Bauhaus-Universität Weimar
Fakultät Medien
Institut für Web Technology & Information Systems

Abschlussarbeit
zur Erlangung des akademischen Grades
BACHELOR OF SCIENCE

AUTOMATISCHE ERKENNUNG VON BEARBEITUNGSKONFLIKTEN IN WIKIPEDIA

von

DENNIS HOPPE
Rudolf-Breitscheid Straße 21
99423 Weimar

Betreuer
Prof. Dr. Benno Stein
Dipl.-Inf. Martin Potthast

Februar 2008

Erklärung

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen worden ist. Alle Ausführungen, die wörtlich oder sinngemäß übernommen worden sind, sind als solche gekennzeichnet.

Weimar, den 1. Februar 2008 Dennis Hoppe

Kurzfassung

Gegenwärtig schreiben bei der freien Online-Enzyklopädie Wikipedia über 300.000 registrierte Benutzer mit. Besucher eines Artikels sind in der Lage, direkt ohne vorherige Authentifizierung Inhalte zu bearbeiten. Vor allem bei populären Artikeln, die von vielen Benutzern bearbeitet und betrachtet werden, kommt es immer wieder zu Meinungsverschiedenheiten zwischen einzelnen Benutzern. Sie streiten sich dabei überwiegend um den Inhalt des Artikels. Wikipedia bietet Benutzern Diskussionsseiten zu jedem Artikel an, auf denen Benutzer ihre Beweggründe hinsichtlich der von ihnen vorgenommen Änderungen am Artikel vortragen sollen. Diese werden allerdings dazu selten verwendet, denn die Mechanismen, die es Benutzern einerseits erlauben, mit Leichtigkeit Artikel zu verändern, führen andererseits dazu, dass sie bei Meinungsverschiedenheiten auf zeitraubende Diskussionen verzichten. Stattdessen nehmen sie die Änderungen im Artikel einfach zurück. Ein andauerndes Wechselspiel zwischen Wiederherstellen und Zurücknehmen der Inhalte mündet in einem sogenannten Bearbeitungskonflikt. Diese stören andere Autoren und verhindern die Weiterentwicklung des Artikels.

Die englische Wikipedia umfasst derzeit mehr als zwei Millionen Artikel. Minütlich kommen bis zu einhundert neue Änderungen hinzu, von denen im Durchschnitt jede 30. Bearbeitung Teil eines Bearbeitungskonflikts ist. Die Wikipedia-Gemeinschaft ist daher bestrebt, zwischen Benutzern, die an Bearbeitungskonflikten beteiligt sind, zu vermitteln. Gegenwärtig sind Bearbeitungskonflikte jedoch ausschließlich manuell durch Beobachter erkennbar, so dass bei über zwei Millionen Artikeln und mehreren tausend Änderungen täglich viele Meinungsverschiedenheiten unentdeckt bleiben.

In der vorliegenden Arbeit wird erstmalig ein Verfahren zur automatischen Erkennung von Bearbeitungskonflikte namens *Edward* vorgestellt. Auf diese Weise werden Meinungsverschiedenheiten bereits im Entstehen erkennbar und sind zu schlichten. Eine aufwendige Beobachtung von Artikeln durch unparteiische Dritte ist verzichtbar.

Zur Evaluierung des Verfahrens wurde ein Referenzkorpus per Hand angelegt, in dem 51 Bearbeitungskonflikte dokumentiert sind. Von diesen werden 49 Bearbeitungskonflikte erkannt. Beteiligte Versionen eines Artikels an Bearbeitungskonflikten werden mit einer Precision von 0,95 und einem Recall von 0,90 automatisch erfasst.

Inhaltsverzeichnis

1 Einleitung

Das World Wide Web hat die Art und Weise, wie Menschen zusammenarbeiten, revolutioniert. Eine in diesem Zusammenhang bedeutende Web-Technologie ist das Wiki. Es handelt sich dabei um eine Server-Software für Webseiten, die es Benutzern erlaubt, Web-Inhalte aktiv mitzugestalten. Wikis setzen dabei weder Kenntnisse von Technologien noch von Dokumentsprachen wie HTML voraus. Sie ermöglichen das Publizieren von Inhalten auf Basis einer einfach zu erlernenden und im Umfang stark begrenzten sogenannten Wiki-Syntax.

In diesem Zusammenhang ist die freie Online-Enzyklopädie Wikipedia zu nennen. Sie ist momentan mit weit über fünf Millionen Artikeln und 300.000 registrierten Benutzern das größte Wiki. Wikipedia setzt in gleicher Weise wie andere Wikis keine vorherige Authentifizierung von Benutzern voraus. Jeder Benutzer kann direkt Inhalte bearbeiten.

Die geringe Hemmschwelle beim Einfügen neuer Inhalte als auch die Anonymität der Benutzer stellen Wikipedia tagtäglich vor große Herausforderungen. Die genannten Faktoren sind einerseits für die Popularität von Wikis und im Speziellen von Wikipedia verantwortlich, andererseits sind sie die Ursache für zuvor nicht beobachtbare Probleme: Vandalismus an Artikeln und Konflikte unter Benutzern.

Vandalismus bezeichnet in Wikipedia gezielt von einzelnen Benutzern durchgeführte destruktive Änderungen an Artikeln. Sie verfolgen das Ziel, die Qualität von Artikeln durch unsinnige oder falsche Informationen zu mindern. Auch die systematische Zerstörung von Artikeln durch Löschen dieser oder einzelner Abschnitte ist kein Einzelfall. Da die Autorengemeinschaft in Wikipedia sich selbst kontrolliert, liegt es in ihrer Hand, entsprechende Gegenmaßnahmen zu unternehmen. Freiwillige Benutzer und Administratoren sind in der Konsequenz ausschließlich damit beschäftigt, zerstörerische Bearbeitungen manuell zu entdecken und die betroffenen Artikel zu reparieren. Für weitere Beiträge, die der Weiterentwicklung von Artikeln dienen, bleibt keine Zeit.

