gesamtheit betrachtet, sondern durch eine Menge von beobachtbaren Eigenschaften repräsentiert. Die Eigenschaften werden von Eigenschaftsvektoren zusammengefasst, so dass die Objekte $o \in O$ zu Eigenschaftsvektoren $\mathbf{x} = \alpha(o)$ abstrahiert werden. Im Idealfall sind die ausgewählten Features ausreichend diskriminierend zwischen den Klassen, so dass sich die Eigenschaftsvektoren $\mathbf{x} = \alpha(o)$ im Raum, den die Eigenschaften aufspannen, zu klassenspezifischen Regionen gruppieren. In Abbildung 3.1 ist diese Situation schematisch dargestellt.

Um die tatsächliche Klassifikation $\gamma : O \to C$ durch eine Zuordnung $c : X \to C$, auf Basis der Eigenschaftsvektoren $\mathbf{x}$ zu approximieren, wird ein Menge von Beispielen

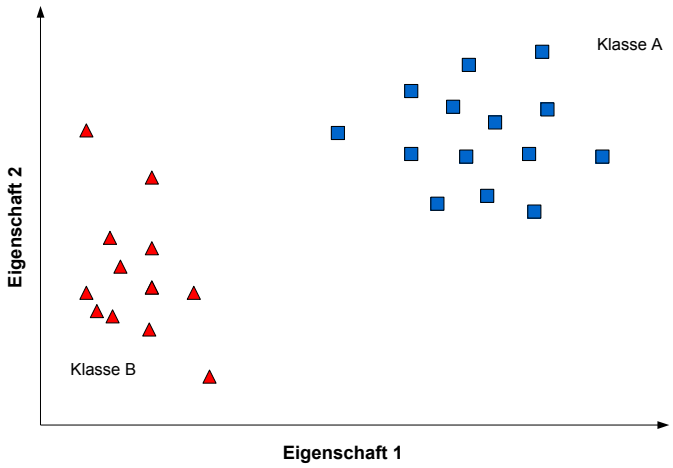


Abbildung 3.1: Die beiden Klassen A und B werden durch zwei Eigenschaften repräsentiert. Die Eigenschaftsvektoren der Objekte gruppieren sich in klassenspezifischen Regionen des Eigenschaftsraumes.

$(\alpha(o), \gamma(o))$ bzw. $(\mathbf{x}, c(\mathbf{x}))$ benötigt. Für die Beispiele in der Menge mit $\mathbf{x} = \alpha(o)$ ist $c(\mathbf{x})$ definiert als $\gamma(o)$. Das bedeutet, für jedes Beispiel in der Menge ist sowohl die Vektorrepräsentation $\mathbf{x}$ als auch die tatsächliche Klassenzugehörigkeit $\gamma(o)$ bekannt. Die Menge der Beispiele wird auch als Trainingsmenge bezeichnet, da das Lernverfahren damit trainiert wird. Der in Abschnitt 2.3.1 vorgestellte Korpus ist die Trainingsmenge des Klassifizierers, der trainiert wird, Vandalimus-Edits zu erkennen.

Beim überwachten Lernen wird der Zusammenhang zwischen den beobachtbaren Eigenschaften und der Klassenzugehörigkeit, also $c : \mathbf{X} \to C$, aus den Beispielen der Trainingsmenge gelernt. Um dies zu erreichen gibt es verschiedene Möglichkeiten. Das Verfahren, das wir für die Vandalismuserkennung einsetzen, versucht die Objekte im Eigenschaftsraum durch eine Ebene zu trennen, so dass Objekte die auf der einen Seite der Ebene liegen Klasse A, und die auf der anderen Seite Klasse B angehören. Abbildung 3.2 zeigt schematisch die Separierung zweier Klassen A und B, durch eine von vielen möglichen Trennungsebenen. Ist der Eigenschaftsraum hochdimensional, wird die Trennungsebene auch als Hyperebene bezeichnet. Das Annäheren von $c : \mathbf{X} \to C$ an $\gamma : O \to C$ entspricht der Suche im Eigenschaftsraum nach einer Hyperebene, welche die Eigenschaftsvektoren der Objekte, ihren Klassen entsprechend, optimal voneinander trennt. Deshalb wird versucht die Eigenschaften so zu wählen, dass die Objekte der verschiedene Klassen im

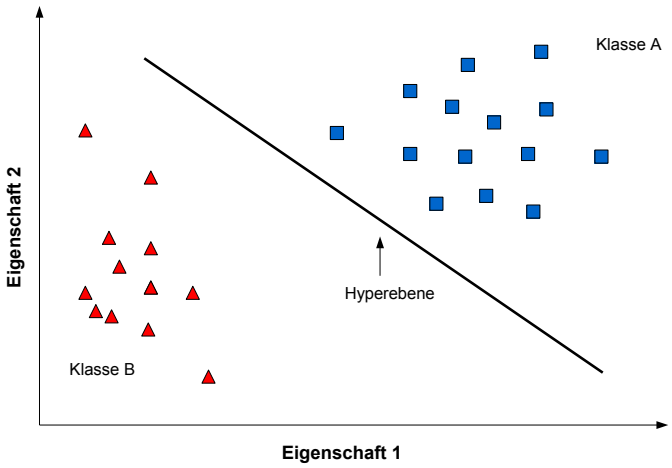


Abbildung 3.2: Die durchgezogene Linie ist die Hyperebene, welche die klassenspezifischen Regionen voneinander trennt. Diese oder eine vergleichbare Ebene gilt es durch das Training auf der Trainingsmenge zu bestimmen.

Eigenschaftsraum linear separierbar sind.
Die approximierte Zuordnung $c : \mathbf{X} \rightarrow C$ wird auch als gelernter oder trainierter Klassifikator für $\mathbf{X}$ bezeichnet.

### 3.1.2 Vandalismuserkennung als One-Class-Klassifikationsproblem

Die automatische Erkennnug von Vandalimus in Wikipedia ist ein Klassifikationsproblem mit zwei Klassen, konstruktiven und destruktiven Edits, zu dessen Lösung wir überwachtes Lernen einsetzen.

Die Objekte $O$, die es zu klassifizieren gilt sind die Edits $E = \{e_1, \ldots, e_{|E|}\}$, wobei ein Edit $e$ zwei aufeinanderfolgende Revisionen des selben Wikipedia-Artikels $a$ darstellt $e = \langle a_{old}, a_{new} \rangle$.
Genauer gesagt definieren wir die Vandalismuserkennung als ein One-Class-Klassifikationsproblem. Die One-Class-Klassifikation ist eine Spezialform der Klassifikation. Entgegen der Namensgebung existieren auch bei der One-Class-Klassifikation zwei Klassen, jedoch werden Besonderheiten der beiden Klassen berücksichtigt: Die Verteilung der Klassen in der realen Situation und wie gut sich die Klassen anhand von beobachtba-

ren Eigenschaften beschreiben lassen, haben Einfluss auf die Definition der Klassen als Zielklasse und als Klasse der Außenliegenden [28]:

- *Zielklasse*: Die Zielklasse steht im Fokus des Interesses. Sie gilt es anhand beobachtbarer Eigenschaften möglichst genau zu spezifizieren, weshalb im Zusammenhang mit One-Class-Klassifikation oft von Datenbeschreibung gesprochen wird. Die Beispiele der Zielklasse in der Trainingsmenge sollten repräsentativ für diese Klasse von Objekten sein. Dies ist notwendig, um ein Schema zu erlernen, durch das sich Objekte der Zielklasse von anderen Objekten abgrenzen lassen.

- *Klasse der Außenliegenden*: Die Objekte, die nicht in das Schema der Zielklasse passen, und somit außerhalb der Beschreibung der Zielklasse liegen, bilden die Klasse der Außenliegenden.

Ein Klassifizierer wird auf den beschreibenden Eigenschaften der Zielklasse trainiert, um Objekte dieser Klasse möglichst präzise erkennen zu können. Es wird eine möglichst genaue Beschreibung der Zielklasse angestrebt, um die Frage „Gehört das Objekt zur Zielklasse?“ - mit ja oder nein beantworten zu können. Lautet die Antwort nein, ist darüber hinaus nicht von Bedeutung, was genau denn nun eigentlich das Objekt ist. Wichtig ist nur, dass es nicht zur Zielklasse gehört. Abbildung 3.3 veranschaulicht die Problematik der One-Class-Klassifikation. Im Gegensatz dazu stellt sich bei der klassischen Klassifikationsaufgabe die Frage, „Zu welcher Klasse gehört das Objekt?“.

Die Gründe für eine Einteilung in Zielklasse und Klasse der Außenliegenden sind in der Praxis oft pragmatischer Natur. Vorraussetzungen für die Zielklasse sind, dass ausreichend Beispiele für die Trainingsmenge vorhanden sind und diese Beispiele so gut wie möglich das Spektrum der Objekte innerhalb dieser Klasse repräsentieren. Darüber hinaus ist wichtig, die Zielklasse ausreichend genau durch die beobachtbaren Eigenschaften spezifizieren zu können. Demgegenüber stehen die Außenliegenden. Für diese Klasse kann kein Muster gelernt werden.
In der Praxis liegt dies meist daran, dass für die Klasse der Außenliegenden keine, oder nur wenige repräsentative Beispiele zur Verfügung stehen, welche eine Trainingsmenge bilden könnten. Ebenso gibt es Situationen, in denen zu viele Beispiele für die Außenliegenden existieren und es schwierig, wenn nicht gar unmöglich ist, spezifische Charakteristiken der Klasse zu erkennen und die Klasse dadurch zu beschreiben.

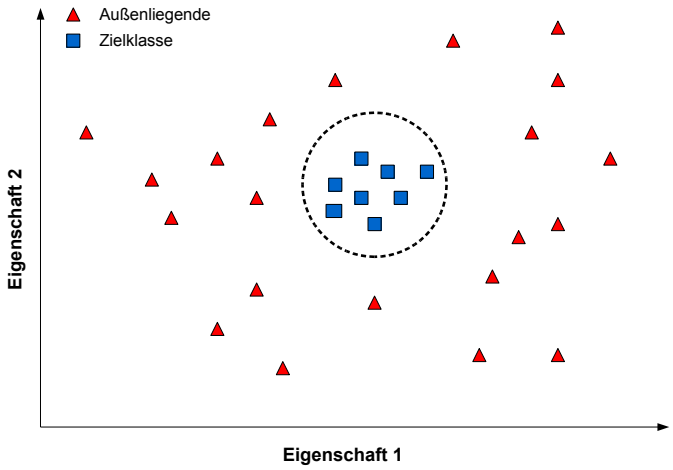


Abbildung 3.3: Die gestrichelte Linie symbolisiert die Beschreibung der Zielklasse auf Basis der beobachtbaren Eigenschaften. Die Beschreibung grenzt die Objekte der Zielklasse gegenüber den außenliegenden Objekten ab.

Bei der Erkennung von Vandalismus in Wikipedia bilden die destruktiven Edits die Zielklasse. Vandalismus-Edits weisen Charakteristiken auf, z. B. übermäßiger Einsatz von Großbuchstaben, massenhaftes Löschen, etc., die sie eindeutig gegenüber konstruktiven Edits abgrenzen. Die konstruktiven Edits bilden die Klasse der Außenliegenden, welche durch unzählige Vertreter überrepräsentiert ist.
Es gilt also Eigenschaften zu finden, die sehr spezifisch für Vandalismus-Edits sind und somit abgrenzend wirken. Eine Eigenschaft wird durch ein Feature $f$ quantifiziert. Ein Feature $f$ ist eine Funktion $f : E \to \mathbf{R}$, welche eine beobachtbare Eigenschaft eines Edits $e$ auf eine reelle Zahl abbildet. Die Menge der Eigenschaften $\mathcal{F}$, bezogen auf einen Edit $e$, kann dazu benutzt werden, diesen Edit in Form eines Vektors zu repräsentieren, $\mathbf{e} = (f_1(e), \ldots, f_{|\mathcal{F}|}(e))^T$. Folglich lässt sich die Menge der Edits $E$ ebenso durch die Menge der Vektorrepräsentationen $\mathbf{E}$ darstellen.
Anhand der Trainingsmenge $\mathbf{E}_V$ der Form $(e, \gamma(e))$ bzw. $(\mathbf{e}, c(\mathbf{e}))$ soll der Zusammenhang zwischen den Features, in Form der Vektorrepräsentation $\mathbf{e}$, und der Klassenzugehörigkeit $\gamma(e)$, eines Edits $e$ gelernt werden. Der trainierte Klassifizierer $c : \mathbf{E} \to \{1, 0\}$ wird dann dazu verwendet, für zuvor unbekannte Edits eine Aussage darüber zu treffen, ob diese zur Zielklasse gehören $c(\mathbf{e}) = 1$ oder nicht $c(\mathbf{e}) = 0$.