Weiterhin stellen Konflikte unter Benutzern ein Problem dar. In einer sozialen, virtuellen Gemeinschaft, wie auch Wikipedia sie darstellt, sind Spannungen und Meinungsverschiedenheiten nicht zu verhindern. Diese werden jedoch selten durch den Austausch von Argumenten ausgetragen. Begünstigt durch den marginalen Aufwand, Bearbeitun-

gen erneut hinzuzufügen, fügen Benutzer ihre inhaltlichen Änderungen oftmals direkt wieder ein, falls diese von anderen Benutzern entfernt werden. Ein ständiges Hin und Her kann entstehen. Ein derartiger Interessenkonflikt, bei dem zwei oder mehr Benutzer ihre Bearbeitungen am Artikel ständig gegenseitig zurücknehmen, wird Bearbeitungskonflikt genannt. Es ist ein spezielles Problem von Wikis und wurde zum ersten Mal bei Wikipedia beobachtet.

Ein Bearbeitungskonflikt in Wikipedia behindert nicht nur andere Benutzer an der Weiterentwicklung, sondern mindert ebenso wie Vandalismus die Qualität der betroffenen Artikel. Es wurden schon Meinungsverschiedenheiten beobachtet, die sich über Tage, Wochen oder sogar Monate hingezogen haben. Bestimmte Kontroversen leben über Jahre hinweg immer wieder auf. Wikipedia selbst, die virtuelle Gemeinschaft, ist nur in der Lage, Konflikte zu erkennen, wenn sie offen sichtbar werden. Diese werden jedoch oftmals nicht ausdiskutiert, sondern stillschweigend im Artikel ausgetragen.

Ähnlich wie bei Vandalismus sind freiwillige Helfer notwendig, die stetig Bearbeitungen in Wikipedia beobachten und gegebenenfalls einschreiten. Ihre Aufgaben beschränken sich dabei auf das Reparieren von Artikeln und die Schlichtung von Benutzerkonflikten.

Zusammenfassend mindern beide Herausforderungen – Vandalismus und Bearbeitungskonflikte – die Qualität von Artikeln und stören andere Benutzer bei der Weiterentwicklung dieser. Bei über zwei Millionen Artikeln in der englischen Wikipedia und bis zu einhundert neuen Änderungen, die minütlich hinzugefügt werden, sind diese Aufgaben manuell nicht mehr zu leisten. Gegenwärtige, effektive Maßnahmen zur automatischen Konflikterkennung in Wikipedia sind derzeit nicht bekannt. In der Folge ist es wichtig, diese Arbeit durch automatische Werkzeuge zu unterstützen. Hierzu wird in dieser Ausarbeitung erstmalig ein Verfahren vorgestellt, welches Bearbeitungskonflikte automatisch offenlegt. Eine frühzeitige Erkennung wird möglich, so dass unparteiische Dritte gezielt zwischen den beteiligten Konfliktparteien vermitteln können.

Nach einer kurzen Vorstellung von Wikipedia und einen Einblick in die Funktionsweise dieser, werden im zweiten Kapitel Ursachen von Bearbeitungskonflikten diskutiert. Es wird herausgestellt, warum insbesondere Wikis solche Konflikte begünstigen. In drittem Kapitel werden strukturelle Eigenschaften von Bearbeitungskonflikten definiert. Dabei erschließt sich sowohl eine globale als auch eine lokale Betrachtungsweise dieser. Anschließend wird der Algorithmus *Edward* im vierten Kapitel vorgestellt, der eine automatische Erkennung von Konflikten ermöglicht. Zur Evaluierung des Verfahrens im vierten Kapitel war eine manuelle Erstellung eines Referenzkorpus notwendig, da bislang keine maschinenlesbare Referenzkollektion, in der Bearbeitungskonflikte dokumentiert

sind, existiert. Abschließend werden die Ergebnisse der Evaluierung diskutiert und ein Ausblick über weitere Herausforderungen gegeben. Im Anhang sind weitere Beispiele für Bearbeitungskonflikte aufgeführt. Weiterhin wird die Neutralität und Glaubwürdigkeit von Wikipedia-Artikeln diskutiert sowie ein XML-Schema präsentiert, anhand dessen Bearbeitungskonflikte XML-basiert dokumentierbar sind.

1.1 Entstehung von Wikipedia

Im März 2000 wurde das Vorläuferprojekt von Wikipedia, *Nupedia*, durch Jimmy D. Wales und Lawrence M. Sanger mit dem Ziel gegründet, die weltweit größte und freie Enzyklopädie im Internet aufzubauen. Bis September 2003 – der Aufgabe von Nupedia – wurden insgesamt 27 *fertige* Artikel und etwas mehr als 60 unvollendete Artikel veröffentlicht (Nupedia, 2003).

Nupedia scheiterte aufgrund des sehr restriktiven, nicht öffentlichen redaktionellen Prozesses. Nur registrierte Autoren durften unter der Aufsicht von Experten Artikel verfassen. Die Richtlinien von Nupedia verlangten zusätzlich von einem Experten die Promotion auf seinem Fachgebiet. Der redaktionelle Teil wurde ebenfalls durch ein striktes Regelwerk vorgegeben, so dass insgesamt sieben zeitraubende Stadien bis zur endgültigen Veröffentlichung eines neuen Artikels durchlaufen werden mussten (Nupedia, 2001).

Innerhalb von zwei Jahren konnte darum keine große Nutzergemeinde aktiviert werden. Aufgrund der langsamen Entwicklung von Nupedia forcierte Sanger als Konsequenz im Januar 2001 das Schwesterprojekt *Wikipedia*. Mit dem gegenteiligen Prinzip – jeder Benutzer darf frei an Artikeln arbeiten – sollte der langwierige Zulassungsprozess von Artikeln umgangen und die Entstehung neuer Artikel beschleunigt werden. Anders als Nupedia entwickelte sich Wikipedia in derselben Zeitspanne rasant. Im September 2006 existierten Wikipedias in 259 Sprachen mit insgesamt mehr als 5,3 Millionen Artikeln. Alleine die vier größten Sprachen – Englisch, Deutsch, Französisch und Polnisch – stellten zusammen mehr als 2,5 Millionen Artikel (Wikipedia, 2007l). Gegenwärtig sind keine offiziellen Zahlen vorhanden. Die Anzahl der englischen Artikel stieg jedoch von 1,5 Millionen Ende 2006 auf beinahe 2,2 Millionen zu Beginn des Jahres 2008.

Sowohl Nupedia als auch Wikipedia sind unter der *GNU Free Documentation Licence* lizensiert. Jeder ist berechtigt Veröffentlichungen in Wikipedia auf kommerzielle oder nicht kommerzielle Weise frei zu kopieren, zu verbreiten und Inhalte abzuändern (Free Software Foundation, 2002).

1.2 Funktionsweise eines Wiki

Artikel in Wikipedia werden durch eine Menge von Versionen repräsentiert. Speichert ein Benutzer Bearbeitungen für einen Artikel, so wird eine neue Version erzeugt und diese dem Artikel hinzugefügt. Jede Version stellt dabei eine eigenständige, vollständige Momentaufnahme des Artikels dar. Für jeden Artikel existiert schließlich eine sogenannte Versionsgeschichte, in der alle Versionen einsehbar sind. Abbildung 1.1 zeigt sie für den Artikel *Bauhaus-Universität Weimar*. Benutzer können auf diese Weise auf einen Blick erfassen, welcher Autor zu einem bestimmten Zeitpunkt Änderungen an einem Artikel vorgenommen hat. Eine Differenzseite hebt desweiteren inhaltliche Änderungen zwischen jeweils zwei Versionen farblich hervor.

Benutzer sind imstande, Bearbeitungen an Artikeln ohne vorherige Anmeldung durchzuführen. Diese sind ohne Prüfung Dritter sofort für Besucher des Artikels sichtbar. Dies ist einerseits ein Grund für die Popularität von Wikis, andererseits versetzt es Teilnehmer in die Lage, mühelos Artikeln Nonsens hinzuzufügen. Es existiert in Wikipedia keine übergeordnete Instanz, die das Verhalten der Benutzer koordiniert. Vielmehr kontrolliert und organisiert sich die Autorengemeinschaft von Wikis und im Speziellen von Wikipedia selbst.

Bei der Vielzahl an – weltweit über 300.000 registrierten – Benutzern stellt sich deshalb zwangsläufig die Frage, warum Wikipedia nicht im Chaos versinkt. Es existieren hierfür Mechanismen, die das Gelingen von Wikis gewährleisten.

Zum einen unterstützt die Versionsgeschichte Benutzer dabei, effizient unerwünschte Änderungen an Artikeln wieder rückgängig zu machen. Versucht ein Teilnehmer gezielt durch destruktives Verhalten Artikel zu *zerstören*, so müssen andere Benutzer die verworfenen Inhalte nicht neu verfassen, sondern wählen gezielt in der Versionsgeschichte eine frühere Version des Artikels aus stellen diese wieder her. Ein Artikel wird auf diese Weise *zurückgenommen*.

Möller (2006) führt weitere Mechanismen auf, die vor allem bei Wikipedia für das Gelingen verantwortlich sind. Sie erlauben genauso wie die Versionsgeschichte eine systematische Beobachtung von Artikeln. Tabelle 1.1 stellt diese vor. Bei unerwünschter Weiterentwicklung von Artikeln wie dem Löschen ganzer Abschnitte sind Benutzer in der Lage frühzeitig einzuschreiten, um Bearbeitungen wieder rückgängig zu machen. Hervorzuheben sind die Beobachtungsliste sowie die dynamische Spezialseite, die die letzten Artikelbearbeitungen anzeigt.

Abbildung 1.1 *: Auszug aus der Versionsgeschichte des Artikels* Bauhaus-Universität Weimar. *Am Ende jeder Zeile befindet sich ein Verweis („rückgängig“), der die Artikelrevision direkt zurückstellen lässt.*

Mechanismus	Erklärung
Beobachtungsliste	Liste der zuletzt vorgenommenen Änderungen an Artikeln.
Diskussionsseiten	Benutzer können hier über Artikelinhalte diskutieren.
Kategorien-System	Die Einteilung von Artikeln in bestimmte Kategorien ermöglicht die einfachere Erfassung von problematischen Seiten. Beispielsweise werden Artikel mit einer subjektiven, einseitigen Sichtweise in eine entsprechende Kategorie eingeordnet.
Administratoren	Ein Administrator besitzt mehr Rechte als ein normaler Benutzer und soll für die Einhaltung der Richtlinien von Wikipedia sorgen. Dazu stehen ihm Mittel wie das Sperren von Benutzern oder Löschen von Seiten zur Verfügung.
Spezialseiten	Dynamische Zusammenstellung von Artikeln nach bestimmten Kriterien wie *neue*, *verbesserungswürdig* oder *potenziell vandaliert*.

Tabelle 1.1 *: Mechanismen, die nach Möller (2006) eine systematische Artikelprüfung ermöglichen, um bei unerwünschter Weiterentwicklung von Artikeln frühzeitig Inhalte wiederherzustellen.*

2 Bearbeitungskonflikte in Wikipedia

In Wikipedia wird ein Bearbeitungskonflikt zwischen Benutzern wie folgt beschrieben (Wikimedia, 2003):

> „[...] edit wars on Wikipedia are reversion wars involving two Wikipedians, or sometimes two factions of Wikipedians. To wit, one Wikipedian edits an article, another Wikipedian reverts the article, and the first Wikipedian reinstates the changes that he or she made to the previous version, prompting the second Wikipedian to revert to the previous version. [...] on Wikipedia, an edit war can go on indefinitely, making an article's edit history somewhat useless."