### 3.1.3 Klassifikation mittels Regression

Für die Approximation von $c : \mathbf{E} \rightarrow C$, existieren verschiedene mathematische Möglichkeiten. Die Regression oder auch Regressionsanalyse ist eines dieser geeigneten Verfahren. Bei der Regression handelt es sich um ein statistisches Verfahren, bei dem der Zusammenhang zwischen einer abhängigen Variablen, der so genannten Zielvariable $\mathcal{Y}$ und ein oder mehrerer unabhängigen Variablen, den so genannten Kovarariablen $\mathcal{X}_1, \ldots, \mathcal{X}_n$, untersucht wird.[1] Um diesen Zusammenhang zu modellieren, können unterschiedliche Funktionen verwendet werden. Im einfachsten Fall handelt es sich um eine lineare Funktion. Man spricht dann von einer linearen Regression. Die grundlegende Funktionsweise einer linearen Regression, wie sie in [14] beschrieben wird, sowie die Anwendung zur Klassifikation soll nachfolgend mit Hilfe der in Abschnitt 3.1.2 eingeführten Formalisierung erläutert werden.

Die Klassenzugehörigkeit $\gamma : E \rightarrow C$ entspricht in diesem Fall der abhängigen Variable $\mathcal{Y}$ und die Feature $f \in \mathcal{F}$ stellen die unabhängigen Variablen $\mathcal{X}_1, \ldots, \mathcal{X}_n$ dar. Die Zielklasse und die Klasse der Außenliegenden werden durch entsprechende Zahlenwerte repräsentiert $C = \{0, 1\}$. Um den Zusammenhang zwischen Zielvariable und Kovariablen zu modellieren, wird eine Linearkombination der Feature $f \in \mathcal{F}$ aufgestellt, wobei jedes Feature $f$ mit einem Faktor $w$ gewichtet wird. Die Faktoren $w$ werden im Kontext auch als Regressionskoeffizienten bezeichnet. Dies lässt sich durch folgende Formel ausdrücken:

$$\gamma(e) = w_0 + w_1 f_1 + \ldots + w_{|\mathcal{F}|} f_{|\mathcal{F}|} = w_0 + \sum_{j=1}^{|\mathcal{F}|} w_j f_j \tag{3.1}$$

Mit Hilfe der $n$ Trainingsbeispiele in $\mathbf{E}_V$, für die die tatsächliche Klassenzugehörigkeit $\gamma(e)$ bekannt ist, werden die Regressionskoeffizienten optimiert.

$$\gamma(e_i) = w_0 + w_1 f_{1i} + \ldots + w_{|\mathcal{F}|} f_{|\mathcal{F}|i} = w_0 + \sum_{j=1}^{|\mathcal{F}|} w_j f_{ji} \ , i = 1, \ldots, n \tag{3.2}$$

[1] In der Literatur sind weitere Bezeichnungen üblich. *Zielvariable*: Kriterium, Response-Variable, endogene Variable, Regressand - *Kovariablen*: Prädiktor-Variablen, exogene Variable, Regressoren.

Es sollen diejenigen Parameter $w_i$ ausgewählt werden, bei denen die Summe der Quadrate der Abweichungen ($RSS(w)$) zwischen vorhergesagter Klasse $c(\mathbf{e})$ und tatsächlicher Klasse $\gamma(e)$ minimal wird.

$$RSS(w) = \sum_{i=1}^{n} (\gamma(e_i) - c(\mathbf{e}_i))^2 \tag{3.3}$$

$$RSS(w) = \sum_{i=1}^{n} \left( \gamma(e_i) - w_0 - \sum_{j=1}^{|\mathcal{F}|} w_j f_{ji} \right)^2 \tag{3.4}$$

Um $RSS(w)$ in Formel 3.4 zu minimieren, ist die erste Ableitung nach $w$ notwendig. Das Gleichungssystem lässt sich äquivalent in kompakter Matrizenform angeben. Dabei ist $F$ eine $n \times (|\mathcal{F}|+1)$-Matrix, die pro Zeile einen Eigenschaftsvektor $\mathbf{e} = (f_1(e), \ldots, f_{|\mathcal{F}|}(e))^T$ (mit einer 1 an erster Stelle, wegen des Absolutgliedes $w_0$) der Trainingsmenge enthält. Analog ist $\gamma$ ein $n$-dimensionaler-Vektor $(\gamma(e_1), \ldots, \gamma(e_n))^T$ der die tatsächliche Klassenzugehörigkeit der $n$ Trainingsbeispiele enthält:

$$RSS(w) = (\gamma - wF)^T(\gamma - wF) \tag{3.5}$$

Die erste Ableitung von 3.5 nach $w$ lautet dann:

$$\frac{\partial RSS}{\partial w} = -2F^T(\gamma - wF) \tag{3.6}$$

Unter der Annahme, dass $F$ einen vollen Spaltenrang hat und $F^TF$ positiv definiert ist, wird die erste Ableitung Null gesetzt:

$$F^T(\gamma - wF) = 0 \tag{3.7}$$

Durch Umformen von Gleichung 3.7 ergibt sich die Lösung für den Vektor der Regressi-

onskoeffizienten $\hat{w}$ zu:

$$\hat{w} = (F^T F)^{-1} F^T \gamma \tag{3.8}$$

Die durch die Regression optimierte Funktion $\hat{c}(\mathbf{e}) = \hat{w}_0 + \sum_{n=1}^{|\mathcal{F}|} \hat{w}_n f_n$ kann nun für die Klassifikation $c : \mathbf{E} \to \{0, 1\}$ von unbekannten Edits benutzt werden.
Dazu bedarf es allerdings noch eines Zwischenschrittes, da die Funktion $\hat{c}(\mathbf{e})$ selbst kontinuierliche Werte zwischen 0 und 1 liefert und nicht direkt die vorhergesagte Klassenzugehörigkeit $c : \mathbf{E} \to \{0, 1\}$. Es wird ein Grenzwert $\mu \in [0, 1]$ z. B. $\mu = 0,5$ festgelegt, der für die Zuordnung zu einer Klasse mit dem errechneten Wert von $\hat{c}(\mathbf{e})$ verglichen wird:

$$c(\mathbf{e}) = \begin{cases} 1, & \hat{c}(\mathbf{e}) \geq \mu \\ 0, & \text{sonst} \end{cases} \tag{3.9}$$

Die optimierten Regressionskoeffizienten $(\hat{w_1}, \ldots, \hat{w_n})$ sind Normalenvektoren der Hyperebene und definieren sowohl die Form der Ebene, als auch deren Lage im $|\mathcal{F}|$-dimensionalen Eigenschaftsraum $\mathbb{R}^{|\mathcal{F}|}$.

Im Zuge der Experimente wurde eine Spezialform der Regression verwendet, die so genannte linear logistische Regression. Der Unterschied zur linearen Regression besteht darin, dass eine andere Fehlerfunktion zur Optimierung der Regressionskoeffizienten eingesetzt wird. Das Verfahren ist durch Hall und Frank in [17] und [26] beschrieben.

## 3.2 Quantifizierung von Eigenschaften von Vandalismus

Nachdem im vorherigen Abschnitt das Verfahren der Klassifikation erläutert wurde, folgt nun eine Beschreibung der dafür notwendigen Features $\mathcal{F}$. Vor dem Hintergrund, die Vandalismuserkennung als One-Class-Klassifikationsproblem einzustufen (siehe Abschitt 3.1.2), sind die einzelnen Feature darauf abgestellt, die Klasse der Vandalismus-Edits

möglichst genau zu beschreiben.

### 3.2.1 Auflistung der entwickelten Features

Für die Beschreibung der Vandalismus Zielklasse wurden bestimmte beobachtete Charakteristiken ausgewählt, die durch 16 verschiedene Features quantifiziert werden. Die Features analysieren einen Edit auf unterschiedlichen Abstraktionsebenen und lassen sich dementsprechend unterteilen.
Die unterste Ebene bildet die Zeichenebene, auf der die im Edit verwendeten Zeichen die Berechnungsgrundlage der Features bilden. Darüber steht die Termebene, auf der die Features die Terme des Edittextes und die des Artikels berücksichtigen. Die nächste Ebene ist die Artikelebene. Die dort angesiedelten Features betrachten den Artikel in seiner Gesamtheit. Den Abschluss bildet die Metaebene. Darauf befinden sich die Features, die sich Zusatzinformationen über einen Edit, wie z. B. den dazu hinterlegten Kommentar, zunutze machen, um den Featurewert zu berechnen.
Einen kompakten Überblick über die 16 entwickelten Features liefert die Tabelle 3.1.

Nachfolgend werden die einzelnen Features im Detail beschrieben. Begonnen wird auf der untersten, der Zeichenebene.

**Character-Distribution.** Dieses Feature quantifiziert die Abweichung der Buchstabenverteilung eines Edittextes von der erwarteten Buchstabenverteilung in natürlicher Sprache. Der Edittext ist eine Menge von Zeichen $\Sigma_e$. Eine Teilmenge der Zeichen des Edittextes sind Buchstaben $L_e \subset \Sigma_e$. Die Häufigkeitsfunktion $lf(l, e)$ gibt an, wie oft ein Buchstabe $l$ im Edit $e$ enthalten ist. Darüber hinaus ist $E(lf(l)))$ der sprachabhängige Erwartungswert für die Häufigkeit eines Buchstabens $l$. Für jeden Buchstaben $l$ wird die betragsmäßige Differenz von $lf(l, e)$ und $E(lf(l)))$ gebildet. Die einzelnen Differenzen werden aufsummiert. Je größer der aufsummierte Wert, desto größer die Abweichung des Edittextes von natürlicher Sprache.

$$f_i(e) = \sum |lf(l, e) - E(lf(l)))|\,,\ l \in L_e \tag{3.10}$$

| Feature $f$ | Kurzbeschreibung | sprachabhängig |
|---|---|---|
| **Zeichenebene** | | |
| Char-Distribution | Abweichung der relativen Zeichenhäufigkeiten des Edits vom sprachcharakteristischen Wert | x |
| Char-Sequence | Längste zusammenhängende Wiederholung eines Zeichens im Edit | - |
| Compressibility | Kompressionsrate des Edittextes | |
| Upper-Case-Ratio | Verhältnis von Großbuchstaben zu allen Buchstaben im Edit | - |
| **Termebene** | | |
| Term-Impact | Durchschnittliche relative Häufigkeit eines Terms des Edits in der neuen Artikelrevision | - |
| Longest-Word | Länge des längsten Wortes im Edit | - |
| Pronoun-Frequency | Verhältnis von Pronomen, der ersten und zweiten Person, zu allen Termen des Edits | x |
| Pronoun-Impact | Anteil der Edit Pronomen an der Anzahl der Pronomen der neuen Artikelrevision (nur Pronomen der ersten und zweiten Person) | x |
| Vulgarism-Frequency | Verhältnis von vulgären Worten zu allen Worten des Edits | x |
| Vulgarism-Impact | Anteil der vulgären Worte des Edits an der Anzahl der vulgären Worte der neuen Artikelrevision | x |
| **Artikelebene** | | |
| Size-Ratio | Verhältnis von alter zu neuer Revision, bezogen auf den Umfang | - |
| Replacement-Similarity | Ähnlichkeit von gelöschtem und dafür eingefügtem Text | x |
| Context-Relation | Ähnlichkeit von der alten Artikelrevision und Artikeln Wikipedias zu extrahierten Schlüsselwörtern des Edittextes | x |
| **Metaebene** | | |
| Anonymity | Gibt an, ob ein Edit anonym gemacht wurde | - |
| Comment-Length | Umfang des Kommentars zu einem Edit | - |
| Edits-Per-User | Anzahl früherer Edits durch den selben Benutzer oder IP | - |

Tabelle 3.1: Übersicht über die Features zur Quantifizierung der Charakteristiken von Vandalismus in Wikipedia.