Ein Bearbeitungskonflikt in einem Artikel ist dementsprechend die wiederholte Zurücknahme inhaltlicher Änderungen anderer Benutzer, um eine frühere Version des Artikels mit den präferierten Inhalten wieder herzustellen. Alle zwischenzeitlichen Bearbeitungen am Artikel werden verworfen, bleiben aber in der Versionsgeschichte erhalten. Diese Konfliktform ist gegenwärtig in Wikis zu beobachten[1].

Im Folgenden werden die Ursachen für Bearbeitungskonflikte erörtert sowie eine allgemeine Definition dieser gegeben. Dabei werden verschiedene Konfliktformen voneinander abgegrenzt. Diese Diskussion geht weit über die verwandter Arbeiten hinaus, deren Autoren dieses Thema nur sekundär behandeln.

2.1 Ursachen für Bearbeitungskonflikte in Wikipedia

Zunächst werden Auslöser von Bearbeitungskonflikten diskutiert, bevor im Anschluss ein Überblick über existierende Möglichkeiten zur Konfliktvermeidung in Wikipedia gegeben wird.

[1] Das Auftreten von Bearbeitungskonflikten ist auch in Quellcode-Verwaltungssystemen wie dem CVS wahrscheinlich. Hier stehen Meinungsverschiedenheiten bezüglich präferierter Implementationen von Methoden im Vordergrund. Quellcode-Bearbeitungskonflikte sind nicht Gegenstand dieser Arbeit.

2.1.1 Auslöser von Bearbeitungskonflikten

Die in Kapitel 1.2 vorgestellte Versionsverwaltung von Wikipedia erlaubt es, Artikelversionen mit geringem Aufwand zurückzunehmen. Sie unterstützt Benutzer, Vorfälle von Vandalismus an Artikeln effizient zu reparieren. Das dieser niedrige Aufwand beim Zurücknehmen ebenso Bearbeitungskonflikte begünstigt, wird nachfolgend erörtert. Hierzu ein Beispiel.

Alice und Bob sind zwei Benutzer von Wikipedia. Bob ist ein Vandale. Er sucht sich gezielt Artikel aus und wendet dabei einige Zeit für das Einfügen unsinniger Änderungen auf. Alice wird kurze Zeit später beim Bearbeiten eines von Bob veränderten Artikels auf den Vandalismus aufmerksam. Sie zögert nicht lange und nimmt die Änderungen von Bob zurück, indem sie eine frühere Artikelversion innerhalb weniger Sekunden wieder herstellt. Da Vandalen oftmals nicht nur einen Artikel, sondern gleich mehrere ändern, schaut Alice sich die letzten Änderungen Bobs an. Sie erkennt in seinen Handlungen destruktive Absichten und repariert in der Folge mit wenig Aufwand die betroffenen Artikel.

Neus (2001) prägt in diesem Zusammenhang den Begriff der *künstlichen Informationsökonomie*[2]. Für Alice ist der Aufwand, Änderungen am Artikel rückgängig zu machen, deutlich geringer als der Aufwand für Bob, Bearbeitungen an diesem vorzunehmen. In der Folge verhindere dieser Umstand letztendlich nach Neus Artikel mit minderer Qualität und fördere stattdessen ein allgemein hohes inhaltliches Niveau.

Neus berücksichtigt in seiner Argumentation nicht, das die Zurücknahme nicht ausschließlich der Reparatur von Artikeln dienlich ist. In dieser Ausarbeitung wird die Hypothese aufgestellt, dass der marginale Aufwand dieser sogenannten Zurücknahmen zugleich ein Auslöser für Bearbeitungskonflikte ist. Im Gegensatz zu Vandalismus beziehen sich diese sachlich auf das jeweilige Thema des Artikels. Im persönlichen Umgang werden solche Meinungsverschiedenheiten ausdiskutiert. In Wikipedia stehen dafür Diskussionsseiten für jeden Artikel zur Verfügung. Vor allem verleitet die Einfachheit des Zurücknehmens fremder Textänderungen Benutzer dazu, bei Streitigkeiten auf eine zeitraubende, öffentliche Diskussion zu verzichten. Einzig die Kommentare, die bei jeder Bearbeitung des Artikels mit angegeben werden können, dienen dem Austausch von Argumenten oder auch von Beleidigungen. Prinzipiell rücken erstgenannte in den Hintergrund. Solange ein Benutzer fortwährend seine präferierten inhaltlichen Änderungen mit wenig Aufwand erneut wiederherstellen kann, hat er sein Ziel bereits erreicht. Eine Diskussion ist so-

[2] engl. „artifical information economy“

mit für ihn verzichtbar. Über den Ausgang von Meinungsverschiedenheiten entscheidet schließlich unserer Annahme nach einzig das Durchhaltevermögen eines Benutzers[3].

Nach Neus führt der geringe Aufwand für das Zurücknehmen unerwünschter Änderungen in der Folge zu qualitativen Artikeln. Die Diskussion zeigte, dass simultan das Entstehen von Bearbeitungskonflikten begünstigt wird.

2.1.2 Maßnahmen zur Schlichtung und Vermeidung von Konflikten

Benutzer eines Artikels fühlen sich oftmals durch Bearbeitungskonflikte Dritter gestört. Dies ist darin begründet, weil ihre Bearbeitungen am Artikel mit hoher Wahrscheinlichkeit durch das Zurücknehmen der Artikelversionen durch die Konfliktparteien wiederholt verworfen werden. Eine Weiterentwicklung des Artikels wird schließlich verhindert und die Qualität der betroffenen Artikel gemindert. Die Gemeinschaft von Wikipedia ist deshalb bestrebt, zwischen Konfliktparteien, die ohne Hilfe Dritter zu keinem Konsens kommen würden, frühzeitig zu vermitteln. Folgend werden einige Maßnahmen zur Schlichtung und Vermeidung von Konflikten vorgestellt.