Es wird deutlich, dass dieses Feature sprachabhängig ist, weil die erwarteten Buchstabenhäufigkeiten je nach Sprache unterschiedlich sind. Die verwendeten Erwartungswerte für die Buchstaben der englischen Sprache sind in Anhang A.4 aufgeführt. Eine Adaption auf andere Sprachen ist mit wenig Aufwand möglich.
Neben diesem Feature gibt es weitere sprachabhängige Features. Bei den folgenden Beschreibungen wird darauf explizit hingewiesen.

**Character-Sequence.** Dieses Feature zielt ebenso wie das Character-Distribution Feature auf das abnormale Sprachbild mancher Vandalismus-Edits. Es ist jedoch enger gesteckt, da es darauf ausgelegt ist, mehrfache Wiederholungen desselben Zeichens, wie

z. B. „!!!!!!!!!!!!“, zu quantifizieren. Dafür wird der Edittext sequentiell durchlaufen und so die längste zusammenhängende Wiederholung eines Zeichens $z \in \Sigma_e$ ermittelt. Der Grad $m$ der längsten zusammenhängenden Wiederholung bildet den Wert des Features.

$$f_i(e) = \max(m \cdot z),\ z \in \Sigma_e \tag{3.11}$$

**Compressibility.** Das dritte und letzte Feature, welches sich mit der Natürlichsprachlichkeit eines Edit $e$ befasst, ist das Compressibility Feature. Die Motivation dieses Features ist, dass sich nicht natürlichsprachlicher Text wie z. B. „lolololololololol“, aufgrund seiner Struktur besser komprimieren lassen sollte, als natürlichsprachlicher Text. Den Featurewert bildet die Komprimierungsrate, also das Verhältnis von komprimiertem zu unkomprimiertem Text.

$$f_i(e) = \frac{|compress(e)|}{|e|} \tag{3.12}$$

Hierbei steht $compress(e)$ für einen beliebigen Textkompressionsalgorithmus, der auf $e$ angewandt wird.

**Upper-Case-Ratio.** Dieses Feature arbeitet zwar ebenfalls auf der Zeichenebene, befasst sich aber im Gegensatz zu den drei vorherigen Features nicht mit der Sprache des Edittextes, sondern mit seiner Formatierung. Wie Tabelle 2.3 in Abschnitt 2.3 zu entnehmen ist, spielt plakative Formatierung bei ca. 28 % des Einfügevandalismus und ca. 32 % des Ersetzungsvandalimus eine Rolle. Die Erfahrung, die im Zuge dieser Arbeit gesammelt wurde zeigt, dass dabei überwiegend die Schreibweise in Großbuchstaben zum Einsatz kommt, wie z. B. „HELLO PARTY PEOPLE“. Diese Charakteristik wird durch das Upper-Case-Ratio Feature aufgegriffen. Der Wert des Features gibt das Verhältnis von Großbuchstaben, $U_e \subset \Sigma_e$, zu allen Buchstaben, $L_e \subset \Sigma_e$, des Edittextes an.

$$f_i(e) = \frac{|U_e|}{|L_e|} \tag{3.13}$$

Damit verlassen wir die Zeichenebene und kommen zu den Features, die auf der Termebene angesiedelt sind.

**Term-Impact.** Die Idee, die hinter diesem Feature steht, ist, eine Angabe darüber zu machen, inwieweit der im Edittext verwendete Wortschatz dem Vokabular des Artikels entspricht. Werden neue Begriffe eingeführt oder wird auf bereits im Artikel verwendetes Vokabular zurückgegriffen? Dieses Feature zielt auf Vandalismus bei denen themenfremde Inhalte eingefügt oder ersetzt werden. Ebenso ergeben sich charakteristische Featurewerte, wenn große Teile des Artikels gelöscht werden und dadurch Begriffe des Artikelwortschatzes mit verschwinden.
Der Text des Edits wird als eine Menge von Termen $T_e$ repräsentiert. Die Häufigkeitsfunktion $tf(t, e)$ gibt an, wie oft ein Term $t$ im Edit $e$ enthalten ist. Hingegen ermittelt $tf(t, a_{new})$, wie oft der Term $t$ in der neuen Revision des Artikels auftritt. Für jeden Term $t \in T_e$ wird das Verhältnis $\frac{tf(t,e)}{tf(t,a_{new})}$ bestimmt. Die einzelnen Verhältniswerte werden aufsummiert und der Durchschnitt über die Anzahl aller Terme im Edit $|T_e|$ ermittelt. Dieser Durchschnitt bildet den Wert des Features.

$$f_i(e) = \frac{\sum \frac{tf(t,e)}{tf(t,a_{new})}}{c} ar|T_e|,\ t \in T_e \tag{3.14}$$

**Longest-Word.** Dieses Feature greift nochmals, wie schon das Character-Distribution Feature und andere, die Natürlichsprachlichkeit des Edittextes auf. Jedoch geschieht dies auf der Termebene und nicht wie bei den entsprechenden Features zuvor auf der Zeichenebene. Grundlage bildet die Annahme, dass Worte des normalen Sprachgebrauchs eine gewisse Wortlänge nicht überschreiten. Trifft man im Edittext nun auf einen Term, der diesen Grenzwert überschreitet, liegt der Verdacht des Vandalismus nahe. Das Longest-Word Feature nimmt die Länge des längsten Terms des Edittextes als Featurewert an.

$$f_i(e) = \max(|t|),\ t \in T_e \tag{3.15}$$

**Pronoun-Frequency.** Um Vandalismus aufzudecken, der sich durch das Einstreuen von bewertenden Aussagen auszeichnet, wurden die beiden Features Pronoun-Frequency und

Pronoun-Impact entwickelt. Mit der Äußerung von persönlicher Meinung geht die Benutzung von Pronomen der ersten und zweiten Person einher. Die dritte Person ist hingegen die präferierte Form in Lexikonartikeln.
Die Pronomen der ersten und zweiten Person im Edittext sind definiert als eine Teilmenge aller im Edit enthaltenen Terme $P_e \subset T_e$. Der Wert des Pronoun-Frequency Features ist bestimmt durch das Verhältnis von Pronomen $P_e$ der ersten und zweiten Person, zu allen Termen $T_e$ des Edittextes.

$$f_i(e) = \frac{|P_e|}{|T_e|} \tag{3.16}$$

Beide Pronoun Features sind sprachabhängig, da die Pronomen der ersten und zweiten Person der verwendeten Sprache bekannt sein müssen.
Eine Übersicht über die verwendeten Pronomen befindet sich im Anhang A.3.

**Pronoun-Impact.** Dieses Feature tritt dem Pronoun-Frequency Feature gewissermaßen entgegen. Die losgelöste Feststellung der Verwendung von Pronomen der ersten und zweiten Person ist kein ausreichender Indikator für Vandalismus. Es ist notwendig den Kontext des Artikels in die Betrachtung mit einzubeziehen. Im Artikel über das Pronom „You“ zum Beispiel gehört das Pronom selbst, sowie dessen Deklinationen zum Artikelvokabular und ist somit auch in einem Edit zulässig.
Die Verfahrensweise dieses Features gleicht dem Vorgehen des Term-Impact Features, mit dem Unterschied, dass nur die Pronomen der ersten und zweiten Person des Edittextes berücksichtigt werden.
Die Pronomen der ersten und zweiten Person sind eine Teilmenge, $P_e \subset T_e$, aller im Edittext verwendeten Terme. Die Häufigkeitsfunktion $pf(p, e)$ gibt an, wie oft ein Pronom im Edit enthalten ist. Hingegen ermittelt $pf(p, a_{new})$, wie oft ein Pronom in der neuen Revision des Artikels auftritt. Für jedes Pronom $p \in P_e$ wird das Verhältnis $\frac{pf(p,e)}{pf(p,a_{new})}$ bestimmt und die einzelnen Verhältniswerte aufsummiert. Der Durchschnitt über die Anzahl aller Pronomen der ersten und zweiten Person im Edit $|P_e|$ ergibt den Wert des

Features.

$$f_i(e) = \frac{\sum \frac{pf(p,e)}{pf(p,a_{new})}}{|P_e|},\ p \in P_e \tag{3.17}$$

**Vulgarism-Frequency.** Die Beobachtung, dass Vandalismus in Wikipedia oft mit dem Einsatz von anstößigem bzw. obszönen Vokabular verbunden ist, greifen die beiden Features Vulgarism-Frequency und Vulgarism-Impact auf. Die vulgären Terme sind eine Teilmenge aller im Edit enthaltenen Terme $V_e \subset T_e$. Das vorliegende Feature berechnet dabei das Verhältnis von vulgären Termen $V_e$ zu allen Termen $T_e$ des Edittextes.

$$f_i(e) = \frac{|V_e|}{|T_e|} \tag{3.18}$$

Um die anstößigen Terme im Edittext zu finden, wird ein Abgleich mit einer Liste anstößiger englischer Wörter, in verschiedene Schreibvarianten, durchgeführt. Durch den Einsatz dieser Liste sind beide Vulgarism Features sprachabhängig.

**Vulgarism-Impact.** Diese Feature ist eine Art Gegengewicht zum Vulgarism-Frequency. Ein Term, der bei der alleinigen Betrachtung des Edittextes als vulgär eingestuft wurde, muss dies nicht zwangsläufig im Kontext des Artikels sein. So ist z. B. das Wort „Nazi“ im Artikel über Adolf Hitler nicht vulgär, da es zum Artikelvokabular zählt. Im Artikel über Meerschweinchen hingegen ist es als anstößig einzustufen.
Dieses Feature arbeitet analog zum Term-Impact und Pronoun-Impact Feature. Bei der Berechnung werden nur die vulgären Terme des Edittextes berücksichtigt.
Die vulgären Terme sind eine Teilmenge, $V_e \subset T_e$, aller im Edittext verwendeten Terme. Die Häufigkeitsfunktion $vf(v, e)$ gibt an, wie oft ein vulgärer Term im Edit enthalten ist. Hingegen ermittelt $vf(v, a_{new})$, wie oft ein vulgärer Term in der neuen Revision des Artikels auftritt. Für jeden vulgären Term $v \in V_e$ wird das Verhältnis $\frac{vf(v,e)}{vf(v,a_{new})}$ bestimmt und die einzelnen Verhältniswerte aufsummiert. Der Durchschnitt über die Anzahl aller

vulgärer Terme im Edit $|V_e|$ ergibt den Wert des Features.

$$f_i(e) = \frac{\sum \frac{vf(v,e)}{vf(v,a_{new})}}{|V_e|},\ v \in V_e \tag{3.19}$$

Damit ist die Auflistung aller Features der Termebene komplett und es folgen nun die Features, die auf Artikelebene operieren.