Der Vermittlungsausschuss Sofern zwischen zwei oder mehr Benutzern längerfristig Streitigkeiten bestehen, wendet sich oftmals einer von beiden oder eine dritte Person, die auf die Meinungsverschiedenheit aufmerksam wurde, an den Vermittlungsausschuss. Ziel ist es, den Konflikt zu schlichten.

Der Vermittlungsausschuss ist eine inoffizielle Einrichtung in Wikipedia. Es werden „normale“ Benutzer, sogenannte Mediatoren, eingesetzt, um bei Streitigkeiten zu vermitteln. In der deutschen Wikipedia sind sie nicht fest gewählt, sondern helfen auf freiwilliger Basis. Mediatoren sind keine Administratoren und können nicht über weitere Maßnahmen wie das Sperren von Benutzern oder das Schützen von Artikeln vor Bearbeitungen verfügen. Ein Vermittler versucht die Meinungsverschiedenheiten in möglichst sachlicher Diskussion mit den beteiligten Benutzern zu schlichten. Hierbei wird oftmals ein themenbezogener Konsens ausgearbeitet. Bei den Ergebnissen des Vermittlungsausschusses handelt es sich ausschließlich um Empfehlungen an das Verhalten der beteiligten Benutzer.

Jedem Antrag auf Konfliktschlichtung wird eindeutig eine Seite in Wikipedia zugesprochen. Diese wird nach bestimmten Kriterien automatisch generiert und weist einen protokollarischen Charakter auf. Das Archiv des Ausschusses reicht bis 2003 zurück und

[3] Vorausgesetzt, kein Administrator wird auf den Konflikt aufmerksam und nimmt entsprechende Maßnahmen zur Schlichtung der Meinungsverschiedenheit vor.

umfasst derzeit in etwa 750 Fälle. Obwohl es sich bei den „Urteilen“ des Vermittlungsausschusses einzig um Empfehlungen handelt, wenden sich viele Benutzer in erster Instanz an das Komitee (Wikipedia, 2007g,h,q).

Das Schiedsgericht Das Schiedsgericht wird wie der Vermittlungsausschuss gleichermaßen nur auf Anfrage von Benutzern aktiv. Streitfälle sind von einem Benutzer selbst vorzutragen. Es wird daraufhin in einem Gremium mehrheitlich entschieden, ob der jeweilige Fall angenommen wird oder nicht.

Das Schiedsgericht stellt eine offizielle Institution von Wikipedia dar. Im Gegensatz zum Vermittlungsausschuss besteht die Kommission, welche eine vermittelnde Rolle einnimmt, aus zehn gewählten Mitgliedern. Jeweils fünf dieser werden nach sechs Monaten neu gewählt. Das Schiedsgericht definiert sich selbst als „letzte Instanz“ im Konfliktschlichtungsprozess.

Bei Annahme eines Streitfalls wird ein Vermittlungsverfahren mit ähnlichem Charakter wie beim Vermittlungsausschuss eingeleitet. Das Urteil ist jedoch für die beteiligten Benutzer bindend. Hier liegt in jeder Hinsicht der wesentliche Unterschied zwischen den beiden vorgestellten Einrichtungen. Mögliche Strafen umfassen die Verwarnung oder die zeitlich begrenzte Sperrung von Benutzern oder Administratoren.

Das Schiedsgericht wird in der deutschen Wikipedia kaum angenommen. Seit April 2007 wurden 16 Fälle abgeschlossen. Das englische Pendant bearbeitete innerhalb von drei Jahren mehrere hundert Fälle (Wikipedia, 2007u,s).

Die „Three-Revert“ Richtlinie Diese Regelung wurde zur Konfliktvermeidung eingeführt. Sie besagt, dass ein Benutzer, der Versionen eines Artikels an einem Tag mehr als dreimal zurücknimmt, von einem Administrator bis zu 24 Stunden gesperrt werden kann. Bevor ein Administrator jedoch die Sperrung veranlassen kann, muss der Bearbeitungskonflikt oder der wiederholte Vandalismus am Artikel manuell gesichtet worden sein (Wikipedia, 2007v). Benutzer, die eine Verletzung dieser Richtlinie beobachten, können eine Notiz auf einer dafür speziell angelegten Seite hinterlassen (Wikipedia, 2007m). Administratoren werden dadurch vermehrt auf Verstöße aufmerksam gemacht.

Ein Versuch in der deutschen Wikipedia bestätigte die Restriktivität, mit der diese Regel eingehalten wird. Drei Zurücknahmen innerhalb weniger Minuten für einen Artikel führten zu einer unbegrenzten Sperre des eigens zu diesem Versuch angelegten Benutzerkontos.

Die Administratoren in Wikipedia Administratoren sind mit zusätzlichen Rechten ausgestattet. Diese zeichnen sie gegenüber anderen Benutzern aus. Bezugnehmend zur Konfliktvermeidung sind sie in der Lage, Störenfriede zu sperren. In gleicher Weise können sie Artikel vor weiteren Bearbeitungen schützen. Diese Maßnahme wird oft bei Bearbeitungskonflikten vorgenommen, an denen viele Benutzer beteiligt sind und nicht sofort entschieden werden kann, welcher Benutzer zu keinem Konsens bereit ist. Ein sogenannter Seitenschutz konnte schon für mehrere Wochen ohne Unterbrechung beobachtet werden. Bearbeitungen am Artikel sind während dieser Zeit ausgeschlossen.

Schlussbemerkungen Die freie Enzyklopädie entwickelte im Laufe der Jahre funktionierende Mechanismen, um Interessenkonflikte durch die Dynamik der Gemeinschaft zu lösen. Elementar ist die Ausrichtung von Wikipedia, gegenwärtig nur reaktiv auf die Herausforderungen zu reagieren. Alle hier vorgestellten Maßnahmen setzen explizit die Erkennung eines Konflikts voraus. In der Konsequenz werden Vermittler der Streitschlichtungsverfahren nicht selbst aktiv. Benutzer müssen existierende Konflikte selbst vortragen. Da oftmals es keine Konfliktpartei als Notwendigkeit erachtet, einen der vorgestellten Ausschüsse zu unterrichten oder kein unparteiischer Dritter auf den Konflikt aufmerksam wurde, bleibt mit hoher Wahrscheinlichkeit ein Großteil der Auseinandersetzungen verborgen. Das in dieser Ausarbeitung vorgestellte Verfahren zur automatischen Erkennung von Bearbeitungskonflikten ist daher dem Vermittlungsprozess mit hoher Wahrscheinlichkeit förderlich.