**Size-Ratio.** Um Vandalismus, bei dem größere Teile des Artikels oder der komplette Artikel gelöscht werden, zu erfassen, wurde das Size-Ratio Feature eingeführt. Der Featurewert ergibt sich aus dem Verhältnis des Umfangs der neuen $a_{new}$ zur alten $a_{old}$ Artikelrevision, gemessen in Byte.

$$f_i(e) = \frac{|a_{new}|}{|a_{old}|} \tag{3.20}$$

**Context-Relation.** Bei der Berechnung dieses Features kommt das so genannte Vektor-Space-Model zum Einsatz. Es wurde erstmals von Salton et al. in [24] beschrieben und dient im Information Retrieval dazu, die Ähnlichkeit von Dokumenten zu bestimmen.
Im Vector-Space-Model wird ein Dokument $d$ als ein Vektor $\mathbf{d} = (t_1, \ldots, t_n)^T$ der im Dokument enthaltenen Terme repräsentiert. Jeder Term entspricht dabei einer Dimension des Vektors und besitzt einen eindeutigen Index. Der Vektor enthält nicht den Term selbst, sondern im einfachsten Fall eine Eins. Die Terme können zusätzlich gewichtet werden, um die Relevanz des einzelnen Termes $t$ für das Dokument $d$ zu modellieren. Eine einfache Form der Gewichtung ist z. B. die Termfrequenz $tf(t, d)$, die angibt wie oft der Term $t$ im Dokument $d$ enthalten ist.
Zwei Dokumente $d_1$ und $d_2$ können auf diese Weise als Vektoren, $\mathbf{d_1}$ und $\mathbf{d_2}$, im Raum, den die Terme beider Dokumente aufspannen, betrachtet werden. Die Ähnlichkeit der beiden Dokumente lässt sich nun auf eine Betrachtung der beiden Vektorrepräsentationen reduzieren. Es existieren unterschiedliche Ähnlichkeitsmaße um dies zu realisieren. Ein gängiges Maß ist die Kosinusähnlichkeit.
Bei der Kosinusähnlichkeit wird der Kosinus des Winkels $\varphi$, den die beiden Vektorre-

präsentationen $\mathbf{d_1}$ und $\mathbf{d_2}$ der Dokumente miteinander einschließen, ermittelt.

$$sim(d_1, d_2) = \cos(\varphi) = \frac{\mathbf{d_1} \cdot \mathbf{d_2}}{|\mathbf{d_1}|\,|\mathbf{d_2}|} \tag{3.21}$$

Ein Kosinus von Eins impliziert, dass die beiden Dokumentvektoren, $\mathbf{d_1}$ und $\mathbf{d_2}$, aufeinander fallen. Dies entspricht der maximalen Kosinusähnlichkeit und deutet auf eine hohe Ähnlichkeit der Dokumente, $d_1$ und $d_2$, hin. Ein Kosinus von Null hingegen impliziert, dass die beiden Dokumentvektoren senkrecht zueinander stehen. Die beiden Dokumente haben keinen einzigen gemeinsamen Term, was dafür spricht, dass die beiden Dokumente, $d_1$ und $d_2$, unähnlich sind.

Bei einem Edit stellt sich die Frage, ob der eingefügte Inhalt relevant für den Artikel ist, bzw. der gelöschte Inhalt relevant für den Artikel war. Um dieser Frage nachzugehen, verfährt das Context-Relation Feature folgendermaßen: Zunächst werden $n$ Schlüsselwörter aus dem Edittext extrahiert. Dafür werden die $n$ häufigsten Substantive des Edittexts bestimmt. Anschließend wird überprüft, ob für ein Schlüsselwort $key$ ein Artikel $a_{key}$ in Wikipedia existiert. Es wird folgende Annahme getroffen: Besitzt der Edit Relevanz, so sollten die alte Artikelrevision $a_{old}$ und der Artikel des Schlüsselwortes $a_{key}$ eine gewisse Ähnlichkeit aufweisen, da sie in einem weiteren Sinne von verwandten Themen handeln sollten. Diese Ähnlichkeit $sim(a_{old}, a_{key})$ wird mit Hilfe des Vector-Space-Models bestimmt. Das Maximum der ermittelten Ähnlichkeiten von alter Revision des Artikels $a_{old}$ und den gefunden Artikeln $A_{key}$ wird als Featurewert angenommen.

$$f_i(e) = max(sim(a_{old}, a_{key})),\ a_{key} \in A_{key} \tag{3.22}$$

Dieses Feature hängt von Wikipedia ab und ist somit indirekt ebenfalls sprachabhängig.

**Replacement-Similarity.** Die Besonderheit des Replacement-Similarity Feature ist, dass es nur bei der Manipulation durch Ersetzen angewendet werden kann. Dieses Feature soll quantifizieren, wie ähnlich sich der entfernte und der dafür eingefügte Text sind. Ergibt sich eine geringe oder gar keine Ähnlichkeit, wurde relevanter Text aus dem Artikel entfernt und an dessen Stelle Text eingefügt, der von etwas Anderem handelt. Das kann

ein Indiz für Vandalismus sein. Wird hingegen eine hohe Ähnlichkeit ermittelt, wurde der entfernte Text, durch thematisch ähnlichen Text ersetzt, wie es z. B. der Fall ist, wenn ein Absatz umformuliert wird.
Die Ähnlichkeit von gelöschtem (orginal) Text *del* und des dafür eingefügten (ersetzten) Textes *ins* wird mit Hilfe des Vector-Space-Models bestimmt.

$$f_i(e) = sim(ins, del) \tag{3.23}$$

Die letzten drei Features sind auf der Metaebene eingestuft, da sie sich Zusatzinformationen über den Edit zunutze machen.

**Anonymity.** Nach dem Ergebnis von *Study 1* [11], wurden 97 % des gefundenen Vandalismus von unregistrierten Benutzern verübt. Diese Charakteristik bildet die Grundlage für das Anonymity Feature. Es ist binär und nimmt den Wert Eins an, falls der Edit $e$ anonym durchgeführt wurde und Null, falls ein registrierter Benutzer dafür veranwortlich ist.

**Comment-Length.** Die Idee für dieses Feature bildet die Annahme, dass der Vandale seine Tat nicht noch kommentiert, sondern möglichst schnell das Ergebnis seiner Manipulation vor Augen haben möchte. Hingegen ist es innerhalb der Gemeinschaft Wikipedias üblich, seinen Edit zu kommentieren, wodurch sich destruktive gegenüber konstruktiven Edits abgrenzen lassen sollten.
Der Wert des Features ist die Länge des Kommentars in Byte.

**Edits-Per-User.** Grundlage dieses Features bildet die Überlegung, dass konstruktive Beiträge häufig von aktiven Benutzern Wikipedias stammen. Demgegenüber führen Vandale nur wenige Edits durch, da sie im Wiederholungsfall meist blockiert werden und somit keine größere Anzahl von Edits ansammeln können. Diese Vermutung wird durch die Erkenntnisse von Priedhorsky et al. [22] unterstützt.
Der Wert des Features ist die Anzahl aller bisher erfolgten Edits, die dem Benutzer oder der IP zugeordnet werden können.

# 4 Evaluierung

Das vorliegende Kapitel beinhaltet die Auswertung der Ergebnisse für die automatische Erkennung von Vandalismus in Wikipedia durch einen trainierten Lernalgorithmus. Zunächst wird erläutert, welche Experimente durchgeführt und welche Vergleichsmaße herangezogen wurden, um die Klassifikation des Lernalgorithmus zu beurteilen. Danach folgt eine Gegenüberstellung der erreichten Klassifikationsgüte mit den Ergebnissen, die die beiden etablierten Bots, AntiVandaleBot und ClueBot, bei Erkennung von Vandalismus erzielen. Anschließend werden die Features einzeln auf ihre Klassifikationsgüte hin untersucht, um festzustellen, wie gut jedes der entwickelten Features geeignet ist, destruktive Edits von konstruktiven Edits abzugrenzen. Abschließend wird die praktische Einsetzbarkeit der Vandalismuserkennung durch ein Lernverfahren analysiert. Zum einen wird der Durchsatz in Edits pro Sekunde untersucht, zum anderen werden mögliche Einschränkungen aufgrund der Sprachabhängigkeit bestimmter Features diskutiert.

## 4.1 Methodik und Vergleichsmaße

Die Experimente wurden mit Hilfe des WEKA (Waikato Environment for Knowledge Analysis) Frameworks [33] durchgeführt. Es handelt sich dabei um eine, von der Universität Waikato entwickelte, Software, die eine Reihe von Verfahren des maschinellen Lernens implementiert.
Es wurde der „SimpleLogistic“ Klassifizierer für die Experimente benutzt. Das Verfahren zur Klassifikation basiert auf der linear logistischen Regression und ist durch Hall und Frank in [17] und [26] beschrieben.

Die Klassifikation durch den Lernalgorithmus wurde in verschiedenen Punkten, wie z. B. Klassifikationsgüte und Durchsatz, mit den beiden Bots AntiVandalBot [27] (kurz AVB)

und ClueBot [5] verglichen. Um die Vergleichbarkeit der Durchsatzmessungen von Lernverfahren und Bots zu gewährleisten, war es notwendig, eine einheitliche Laufzeitumgebung zu schaffen. Zu diesem Zweck wurde das Regelwerk der beiden Bots in JAVA nachimplementiert.

### 4.1.1 Wahrheitsmatrix und Bewertungsgrößen der Klassifikation

Die Ergebnisse einer Klassifikation lassen sich, bei zwei Klassen, gemäß der vier Kombinationsmöglichkeiten von tatsächlicher und vorhergesagter Klasse in einer 2x2 Matrix anordnen. Für die Klassifikation von Edits, in konstruktive und destruktive Beiträge, ist dies in Tabelle 4.1 dargestellt. Diese Matrix wird als Wahrheits- oder Konfusionsmatrix bezeichnet. Aus der Wahrheitsmatrix lassen sich wichtige Vergleichsmaße wie Precision und Recall [30], und die Klassifikationsgüte ableiten.
Die Zellen der Matrix enthalten die Anzahl der Klassifikationen, die in entsprechender Kombination beim Test des Klassifizierers aufgetreten sind. Es sind englische Bezeichnungen für diese Kombinationen gebräuchlich. Das erste Wort (true, false) drückt aus, ob die Klassifikation korrekt ist und das zweite Wort (positive, negative) gibt an, um welche der beiden Klassen es sich handelt [4]. In der Tabelle 4.1 sind die Vandalismus Edits die positive Klasse und die konstruktiven Edits die negative Klasse, unter Berücksichtigung der Fragestellung: „Ist der Edit Vandalismus?“. Im Sinne eines One-Class-Klassifikationsproblems entspricht die positive Klasse folglich der Zielklasse und die negative Klasse den Außenliegenden [29].

Auf der grau hinterlegten Diagonalen der Wahrheitsmatrix befinden sich die korrekt klassifizierten Edits $n_{v\to v}$ und $n_{k\to k}$, also konstruktive Edits, die als konstruktiv erkannt

| klassifiziert als → | konstruktiv | Vandalismus |
|---|---|---|
| konstruktiv | true-negative $n_{k\to k}$ | false-positive $n_{k\to v}$ |
| Vandalismus | false-negative $n_{v\to k}$ | true-positve $n_{v\to v}$ |

Tabelle 4.1: Die Wahrheitsmatrix spiegelt die möglichen Kombinationen bei der Klassifikation von zwei Klassen wieder. Auf der grau hinterlegten Diagonalen befinden sich die korrekt klassifizierten Edits $n_{v\to v}$ und $n_{k\to k}$. Auf der anderen Diagonalen die missklassifizierten Edits $n_{v\to k}$ und $n_{v\to k}$. Aus der Wahrheitsmatrix lassen sich wichtige Bewertungsgrößen der Klassifikation, wie z. B. Klassifikationsgüte, Precision und Recall, ableiten.

wurden, bzw. Vandalismus, der als solcher erkannt wurde. Daraus errechnet sich der Prozentsatz korrekt klassifizierter Edits: $acc = \frac{n_{v\to v}+n_{k\to k}}{n_{v\to v}+n_{k\to k}+n_{v\to k}+n_{k\to v}} \cdot 100$. Dies ist die so genannte „overall accuracy“, also der Anteil der korrekt klassifizierten Edits an allen getroffenen Vorhersagen. Dieser Wert liefert einen aussagekräftigen Eindruck von der Klassifikationsgüte des Lernalgorithmus. Darüber hinaus ist die Anzahl der false-positive $n_{k\to v}$ Vorhersagen von Interesse. Es wird angestrebt, möglichst wenig konstruktive Edits als Vandalimus einzustufen.
Alternativ zu den absoluten Angaben kann eine prozentuale Betrachtung, bezogen auf tatsächliche Klassenzugehörigkeit erfolgen. Es wird dann von einer Quote oder Rate gesprochen. So ergibt sich z. B. die false-positive Quote zu $q_{k\to v} = \frac{n_{k\to v}}{n_{k\to k}+n_{k\to v}} \cdot 100$.

Zwei im Information Retrieval weit verbreitete Maße, die ebenfalls bei der Bewertung von Klassifikationsergebnissen zum Einsatz kommen, sind Precision und Recall. Sie beziehen sich auf die relevanten Beispiele der Trainingsmenge, was im vorliegenden Fall alle Bespiele der Zielklasse Vandalismus sind. Precision und Recall lassen sich ebenfalls aus den Angaben der Wahrheitsmatrix ableiten.

- *Precision*: Die Precision gibt an, wie viele der als Vandalismus eingestuften Edits auch tatsächlich Vandalismus sind. Aus der Einteilung der Wahrheitsmatrix 4.1 ergibt sich die Precision zu $\frac{n_{v\to v}}{n_{k\to v}+n_{v\to v}}$. Je größer dieser Wert ist, desto besser, da dies weniger false-positive $n_{k\to v}$ Missklassifikationen bedeutet.

- *Recall*: Der Recall hingegen ist das Verhältnis von korrekt klassifizierten Vandalismus-Edits zu allen Vandalismus-Edits innerhalb der Trainingsmenge. Er ergibt sich aus der Wahrheitmatrix 4.1 wie folgt $\frac{n_{v\to v}}{n_{v\to k}+n_{v\to v}}$. Auch hier gilt, je größer der Wert, desto besser, da dies mehr true-positive $n_{v\to v}$ Klassifikationen, also mehr identifizierten Vandalismus bedeutet.

### 4.1.2 Das Verfahren der k-Fold-Cross-Validation

Bei den Experimenten kam das Verfahren der k-Fold-Cross-Validation zum Einsatz. Die Motivation dafür, sowie die Verfahrensweise bei der k-Fold-Cross-Validation, wird nachfoldend erläutert.
Beim überwachten Lernen wird ein Klassifizierer auf einer Menge von Trainingsbeispie-

len $B_{train}$ trainiert (Vgl. Abschnitt 3.1.1). Um anschließend zu evaluieren, wie gut der trainierte Lernalgorithmus klassifiziert, wird eine weitere Menge von Beispielen, die so genannte Testmenge $B_{test}$, benötigt. Dabei muss gewährleistet sein, dass kein Beispiel $b$ der Testmenge gleichzeitig auch in der Trainingsmenge enthalten ist, $b \in B_{test} \wedge b \notin B_{train}$, da dies zur Verfälschung der Ergebnisse führen würde.
Um dieser Anforderung bei nur einem Korpus $S$ gerecht zu werden, wird auf das Verfahren der k-Fold-Cross-Validation zurückgegriffen. Es folgt eine kurze Erläuterung des Verfahrens nach [16]:
Die $n$ Beispiele des Korpus werden in $k$ Teilmengen, $B_1, \ldots, B_k$, zu je $\frac{k}{n}$ Beispielen aufgeteilt. Nun werden $k$ Durchgänge von Training und Test des Klassifizierers absolviert, wobei jeweils immer $k-1$ Teilmengen zusammen die Trainingsmenge, $B_{train} = S \backslash B_k$ bilden und die verbleibende $k$-te Teilmenge als Testmenge $B_{test} = B_k$ dient. Nach den $k$ Durchläufen hat jede der $k$ Teilmengen genau einmal als Testmenge fungiert. Jedes der $n$ Beispiele des Korpus wurde einmal durch den Lernalgorithmus klassifiziert. Die Ergebnisse der $k$ Einzelversuche werden ausgewertet, aufsummiert und über $k$ gemittelt. Die Klassifikationsgüte z. B. ergibt sich zu: $acc_{cv} = \frac{1}{k} \sum_{i=1}^{k} acc_i$.
Üblicherweise wird $k = 10$ gewählt, wie es auch im vorliegenden Fall getan wurde.

## 4.2 Auswertung und Vergleich

Die im vorherigen Abschnitt eingeführten Vergleichsmaße wurde dazu verwendet, die Klassifikation durch das Lernverfahren mit der Erkennung durch die Bots zu vergleichen.

### 4.2.1 Vergleich der Klassifikationsgüte des Lernalgorithmus und der Bots

In der Tabelle 4.2 wird zunächst die Klassifikationsgüte, aufgeschlüsselt nach den Manipulationsformen, gegenübergestellt[1]. Die Unterteilung der Manipulationsformen bedeutet, dass nur die Editierungstellen der jeweiligen Manipulationsform berücksichtigt wurden, um die Featurewerte zu berechnen. Die Werte der Zeile „Zusammen“ wurden er-

[1] Die Ergebnisse der Experimente sind in Form von Wahrheitsmatrizen in Anhang A.5 aufgeführt

| Manipulationsform | Klassifikationsgüte (%) | | | | |
|---|---|---|---|---|---|
| | AVB | ClueBot | Lernalgorithmus | Δ AVB | Δ ClueBot |
| Einfügen | 74,4 | 69,8 | 90,1 | +15,7 | +20,3 |
| Ersetzen | 78,2 | 78,2 | 88,8 | +10,6 | +10,6 |
| Löschen | 82,5 | 82,1 | 92,1 | +9,6 | +10,0 |
| Zusammen | 76,2 | 73,1 | 87,5 | +11,3 | +14,4 |

Tabelle 4.2: Anteil der korrekt klassifizierten Edits für die verschiedenen Manipulationsformen und die zusammengefasste Betrachtung aller Editierungsstellen eines Edits.

mittelt, indem die Inhalte aller im Edit befindlichen Editierungstellen zusammengefasst wurden.

Wie Tabelle 4.2 zu entnehmen ist, erzielt der Lernalgorithmus bei allen Manipulationsformen bessere Klassifikationsgüte als die etablierten Bots.
Eine besonders deutliche Verbesserung, mit einem Plus von durchschnittlichen ca. 18 %, zeigt sich im Fall von eingefügtem Inhalt. Dies könnte darin begründet liegen, dass die Bots weit weniger auf diese Form des Vandalismus eingestellt sind, als dies beim Ersetzen und Löschen der Fall ist. Eine vergleichende Betrachtung von Precision und Recall, wie in Tabelle 4.3 zu sehen, gibt weiteren Aufschluss darüber.

Bei der Gegenüberstellung von Precision und Recall fällt auf, dass ClueBot für alle Manipulationsformen eine Precision von 1,0 aufweist. Dies ist bemerkenswert, bedeutet es doch, dass der Bot nicht einen einzigen konstruktiven Edit fälschlicherweise als Vandalismus eingestuft hat, so dass $n_{k \to v} = 0$. Dieser Wert wird jedoch mit einem deutlich geringeren Recall erkauft. Den Spitzenwert erreicht er beim Löschen, mit 49 % gefundenem Vandalismus, beim Einfügen entdeckt er 3 % des Vandalismus. Ebenso wie ClueBot erzielt AntiVandalBot seine Bestwerte für Precision (0,85) und Recall (0,61) beim Löschen

| Manipulationsform | AVB | | ClueBot | | Lernalgorithmus | |
|---|---|---|---|---|---|---|
| | Precision | Recall | Precision | Recall | Precision | Recall |
| Einfügen | 0,67 | 0,35 | 1,0 | 0,03 | 0,82 | 0,87 |
| Ersetzen | 0,69 | 0,53 | 1,0 | 0,29 | 0,86 | 0,76 |
| Löschen | 0,85 | 0,61 | 1,0 | 0,49 | 0,90 | 0,89 |
| Zusammen | 0,71 | 0,43 | 1,0 | 0,16 | 0,83 | 0,77 |

Tabelle 4.3: Precision und Recall für die Zielklasse destruktiver Edits.

und die schlechtesten beim Einfügen. Dennoch lag der Recall für das Einfügen mit 0,35 deutlich höher als bei ClueBot mit einem Recall von 0,03. Diese Werte bestätigen die Vermutung, dass die Bots hauptsächlich für das Identifizieren von Löschvandalismus konzipiert wurden.

Was die Precision betrifft, kann der Lernalgorithmus nicht mit den Werten von ClueBot konkurrieren, jedoch weist er Werte im Bereich von 0,82 bis 0,9, für die verschiedenen Manipulationsformen auf. Am Beispiel von Löschvandalismus bedeutet das, stuft der Klassifizierer einen Edit als Vandalismus ein, so liegt er in 90 % der Fälle richtig.
Der durchgehend bessere Recall des Klassifizierers zeigt, dass dieser deutlich mehr Vandalimusdelikte identifiziert, als die beiden etablierten Bots.

Die Ergebnisse lassen folgende Schlussfolgerungen zu: Vandalismus in Wikipedia kann mit Hilfe maschinellen Lernens erkannt werden und die dabei erzielte Erkennungsleistung übertrifft die der bisherigen Ansätze deutlich.

### 4.2.2 Klassifikationsgüte der einzelnen Feature

Nachdem die beiden Grundfragestellungen der Arbeit positiv beantwortet werden konnten, werden nun weitere Aspekte der Vandalismuserkennung durch einen Lernalgorithmus diskutiert.
Zunächst wird der Frage nachgegangen, welche Features den größten Beitrag zur Erkennung von Vandalismus leisten. Daran lässt sich überprüfen, inwieweit die Charakteristik von Vandalismus, auf denen ein Feature basiert, erfasst werden konnte.

Um zu ermitteln, wie gut jedes einzelne Feature für die Klassifikation von Edits geeignet ist, wurde wieder für den gesamten Korpus eine 10-Fold Cross Validation durchgeführt. Jedoch wurde dem Lernalgorithmus nur jeweils ein Feature bereitgestellt. Die Ergebnisse dieser Experimente lassen sich in Tabelle 4.4 ablesen. Ein Feature wurde als „unbedeutend“ eingestuft, wenn es einen Recall von Null aufwies und somit nicht in der Lage war, Vandalismus zu erkennen.
Bezogen auf die durchschnittliche Klassifikationsgüte erweisen sich vor allem das Edits-Per-User Feature und das Vulgarism-Impact Feature als besonders geeignet, um destruktive Edits zu beschreiben. Die Tatsache, dass Vandalen entweder nach ein paar Edits

| **Feature** $f$ | **Klassifikationsgüte (%)** | | | |
|---|---|---|---|---|
| | Einfügen | Ersetzen | Löschen | Zusammen |
| **Buchstabenebene** | | | | |
| Char-Distribution | unbedeutend | 68.92 | 78.57 | unbedeutend |
| Char-Sequence | 68.64 | 73.15 | 70.36 | 70.43 |
| Compressibility | unbedeutend | unbedeutend | 81.79 | unbedeutend |
| Upper-Case-Ratio | 72.78 | 75.05 | unbedeutend | 70.32 |
| **Termebene** | | | | |
| Term-Impact | unbedeutend | 75.48 | 62.86 | unbedeutend |
| Longest-Word | unbedeutend | 69.98 | 83.57 | 70.85 |
| Pronoun-Frequency | 70.41 | 71.88 | unbedeutend | 69.15 |
| Pronoun-Impact | unbedeutend | 69.77 | 75.00 | 71.17 |
| Vulgarism-Frequency | 75.74 | 79.92 | unbedeutend | 72.02 |
| Vulgarism-Impact | 74.95 | 81.40 | 81.02 | 78.40 |
| **Artikelebene** | | | | |
| Size-Ratio | 69.63 | 79.28 | 82.86 | 73.19 |
| Replacement-Similarity | – | unbedeutend | – | – |
| Context-Relation | unbedeutend | unbedeutend | 65.71 | 69.26 |
| **Metaebene** | | | | |
| Anonymity | unbedeutend | unbedeutend | unbedeutend | unbedeutend |
| Comment-Length | unbedeutend | unbedeutend | unbedeutend | unbedeutend |
| Edits-Per-User | 83.83 | 78.65 | 83.21 | 80.43 |

Tabelle 4.4: Übersicht über die Klassifikationsgüte der einzelnen Feature. Ein Feature wurde als „unbedeutend“ eingestuft, wenn es nicht in der Lage war, Vandalismus zu erkennen.

die Lust an ihrem Tun verlieren oder blockiert bzw. gesperrt werden, scheint der Grund dafür zu sein. Die guten Leistungen des Vulgarism-Impact Feature waren zu erwarten, wird doch in allen Untersuchungen [11], [31], [22] explizit auf die anstößig, vulgäre Charakteristik von Vandalismus in Wikipedia hingewiesen.
Hingegen sind die Features Comment-Length und Anonymity vollkommen wirkungslos. Die Überlegung die zu den Features geführt haben (siehe Abschnitt 3.2.1), waren also ungeeignet. Ein interessantes Ergebnis für das Anonymity Feature, da nach den Ergebnissen der *Study 1* [7] 97 % des Vandalismus von nicht registrierten Benutzern verübt wurde. Im Gegensatz dazu sprechen Kittur et al. [15] im Fall von anonymen Edits an Artikeln, von überwiegend konstruktiven Beiträgen. Von den im Korpus enthaltenen Vandalismusbeispielen wurden 32,9 % anonym verübt. Demgegenüber steht ein Wert von 25,5 % anonymer Beiträge bei den konstruktiven Edits. Diese Zahlen stützen weder

die Aussage der *Study 1* noch die von Kittur et al.. Aus den Beispielen des Korpus ließ sich keine Gesetzmäßigkeit zwischen der Anonymität und der Klasse eines Edits herleiten. Deshalb hat das Anonymity Feature versagt.