Einzig die „Three-Revert“-Richtlinie ist als Heuristik heranzuziehen, um Bearbeitungskonflikte automatisch zu erkennen. Eine Prüfung auf drei Zurücknahmen, die innerhalb von 24 Stunden für Versionen der Versionsgeschichte durch einen Benutzer vorgenommen wurden, leistet dies.

Im Anhang B werden, neben Bearbeitungskonflikten und Vandalismus, weitere Herausforderungen von Wikipedia herausgestellt. Diese betreffen insbesondere die Qualität von Artikeln und deren inhaltlicher Neutralität.

2.2 Definition und Taxonomie von Bearbeitungskonflikten

In diesem Kapitel wird die in bisherigen Arbeiten eingeführte Definition sogenannter *globaler* Bearbeitungskonflikte kurz motiviert, um anschließend diese durch eine zweite zu erweitern. Dieser Schritt ist notwendig, da eine globale Erschließung des Begriffs „Bearbeitungskonflikt“ unzureichend ist, um ein Verfahren zur automatischen Erkennung dieser zu entwickeln. In der Folge sind *globale* und *lokale* Bearbeitungskonflikte zu unter-

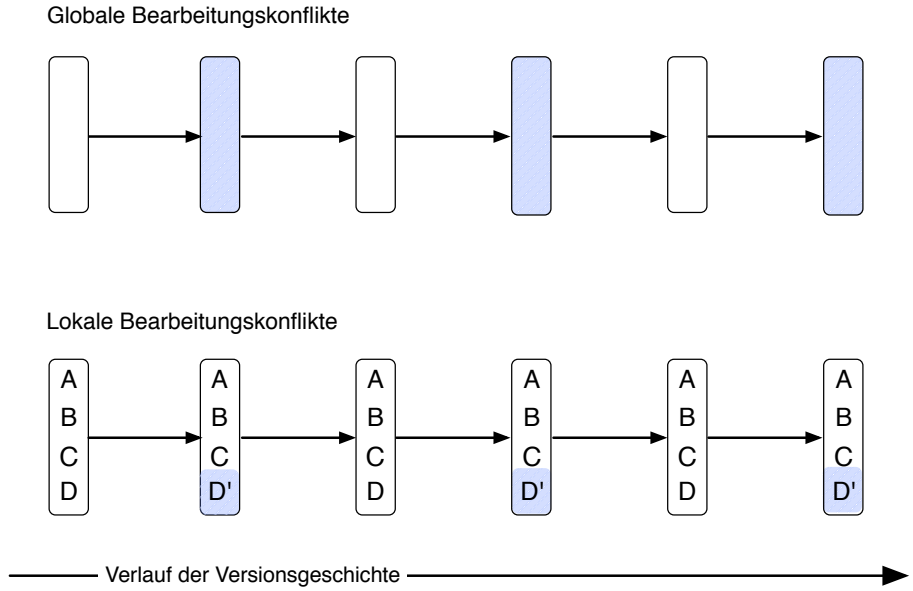

Abbildung 2.1 *: Unterschied zwischen globalen und lokalen Bearbeitungskonflikten. Im oberen Bildbereich ist ein globaler Bearbeitungskonflikt als Folge von Versionen dargestellt. Diese werden als Balken dargestellt, wobei zwei verschiedene Versionen im Wechselspiel auftreten. Bei einer globalen Betrachtung werden Versionen als Ganzes aufgefasst, es sind keine Informationen über die im Artikeltext enthaltenen Abschnitte enthalten. Diese werden dagegen bei einer lokalen Betrachtung im unteren Bildabschnitt berücksichtigt. Hier wird deutlich, dass nicht der vollständige Artikel vom Konflikt betroffen ist, sondern ausschließlich der Abschnitt D sowie der D ersetzende Abschnitt D′.*

scheiden. Erstgenannte betrachten den Text eines Artikels als Ganzes, wobei die lokale Definition berücksichtigt, dass ein Artikel grundsätzlich in Abschnitte unterteilt ist. Abbildung 2.2 stellt den Unterschied schematisch dar.

Zunächst jedoch ein fiktives Beispiel für einen Bearbeitungskonflikt, der im Verlauf der Ausarbeitung erweitert wird.

Beispiel für einen Bearbeitungskonflikt Alice und Bob, beides Benutzer von Wikipedia, sind sich nicht einig über die Formulierung eines Abschnitts in einem Artikel. Alice schreibt diesen in der Folge um. Bob ist mit ihren Bearbeitungen jedoch nicht einverstanden. Da er sich nicht mit Alice auf eine Diskussion einlassen will, verwirft er ihre Veränderungen wieder, in dem er die vorherige Version des Artikels wieder herstellt. Für einen Bearbeitungskonflikt typisch, gibt keiner der beiden Benutzer nach. Alice stellt erneut ihre Bearbeitungen wieder her, die wiederum von Bob durch Zurücknahme ihrer

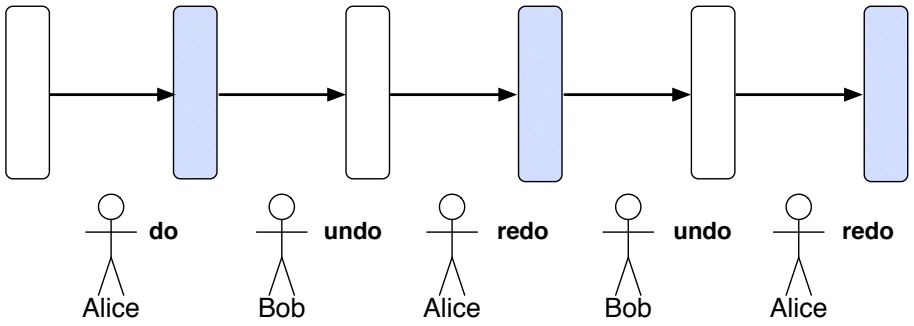

Abbildung 2.2 *: Darstellung des globalen Bearbeitungskonflikts auf Artikelebene zwischen Alice und Bob. Hierbei sind zwei Versionen, repräsentiert durch verschiedenfarbige Balken, am Konflikt beteiligt. Alice führt die Version erstmalig durch die Aktion* do *ein. Im weiteren Verlauf stellt sie diese durch* redo *wieder her, da ihre Änderungen von Bob rückgängig gemacht werden. Seine Aktionen sind durch* undo *gekennzeichnet.*

Version entfernt werden. Dieser Vorgang wiederholt sich noch einige Male, bis einer von beiden aufgibt.

2.2.1 Globale Bearbeitungskonflikte

In der Arbeit von Kittur u. a. (2007) werden Bearbeitungskonflikte auf einer globalen Ebene beschrieben, wobei die Versionen eines Artikels als Ganzes betrachtet werden.

Zunächst wird die vorgestellte Meinungsverschiedenheit zwischen Alice und Bob erweitert, damit sie durch eine globale Strukturerfassung erkennbar wird.

Erweiterung des Beispiels in Hinblick auf globale Bearbeitungskonflikte Alice und Bob nehmen gegenseitig ihre Versionen zurück. Dabei verwenden sie bei ihrem Streit exklusiv die *Zurücknahme-Funktion* von Wikipedia.

An dem Beispiel wird deutlich, dass exakt zwei Versionen an diesem Konflikt beteiligt sind. Da beide Benutzer keine weiteren Änderungen am Artikel vornehmen, ist eine Folge von Duplikaten in der Versionsgeschichte zu beobachten. Diese besteht einerseits aus der Version von Alice, die sich mit der von Bob erneut eingestellten Version abwechselt.

Eine globale Definition von Bearbeitungskonflikten setzt darum für die Erkennung dieser genau zwei identische Versionen, die jeweils wechselseitig zurückgenommen werden, voraus. Ein globaler Bearbeitungskonflikt lässt sich somit anhand eines im Folgenden vorgestellten globalen Musters in einem Artikel identifizieren.

Globales Konflikt-Muster Globale Bearbeitungskonflikte sind anhand eines Musters identifizierbar. Ein Konflikt besteht dabei aus vier oder mehr Zurücknahmen. Diese werden jeweils durch die Benutzeraktionen *undo* („rückgängig machen“) und *redo* („erneut ausführen“) repräsentiert. Diese bilden ein Paar (*undo,redo*). Die Einbeziehung der ersten Änderung (*do*), bei der die betroffene Version erstmalig angelegt wird, entfällt. Sie gilt als „normale“ Weiterentwicklung des Artikels und ist somit nicht Bestandteil eines Bearbeitungskonflikts.

Bearbeitungen, die von einem Benutzer vorgenommen und später durch ihn selbst wieder zurückgenommen werden, dieser also seine eigenen Änderungen verwirft, sind für die Betrachtung nicht relevant. Ein Bearbeitungskonflikt setzt schließlich zwei oder mehr Teilnehmer voraus. Die Definition einer Mindestlänge von vier dient als Abgrenzung gegenüber „normalen“ Bearbeitungen, die nur einmalig in der Versionsgeschichte vorkommen. Eine solche Version ist damit eindeutig anhand ihres Inhalts in der Versionsgeschichte bestimmbar, wogegen dies bei Versionsduplikaten nicht der Fall ist. Weiterhin werden einmalige Zurücknahmen, die bei der Reparatur von Vandalismus auftreten, nicht als Bearbeitungskonflikt erkannt. Das Muster für globale Bearbeitungskonflikte ist anhand der Anforderungen durch folgenden regulären Ausdruck zu beschreiben:

```
(undo,redo){2,*}(undo)?
```

Abbildung 2.2 veranschaulicht schematisch den Sachverhalt anhand des vorgestellten Bearbeitungskonflikts zwischen Alice und Bob.

Schlussbemerkungen Durch die Betrachtung auf Artikelebene werden Versionen im Ganzen untersucht. Im Ergebnis wird eine Menge an Versionen identifizierbar, die an einer Kontroverse beteiligt sind. Die Definition *globaler* Bearbeitungskonflikte erfasst somit den eigentlichen inhaltlichen, konstruktiven Konflikt unter Benutzern nicht. Die Textstelle, um die sich hier Alice und Bob streiten, wird nicht herausgestellt. Es bleibt nur festzustellen, dass ein Bearbeitungskonflikt existiert, aber um welches Thema sich die Benutzer streiten, ist unbekannt.

2.2.2 Lokale Bearbeitungskonflikte

Jeder Artikel ist in Abschnitte unterteilt, und damit auch seine Versionen. Eine Betrachtung einzelner Abschnitte erlaubt daher die Identifikation lokaler Bearbeitungskonflikte.