### 4.2.3 Durchsatz der einzelnen Feature

Um eine Einschätzung über die Performance des Lernalgorithmus liefern zu können, wurde der Durchsatz gemessen. Für die Messung der Zeit wurde die reine Berechnungszeit des Featurewertes zugrunde gelegt. Die Aggregation der Daten, wie das Laden von der Festplatte oder die Remote-Kommunikation mit der MediaWiki API [6], wurde nicht berücksichtigt. Diese Vorbereitungszeit ist in der Praxis immer vorhanden und abhängig von der Ausführungsumgebung. Damit diese Abhängigkeit nicht verfälschend wirkt und die Messergebnisse vergleichbar sind, wurde wie beschrieben vorgegangen.

| **Feature** $f$ | **Durchsatz** $d$ **in** $\frac{Edits}{s}$ |
|---|---|
| **Zeichenebene** | |
| Char-Distribution | 1347 |
| Char-Sequence | 43 |
| Compressibility | 618 |
| Upper-Uase-Ratio | 656 |
| **Termebene** | |
| Term-Impact | 4 |
| Longest-Word | 319 |
| Pronoun-Frequency | 351 |
| Pronoun-Impact | 53 |
| Vulgarism-Frequency | 181 |
| Vulgarism-Impact | 33 |
| **Artikelebene** | |
| Size-Ratio | 8198 |
| Replacement-Similarity | 9 |
| Context-Relation | 3 |
| **Metaebene** | |
| Anonymity | 8545 |
| Comment-Length | 14242 |
| Edits-Per-User | 813 |
| **Vergleichswerte** | |
| AntiVandalBot | 2 |
| ClueBot | 3 |

Tabelle 4.5: Durchsatz $d$ der einzelnen Feature $f$ gemessen in Edits pro Sekunde. Vergleichswerte bietet der Durchsatz der beiden Bots. Die Werte wurden auf einem Standard PC mit einer 1,53 GHz CPU und 512 MB RAM ermittelt.

In Tabelle 4.5 ist der Durchsatz $d$, in Edits pro Sekunde, der einzelnen Feature angeben. Kein Feature unterschreitet den Durchsatz von 2 bzw. 3 Edits pro Sekunde, den die beiden Bots aufweisen.
Für die Klassifikation eines Edit müssen alle Feature komplett berechnet worden sein. Erfolgt diese Berechnung sequentiell, liegt der Durchsatz für den gesamten Vorgang bei ca. 1,5 Edits pro Sekunde. Es liegt nahe, die Berechnung der einzelnen Feature in Threads auszulagern und sie parallel auszuführen. Dabei bildet das langsamste Feature, Context-Relation mit nur 3 Edits pro Sekunde, den Flaschenhals und gibt somit die obere Schranke für den Durchsatz vor. Für die parallele Berechnung der Features wurde ein Durchsatz von ca. 2,6 Edits pro Sekunde ermittelt, was dem Durchsatzniveau der Bots entspricht.
Um die Angaben über den Durchsatz einordnen zu können, sei an dieser Stelle nochmals auf die, von Priedhorsky et al. [22] ermittelte, Editrate von ca. 280.000 Edits pro Tag[2] verwiesen. Dies entspricht ca. 3,2 Edits pro Sekunde.
Im Zeitraum vom 5. Dezember bis 11. Dezember 2007 ergaben 10 Stichproben der Recent-Changes des englischen Wikipedia eine durchschnittliche Editrate von ca. 1,8 Edits pro Sekunde[3]. Die Stichproben erfolgten zufällig und ohne Berücksichtigung von Stoßzeiten und Ruhephasen in Wikipedia.

### 4.2.4 Klassifikationsgüte ohne rechenintensive Features

Im Kontext der Durchsatzanalyse stellt sich die Frage, wie gut der Lernalgorithmus noch klassifiziert, wenn man die rechenintensiven Features ausklammert. Um dies zu beantworten, wurden Experimente mit einem Feature Set $\mathcal{F}_{d>10}$ durchgeführt, welches nur Features enthält, die einen Durchsatz $d$ von mehr als 10 Edits pro Sekunde aufweisen. Folglich entfallen im Feature Set $\mathcal{F}_{d>10}$ die Features: Term-Impact (4 Edits/s), Context-Relation (3 Edits/s) und Replacement-Similarity (9 Edits/s). Diese Ergebnisse werden zum Vergleich mit den Ergebnissen, basierend auf dem kompletten Feature Set $\mathcal{F}_{alle}$, in Tabelle 4.6 dargestellt.

[2] Für das englische Wikipedia im Zeitraum vom 12.04 - 11.05.2007.
[3] Berücksichtigt wurden nur Edits von Artikeln. Edits an Benutzer- und Diskussionsseiten wurden ignoriert. Ebenso blieben von Bots verursachte Edits unberücksichtigt.

| **Manipulationsform** | **Klassifikationsgüte (%)** Set $\mathcal{F}_{alle}$ | Set $\mathcal{F}_{d>10}$ |
|---|---|---|
| Einfügen | 90,1 | 90,1 |
| Ersetzen | 88,8 | 87,7 |
| Löschen | 92,1 | 90,7 |
| Zusammen | 87,5 | 88,2 |

Tabelle 4.6: Gegenüberstellung der Klassifikationsgüte der beiden Feature Sets $\mathcal{F}_{alle}$ (alle Features) und $\mathcal{F}_{d>10}$ (Features mit einem Durchsatz von mehr als 10 Edits pro Sekunde).

Wie zu erwarten hat die Verwendung eines abgewandelten Feature Sets eine Veränderung der Klassifikationsgüte zur Folge. Dass dabei drei Features unberücksichtigt bleiben, hatte nicht zwangsläufig negative Auswirkungen auf die Klassifikationsgüte, wie die Ergebnisse der drei Kategorien „Einfügen“, „Löschen“ und „Zusammen“ zeigen. Für das Einfügen war dieses Resultat zu erwarten, sind doch alle drei unberücksichtigten Features als unbedeutend für diese Kategorie eingestuft (Vgl. Tabelle 4.4). Selbst mit Kenntnis der Klassifikationsgüte der einzelnen Feature (Tabelle 4.4) ist es schwierig, wenn nicht gar unmöglich, Vorhersagen über das Zusammenspiel der Features bei der Klassifikation zu treffen. Nicht nur die Beschaffenheit der Features, sondern auch die Auswahl bzw. die Anzahl der Features haben Einfluss auf die Klassifikationsgüte. Es wird weitere Experimente geben müssen, die zu optimierten Feature Sets für die jeweilige Manipulationsform führen werden.

### 4.2.5 Klassifikationsgüte ohne sprachabhängige Features

Um die automatische Erkennung auf ein anderssprachiges Wikipedia-Projekt zu übertragen, ist eine Anpassung der sprachabhängigen Features nötig. Teilweise ist dies mit einigem Aufwand verbunden. Zum Beispiel erfordert die Adaption der Vulgarism Features eine neue sprachspezifische Liste mit anstößigen Ausdrücken. Doch ist dieser Aufwand überhaupt notwendig?

Dieser Frage soll zum Abschluss dieses Kapitels nachgegangen werden, indem die Klassifikationsgüte auf Basis aller Features $\mathcal{F}_{alle}$ mit der Klassifikationsgüte auf Basis der sprachunabhängigen Features $\mathcal{F}_{su}$ gegenüber gestellt wird. Im Feature Set $\mathcal{F}_{su}$ sind fol-

| **Manipulationsform** | **Klassifikationsgüte (%)** | |
|---|---|---|
| | Set $\mathcal{F}_{alle}$ | Set $\mathcal{F}_{su}$ |
| Einfügen | 90,1 | 87,2 |
| Ersetzen | 88,8 | 88,4 |
| Löschen | 92,1 | 91,8 |
| zusammen | 87,5 | 84,4 |

Tabelle 4.7: Gegenüberstellung der Klassifikationsgüte der beiden Feature Sets $\mathcal{F}_{alle}$ (alle Features) und $\mathcal{F}_{su}$ (sprachunabhängige Features).

gende Features enthalten (Vgl. Abschnitt 3.2.1 und Tabelle 3.1): Char-Sequence, Compressibility, Upper-Case-Ration, Term-Impact, Longest-Word, Size-Ratio, Anonymity, Comment-Length, Edits-Per-User.

Es zeigt sich eine Verschlechterung der Klassifikationsgüte von maximal ca. 3 % (Tabelle 4.7) für das Einfügen und die zusammengefasste Betrachtung. Ein bemerkenswertes Ergebnis, verringert sich doch die Anzahl der verwendeten Features um annähernd die Hälfte, von 16 auf 9, wodurch sogar starke Features wie z. B. die Vulgarism und die Pronoun Features unberücksichtigt bleiben. Hier zeigt sich eine weitere Besonderheit des maschinellen Lernens. Nicht nur die Auswahl und Beschaffenheit der Features haben Einfluß auf die Klassifikationsgüte, sondern auch, wie sie vom Lernalgorithmus kombiniert werden. Der auf $\mathcal{F}_{su}$ trainierte Klassifizierer hat den Informationsverlust der durch den Wegfall von 7 Features gegenüber $\mathcal{F}_{alle}$ entstanden ist, nahezu kompensiert, indem er die einzelnen Feature $f \in \mathcal{F}_{su}$, den veränderten Bedingungen entsprechend, anders gewichtet hat.

# 5 Zusammenfassung und Ausblick

Die freie Online-Enzyklopädie Wikipedia baut auf dem Wiki-Prinzip auf. Dadurch ist jeder Besucher Wikipedias in der Lage, nicht nur Artikel zu konsumieren, sondern auch Inhalt und Form des Artikel zu verändern. Dies kann auf unterschiedliche Art geschehen, indem neue Inhalte eingebracht, bestehende Inhalte verändert oder gelöscht werden. Durch diese Form des kollaborativen Schreibens werden die Artikel durch die Gemeinschaft beständig weiterentwickelt und verbessert. Manche Benutzer missbrauchen diese Freiheit auf unsoziale Weise. Sie löschen relevanten Inhalt oder den kompletten Artikel, fügen irrelevanten, falschen oder anstößigen Inhalt ein oder verändern gezielt Fakten. Geschieht dies mit destruktiven Absichten, spricht man von Vandalismus.

Ziel dieser Arbeit war es, Verfahren des maschinellen Lernens für die automatische Erkennung von Vandalismus in Wikipedia einzusetzen und diesen Ansatz zu evaluieren. Die Motivation für diese Form der automatischen Erkennung basiert auf den Erfolgen, die maschinelle Lernverfahren bei der ähnlichen Problematik der Email-SPAM Filterung erzielen konnten.

Zunächst wurde untersucht, in welchen Varianten Vandalismus auftritt, was charakteristisch für solche destruktiven Edits ist und was sie dadurch von konstruktiven Edits unterscheidet.
Zu diesem Zweck wurden manuell 301 Vandalismusfälle analysiert und gekennzeichnet. Dabei fiel auf, dass die existierenden Kategorisierungsvorschläge, die den Vandalismus seinen Charakteristiken entsprechend einteilen, unzureichend sind. Für die Einteilung von Vandalismus in Wikipedia wurde im Zuge dieser Arbeit eine Alternative aufgezeigt. Dabei werden Vandalismus-Edits mittels der beiden eindeutig bestimmbaren Kriterien „Manipulationsform“ (Einfügen, Ersetzen, Löschen) und „veränderter Inhalt“ (Text, Link, Medien, Formatierung) kategorisiert. Häufig auftretenden Kombinationen werden zusätzlich beschreibende Charakteristiken zugeordnet, z. B. kann eingefügter Text

anstößig, Nonsens oder themenfremd sein.
Die 301 annotierten Vandalismusfälle und weitere 639 überprüfte konstruktive Edits wurden zum ersten Wikipedia-Vandalismus-Korpus zusammengefasst. Der Korpus bildete Trainings- und Testmenge für den Lernalgorithmus. Darüber hinaus wurde der Korpus dazu verwendet, den Lernalgorithmus mit anderen Erkennungsansätzen zu vergleichen. Der Korpus bekam den Namen WEBIS-VC07-11 und ist für weitere Forschung auf diesem Gebiet frei zugänglich [32].

Um Vandalismus automatisch erkennen zu können, wurden 16 verschiedene Features entwickelt, die unterschiedliche Charakterisktiken von Vandalismus in Wikipedia quantifizieren. Der Algorithmus erlernt anhand einer Menge von manuell klassifizierten Beispielen, einen Zusammenhang zwischen den berechenbaren Features eines Edits und dessen Klasse. Anschließend ist der Klassifizierer in der Lage, auf Basis der Features, für einen unbekannten Edit, vorherzusagen, ob es sich um einen Vandalismus-Edit handelt oder nicht.

Die Klassifikation durch den Lernalgorithmus wurde mit den Erkennungsleistungen zweier etablierter autonomer Anti-Vandalismus-Bots, AntiVandaleBot und ClueBot, verglichen. Die Bots erkennen Vandalismus durch manuell definierte feste Regeln.
Der Lernalgorithmus erreichte eine Klassifikationsgüte von 87,5 % korrekt klassifizierter Edits und übertraf die Ergebnisse von AntiVandalBot 76,2 % und ClubotBot 73,1 % deutlich. Ein weiterer Beleg der Überlegenheit des Lernalgorithmus ist ein Recall von 0,77 für die Vandalismusklasse. Demgegenüber erreichen die etablierten Bots einen Recall von 0,43 (AntiVandalBot) und 0,16 (ClueBot).
Der Anteil der missklassifizierten konstruktiven Edits, welche zu Unrecht als Vandalismus eingestuft wurden, lag mit 7,7 % in etwa auf dem Niveau von AntiVandalBot von 8,3 %.

Ein entscheidender Faktor für die praktische Einsetzbarkeit ist die Verarbeitungsgeschwindigkeit, da im englischen Wikipedia ca. 3,2 Edits pro Sekunde anfallen. Die beiden Bots bearbeiten durchschnittlich 2 bzw. 3 Edits pro Sekunde. Der Durchsatz der einzelnen Feature reicht von ebenfalls 3 bis zu 14242 Edits pro Sekunde. Für die parallele Berechnung der Features, mit Hilfe von Threads, wurde ein Durchsatz von ca. 2,6 Edits pro Sekunde ermittelt, was dem Durchsatzniveau der Bots entspricht. Experimente bei denen die drei rechenintensivsten Features unberücksichtigt blieben zeigten, dass dies nur einen geringen negativen Einfluss (maximal -3 %) auf die Klassifikationsgüte hat.

Unter dem Aspekt einer Adaption auf anderssprachige Wikipedia-Projekte wurde untersucht, wie sich die Klassifikationsgüte verändert, wenn sprachabhängige Features unberücksichtigt bleiben. Als Konsequenz wurden nur noch 9 von 16 Features berücksichtigt. Es stellte sich heraus, dass sich die Klassifikationsgüte um maximal 3 % verschlechtert.

Die nächsten Schritte auf dem Weg zu einem praktisch einsetzbaren Vandalismusfilter für Wikipedia sind die Optimierung der Klassifikationsgüte und die Steigerung des Durchsatzes.
Um die Klassifikationsgüte weiter zu optimieren muss eine detailliertere Studie von Vandalismus durchgeführt werden. Durch die Studie soll festgestellt werden, ob es noch andere Charakteristiken von Vandalismus gibt, die noch durch kein Feature erfasst werden. Sollten weitere spezifische Eigenschaften von Vandalismus in Wikipedia ausgemacht werden, müssen entsprechende Features das existierende Feature Set komplettieren.
Mit der durchzuführenden Studie geht der Ausbau des Korpus einher, da der Korpus möglichst repräsentativ für Vandalismus in Wikipedia gestaltet sein sollte.
In weiterführenden Experimenten werden spezielle auf die Manipulationsform abgestimmte Feature Sets entstehen, die eine maximale Klassifikationsgüte für die jeweilige Manipulationsform erreichen.
Neben der Optimierung der Klassifikationsgüte ist, mit Blick auf die praktische Einsetzbarkeit, die Verbesserung des Durchsatzes ein Betätigungsfeld für aufbauende Arbeiten.

Eine Untersuchung von Vandalismus in Wikipedia unter sozialwissenschaftlichen Gesichtspunkten ist wünschenswert und leistet einen Beitrag zum Verständnis des Phänomens. Impulse aus der sozialwissenschaftlichen Forschung könnten alternative Einteilungen von Vandalismus hervorbringen.

Die automatische Erkennung von Vandalismus in Wikipedia ist ein weiteres Problem, auf das sich Techniken maschinellen Lernens erfolgreich anwenden lassen. Dies wirft die Frage auf, ob maschinelles Lernen auch gegen ebenso unsoziales Verhalten in ähnlich strukturierten Umgebungen, wie z. B. Vandalismus und Trolling in Online-Foren oder Newsgroups eingesetzt werden könnte. Eine Frage die zukünftige Arbeiten beantworten werden.

Ein Kritikpunkt am vorgestellten Verfahren ist, dass ein Edit isoliert betrachtet wird. Es wird sozusagen stillschweigend vorausgesetzt, dass die alte Revision eines Artikels frei von Vandalismus ist. Eine interessante Fragestellung ist, ob sich eine Art automatisch berechenbares Qualitäts- oder Gütemaß für einen Artikel entwickeln ließe. Jemand, der sich als Laie Informationen zu einem bestimmten Thema beschaffen möchte, könnte dann die Qualität des Artikels abschätzen, ohne selbst Experte für die Artikelthematik sein zu müssen.

# A Appendix

## A.1 Organisation des Korpus

Um den Edit eines Artikels vollständig zu erfassen, genügen die beiden eindeutigen ID's der alten und der neuen Artikelrevision. Sämtliche zusätzlichen Informationen über den Edit, zum Beispiel die Revisionen, Informationen über den Benutzer etc. lassen sich dann mit Hilfe der MediaWiki API [6] beschaffen. Damit der Korpus für ein überwachtes Lernverfahren als Trainingsdatensatz dienen kann, muss zudem die Information über die Klasse des Edits bereitgestellt werden.

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <xsd:element name="edits">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="edit" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="edit">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="oldRevisionID" type="xsd:integer"/>
        <xsd:element name="newRevisionID" type="xsd:integer"/>
        <xsd:element name="vandalism" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="vandalism">
    <xsd:complexType>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

Listing A.1: Das XML-Schema für den Korpus. Um einen Edit abzubilden werden zwangsläufig die alte und die neue Revisions ID benötigt. Sollte es sich bei dem Edit um Vandalismus handeln, wird dem `<edit>` Element das optionale `<vandalism>` Element beigefügt.

Für die strukturierte Speicherung der Information wird das XML-Datenformat eingesetzt. Die Anforderung sowohl die beiden Revisions ID's des Edits, als auch die Kennzeichnung des Vandalismus zu erfassen, spiegeln sich im XML- Schema des Korpus wieder (Abbildung A.1).
Die Abbildung A.2 zeigt eine kleine Beispielinstanz des Schemas A.1.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<edits>
    <edit>
        <newRevisionID>87188208</newRevisionID>
        <oldRevisionID>81704907</oldRevisionID>
    </edit>
    <edit>
        <newRevisionID>136337448</newRevisionID>
        <oldRevisionID>136040167</oldRevisionID>
        <vandalism/>
    </edit>
</edits>
```

Listing A.2: Eine Beispiel XML Instanz des Korpus Schemas A.1 mit nur zwei Edits. Aufgeführt sind ein konstruktiver und ein destruktiver Edit.

Durch die verhältnismäßig einfache Art der Dokumentation von Vandalismus ist es möglich, dass die Benutzergemeinschaft Wikipedias an der Erweiterung des Korpus teilnimmt. Das Wiki-System erlaubt es, durch nur einen Klick, einen Edit rückgängig zumachen. Analog dazu könnte dem Benutzer die Möglichkeit gegeben werden, einen Vandalismus-Edit nicht nur zurückzusetzen, sondern diesen automatisch in den Korpus aufzunehmen.

## A.2 URL's der Vandalismusbeispiele

Tabelle A.1 listet die URLs zu den Vandalismusbeispielen auf, die in Abschnitt 2.3.2 angeführt werden.

| Artikel | URL |
|---|---|
| Banana | http://en.wikipedia.org/w/index.php?diff=137416805&oldid=137238632 |
| African American | http://en.wikipedia.org/w/index.php?diff=129798161&oldid=129673256 |
| Google | http://en.wikipedia.org/w/index.php?diff=132936092&oldid=132934473 |
| Dog | http://en.wikipedia.org/w/index.php?diff=136749558&oldid=136517688 |
| Borat | http://en.wikipedia.org/w/index.php?diff=134880408&oldid=134710659 |
| Klondike (solitär) | http://en.wikipedia.org/w/index.php?diff=28646030&oldid=27154685 |
| Bill Gates | http://en.wikipedia.org/w/index.php?diff=133386656&oldid=133044029 |

Tabelle A.1: URL's der in Abschnitt 2.3.2 aufgeführten Vandalismusbeispiele. Letzter Zugriff: 02.10.2007

## A.3 Pronomen

Liste der englische Pronomen (ersten und zweiten Person) und verschiedene Schreibvariationen[1], die für die Berechnung der Pronoun Feature eingesetzt werden (Abschnitt 3.2.1).

- I, me, myself, mine, my, we, us, ourselves, ourself, ours, our, you, yourself, yours, your, thou, thee, thyself, thine, thy, yourselves, you guys, you all, y'all, youse, youse guys, you-uns, you guys, you all, y'all, youse, youse guys, you-uns, yous, yis, yourselves, y'all's, selves, y'all's, yous's, y'all's

[1] http://en.wikipedia.org/wiki/English_personal_pronouns#Full_list_of_pronouns, Letzter Zugriff: 27.10.2007

## A.4 Relative Buchstabenhäufigkeiten

Tabelle A.2 enthält die Erwartungswerte Buchstabenhäufigkeiten für die englische Sprache, die für die Berechnung des Char-Distribution Feature eingesetzt werden (Abschnitt 3.2.1). Die Werte wurden gemittelt aus den absoluten Angaben, welche die Quelle[2] zur Verfügung stellt.

| | $E(lf(e))$ | | $E(lf(e))$ |
|---|---|---|---|
| a | 0.0793 | n | 0.0718 |
| b | 0.0142 | o | 0.0783 |
| c | 0.0305 | p | 0.0225 |
| d | 0.0407 | q | 0.0033 |
| e | 0.1275 | r | 0.065 |
| f | 0.0227 | s | 0.0638 |
| g | 0.0182 | t | 0.0868 |
| h | 0.05 | u | 0.0292 |
| i | 0.0712 | v | 0.0115 |
| j | 0.0027 | w | 0.0188 |
| k | 0.0065 | x | 0.003 |
| l | 0.0388 | y | 0.0193 |
| m | 0.0253 | z | 0.0012 |
| Space | 0.18482 | | |

Tabelle A.2: Erwartungswerte $E(lf(e))$ der Buchstabenhäufigkeiten in der englischen Sprache.

[2] http://www.staff.uni-mainz.de/pommeren/Kryptologie/Klassisch/1_Monoalph/englisch.html, Letzter Zugriff: 27.10.2007

## A.5 Wahrheitsmatrizen des Lernalgorithmus und der Bots

Die nachfolgenden drei Tabellen A.3, A.4 und A.5, zeigen die Klassifikationsergebnisse der beiden Bots, ClueBot [5] und AntiVandaleBot [27], und des Lernalgorithmus dargestellt als Wahrheitsmatrizen (Abschnitt 4.1.1). Auf diesen Ergebnissen beruhen die in Abschnitt 4.2 diskutierten Werte.

(a) Einfügen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 349 (100 %) | 0 (0 %) |
| Vandalismus | 153 (96,8 %) | 5 (3,2 %) |

(b) Löschen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 182 (100 %) | 0 (0 %) |
| Vandalismus | 50 (51 %) | 48 (49 %) |

(c) Ersetzen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 327 (100 %) | 0 (0 %) |
| Vandalismus | 103 (70,6 %) | 43 (29,4 %) |

(d) Zusammen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 439 (100 %) | 0 (0 %) |
| Vandalismus | 253 (84 %) | 48 (16 %) |

Tabelle A.3: Wahrheitsmatrizen der ClueBot Portierung. Absolute Angaben der Fälle für die vier verschiedenen Kombinationsmöglichkeiten. Die prozentuallen Angaben beziehen sich auf eine Zeile, also auf eine tatsächliche Klasse.

(a) Einfügen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 321 (92 %) | 28 (8 %) |
| Vandalismus | 102 (64,6 %) | 56 (35,4 %) |

(b) Löschen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 171 (94 %) | 11 (6 %) |
| Vandalismus | 38 (38,8 %) | 60 (61,2 %) |

(c) Ersetzen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 292 (89,3 %) | 35 (10,7 %) |
| Vandalismus | 68 (46,6 %) | 78 (53,4 %) |

(d) Zusammen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 586 (91,7 %) | 53(8,3 %) |
| Vandalismus | 171 (56,8 %) | 130 (43 %) |

Tabelle A.4: Wahrheitsmatrizen der AntiVandalBot Portierung. Absolute Angaben der Fälle für die vier verschiedenen Kombinationsmöglichkeiten. Die prozentuallen Angaben beziehen sich auf eine Zeile, also auf eine tatsächliche Klasse.

Auffällig ist, dass ClueBot keine false-positive $n_{k \to v}$ Missklassifikationen produziert (Tabelle A.3). In Zusammenhang mit den niedrigen true-positve $q_{v \to v}$ Quoten, lässt sich vermuten, dass den Regeln des Bots eine eher konservative Strategie bei der Vanda-

(a) Einfügen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 319 (91,4 %) | 30 (8,6 %) |
| Vandalismus | 20 (12,7 %) | 138 (87,3 %) |

(b) Löschen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 172 (94,5 %) | 10 (5,5 %) |
| Vandalismus | 12 (12,2 %) | 86 (87,8 %) |

(c) Ersetzen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 309 (94,5 %) | 18 (5,5 %) |
| Vandalismus | 35 (24 %) | 111 (76 %) |

(d) Zusammen

| klassifiziert als → | konstruktive | Vandalismus |
|---|---|---|
| konstruktiv | 590 (92,3 %) | 49 (7,7 %) |
| Vandalismus | 68 (22,6 %) | 233 (77,4 %) |

Tabelle A.5: Wahrheitsmatrizen des Lernalgorithmus. Absolute Angaben der Fälle für die vier verschiedenen Kombinationsmöglichkeiten. Die prozentuallen Angaben beziehen sich auf eine Zeile, also auf eine tatsächliche Klasse.

lismuserkennung zugrunde liegt. Das Gefälle der true-positve $q_{v \to v}$ Quoten (49 % > 29,4 % > 3,2 %) über die Manipulationsformen hinweg zeigt, dass der Bot am besten Löschvandalimsus identifiziert. Hingegen werden nur 3,2 % des Einfügevandalismus von ClueBot erkannt.

Im Gegensatz zu ClueBot weist AntiVandalBot false-positive $q_{k \to v}$ Quoten bis zu 10,7 % auf (Tabelle A.4). Allerdings stiegen ebenso die true-positive $q_{v \to v}$ Quoten für alle Manipulationsformen. Auch bei AntiVandaleBot fällt auf, dass Einfügevandalimus deutlich schlechter identifiziert wird als Vandalismus der anderen beiden Manipulationsformen. So erkennt AntiVandalBot 61,2 % des Löschvandalismus, aber nur 35,4 % des Einfügevandalismus.

Für die zusammengefasste Betrachtung eines Edits liegt die false-positive $q_{k \to v}$ Quote des Lernalgorithmus mit 7,7 % (Tabelle A.5) in etwa auf dem Niveau von AntiVandalBot von 8,3 %. Die true-positive $q_{v \to v}$ Quoten des Lernalgorithmus liegen für alle Manipulationsformen deutlich über dennen der beiden etablierten Bots. Eine besonders deutliche Verbesserung ist bei der Identifikation von Einfügevandalismus zu verzeichnen. Der Lerngalgorithmus erkennt 87,3 %, hingegen identifizieren die Bots nur 3,2 % bzw. 35,4 % des Einfügevandalismus.

# Literaturverzeichnis


[1] Alexa, The Web Information Company: *Traffic Ranking for en.wikipedia.org*, October 2007. http://www.alexa.com/data/details/traffic_details?url=en.wikipedia.org/wiki/Main_Page, Letzter Zugriff: 25.10.2007.

[2] AmiDaniel, Wikipedia User: *WhodunitQuery Tool*. http://en.wikipedia.org/wiki/User:AmiDaniel/WhodunitQuery, Letzter Zugriff: 04.10.2007.

[3] Androutsopoulos, I., J. Koutsias, K.V. Chandrinos, G. Paliouras and C.D. Spyropoulos: *An Evaluation of Naive Bayesian Anti-Spam Filtering*. Proceedings of the workshop on Machine Learning in the New Information Age, G. Potamias, V. Moustakis and M. van Someren (eds.), 11th European Conference on Machine Learning, Barcelona, Spain, 9–17, 2000.

[4] Baeza-Yates, R., B. Ribeiro-Neto .: *Modern information retrieval*. Addison-Wesley Harlow, England, 1999.

[5] Cobi, Wikipedia Benutzer: *Anit-Vandalismus-Bot: ClueBot*, 2007. http://en.wikipedia.org/wiki/User:ClueBot, Letzter Zugriff: 04.10.2007.

[6] Community, MediaWiki: *MediaWiki API*. http://www.mediawiki.org/wiki/API, Letzter Zugriff: 04.10.2007.

[7] Community, Wikipedia: *First Study on Wikipedia Vandalism (Study1)*, 2006. http://en.wikipedia.org/wiki/Wikipedia:WikiProject_Vandalism_studies/Study1, Letzter Zugriff: 04.05.2007.

[8] Community, Wikipedia: *Cleaning up Vandalism*, 2007. http://en.wikipedia.org/wiki/Wikipedia:Cleaning_up_vandalism, Letzter Zugriff: 25.10.2007.

[9] Community, Wikipedia: *List of Wikipedias*, 2007. http://meta.wikimedia.org/wiki/List_of_Wikipedias, Letzter Zugriff: 24.10.2007.

[10] Community, Wikipedia: *Official policy of the English Wikipedia on vandalism*, 2007. http://en.wikipedia.org/wiki/Wikipedia:Vandalism, Letzter Zugriff: 10.09.2007.

[11] Community, Wikipedia: *Wikipedia Special Statistics*, 2007. http://en.wikipedia.org/wiki/Special:Statistics, Letzter Zugriff: 28.08.2007.

[12] Drucker, H., D. Wu and VN Vapnik: *Support vector machines for spam categorization.* IEEE Transactions on Neural Networks, 10(5):1048–1054, 1999.

[13] Griffith, Virgil: *Wikiscanner.* http://wikiscanner.virgil.gr/, Letzter Zugriff: 25.10.2007.

[14] Hastie, T., R. Tibshirani and J. Friedman: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, 2001.

[15] Kittur, A., B. Suh, B.A. Pendleton and E.H. Chi: *He says, she says: conflict and coordination in Wikipedia.* Proceedings of the SIGCHI conference on Human factors in computing systems, 453–462, 2007.

[16] Kohavi, R.: *A study of cross-validation and bootstrap for accuracy estimation and model selection.* Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, 2:1137–1145, 1995.

[17] Landwehr, Niels, Mark Hall and Eibe Frank: *Logistic Model Trees.* Machine Learning, 59(1-2):161–205, 2005.

[18] Leuf, B. and W. Cunningham: *The Wiki way: quick collaboration on the Web.* Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 2001.

[19] Lloydpick, Wikipedia Benutzer: *Anit-Vandalismus-Bot: CounterVandalism-Bot*, 2007. http://en.wikipedia.org/wiki/User:CounterVandalismBot, Last Access: 04.10.2007.

[20] Mitchell, T.M.: *Machine Learning. WCB*. McGraw-Hill, New York, 1997.

[21] Pgk, Wikipedia Benutzer: *Anti-Vandalismus-Bot: pgkbot*, 2005. http://meta.wikimedia.org/wiki/CVN/Bots, Letzter Zugriff: 04.10.2007.

[22] Priedhorsky, Reid, Jilin Chen, Shyong, Katherine Panciera, Loren Terveen and John Riedl: *Creating, destroying, and restoring value in wikipedia.* GROUP '07: Proceedings of the 2007 international ACM conference on Conference on supporting group work, 259–268, 2007.

[23] Sahami, M., S. Dumais, D. Heckerman and E. Horvitz: *A Bayesian approach to filtering junk e-mail.* Learning for Text Categorization: Papers from the 1998 Workshop, 62, 1998.

[24] Salton, G., A. Wong and C.S. Yang: *A vector space model for information retrieval.* Communications of the ACM, 18(11):613–620, 1975.

[25] Stein, Prof. Dr. Benno: *Vorlesungsskript Maschinelles Lernen und Data Mining.* http://www.uni-weimar.de/cms/medien/webis/teaching/lecture-notes.html#machine-learning, Letzter Zugriff: 01.01.2008.

[26] Sumner, Marc, Eibe Frank and Mark Hall: *Speeding up Logistic Model Tree Induction. 9th European Conference on Principles and Practice of Knowledge Discovery in Databases*, 675–683. Springer, 2005.

[27] Tawker, Joshbuddy, und Cyde Weys (Wikipedia Benutzer): *Anti-Vandalismus-Bot: Anti-Vandal-Bot.* http://en.wikipedia.org/wiki/WP:AVB, Letzter Zugriff: 04.10.2007.

[28] Tax, D.M.J.: *One-class classification.* , Delft University of Technology, June 2001.

[29] Tax, DMJ: *Ddtools, the data description toolbox for matlab, version 1.5. 4*, 2006. http://www-ict.ewi.tudelft.nl/~davidt/dd_manual.pdf, Letzter Zugriff: 16.11.2007.

[30] Van Rijsbergen, C. J.: *Information Retrieval, 2nd edition*. Dept. of Computer Science, University of Glasgow, 1979.

[31] Viégas, F.B., M. Wattenberg and K. Dave: *Studying cooperation and conflict between authors with history flow visualizations*. Proceedings of the SIGCHI conference on Human factors in computing systems, Vienna, Austria, 575–582, 2004.

[32] Web Technology & Information Systems Group, Bauhaus University Weimar: *Wikipedia Vandalism Corpus WEBIS-VC07-11*. http://www.uni-weimar.de/medien/webis/research/misuse, 2007. M. Potthast and R. Gerling (editors).

[33] Witten, I.H. and E. Frank: *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2005.