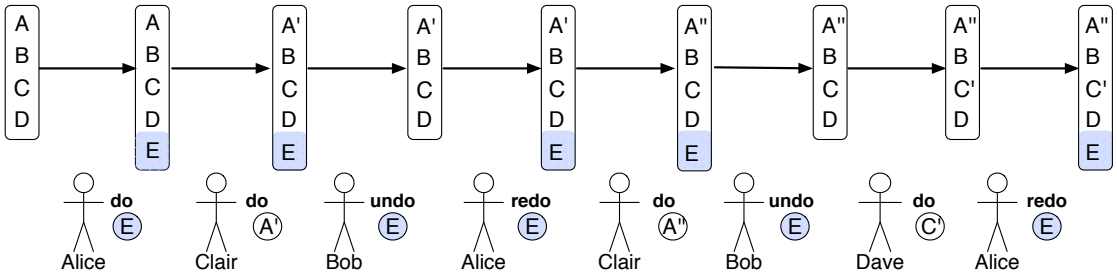

Abbildung 2.3 *: Darstellung eines lokalen Bearbeitungskonflikts. Zwei Benutzer, Alice und Bob, setzen gegenseitig ihre Änderungen bzgl. des Paragraphen* E *zurück. Andere Benutzer, Clair und Dave, entwickeln den Artikel zwischenzeitlich an anderen Stellen weiter; erkenntlich durch eine andere Bezeichnung der Paragraphen. Inhaltliche Änderungen an Paragraphen werden durch Hochkommata neben dem Buchstaben des Paragraphen angezeigt.*

Durch die vielen Änderungen, die an einem Artikel vorgenommen werden, treten Bearbeitungskonflikte oftmals nicht in einer geschlossenen Abfolge von direkt aufeinanderfolgenden (*undo,redo*)- sowie (*redo,undo*)-Paaren in der Versionsgeschichte auf. Benutzerkonflikte treten subtiler in Erscheinung als bislang angenommen, so dass es einer genaueren Betrachtung dieser Bedarf. Hierzu ein Beispiel.

Erweiterung des Beispiels in Hinblick auf lokale Bearbeitungskonflikte Während sich Alice und Bob um die Formulierung des Abschnitts streiten, bearbeiten zwei andere Benutzer, Clair und Dave, zwischenzeitlich andere Textstellen im Artikel. Alice und Bob nehmen auf die Änderungen der anderen Benutzer Rücksicht und fügen daher ihre konfliktbezogenen Änderungen in den Abschnitten als „normale“ Bearbeitung erneut ein. Sie verzichten dabei auf die Zurücknahme-Funktion. Konkret streiten sich Alice und Bob dabei um den Paragraphen E, der von Alice erneut eingefügt und von Bob wiederum entfernt wird. Zugleich nehmen Clair und Dave Änderungen in den Abschnitten A und C vor. Der Streit um Paragraph E wird nach vier Zurücknahmen, an denen dieser beteiligt ist, als lokaler Bearbeitungskonflikt identifiziert. Abbildung 2.3 veranschaulicht die neue Situation.

Würden Alice und Bob darüber hinaus auch die Bearbeitungen von Clair und Dave durch Verwendung der Zurücknahme-Funktion verwerfen, entstehen bei entsprechender Wiederholung zusätzliche lokale Bearbeitungskonflikte für die Paragraphen A und C.

Im Gegensatz zu globalen Bearbeitungskonflikten sind mehrere verschiedene Versionen am Streit beteiligt. Ein Wechselspiel zwischen zwei Versionsduplikaten liegt nicht länger vor, da sich der Artikel fortwährend an anderen Stellen ändert. Da der Konflikt immer

wieder durch Bearbeitungen anderer Benutzer unterbrochen wird, wird nach dem definierten globalen Konflikt-Muster eine Mindestlänge von vier Versionen seltener erreicht. Viele Bearbeitungskonflikte werden in der Folge nicht als solche erkannt.

Ähnlich wie für globale Bearbeitungskonflikte lässt sich ein Muster angeben, welches lokale Bearbeitungskonflikte beschreibt.

Lokales Konflikt-Muster Durch die Betrachtung einzelner Textabschnitte innerhalb von Versionen ist es möglich, Bearbeitungskonflikte auf einer lokalen Ebene versionsübergreifend zu verfolgen. Die Weiterarbeit am Artikel an anderen, nicht konfliktbezogenen, Textstellen beeinträchtigt die Erkennung nicht. Für die lokale Erkennung ist jedoch eine Erweiterung des globalen Konflikt-Musters notwendig, welches nun Zurücknahmen von Benutzern für einen bestimmten Abschnitt t im Text an einer Position i erfasst:

$$\texttt{(undo(}t_i\texttt{),redo(}t_i\texttt{)){2,*}(undo(}t_i\texttt{))?}$$

Die Positionsangabe dient dabei der lokalen Identifikation. Sie ist grundsätzlich für alle durch den Bearbeitungskonflikt betroffenen Abschnitte identisch, um diese von nicht konfliktbezogenen Bearbeitungen am Artikel abzugrenzen.

Bei der Betrachtung einzelner Abschnitte wird deutlich, dass hier die Paare (*undo,redo*) bzw. (*redo,undo*) gegensätzliche Operationen auf dem Text bezeichnen.

Taxonomie Lokale Bearbeitungskonflikte sind in ihrer Gestalt vielfältig. Alice ist in der Lage, neue Textfragmente hinzuzufügen, die Bob im nächsten Schritt wieder entfernt. Im Gegenzug schreibt Bob einen Satz um, den Alice wiederum zurücknimmt. Konkret existieren, entsprechend ihrer Textoperation, folgende Typen von Bearbeitungskonflikt:

(i) Benutzer fügen wiederholt Zeichenfolgen ein, die wiederum gelöscht werden.

(ii) Benutzer ersetzen wiederholt gegenseitig Zeichenfolgen im Artikel.

Lokale Bearbeitungskonflikte lassen sich ferner durch die Granularität der betroffenen Abschnitte kennzeichnen. Alice und Bob können sich um einzelne Paragraphen, Sätze, Wörter oder nur um Zeichen streiten. Resultierende Arten von Bearbeitungskonflikten aus einer Kombination von Textoperation und Granularität sind in einer Taxonomie in Abbildung 2.4 zusammengefasst.

Neben Bearbeitungskonflikten zwischen Benutzern sind Auseinandersetzungen zwischen Administratoren bekannt – sogenannte *Administrationskonflikte*. Administrationskonflikte sind nicht Bestandteil dieser Ausarbeitung, da einerseits die nötigen Informationen zur Erkennung nicht im Quelltext der Artikel verzeichnet sind, sondern in der

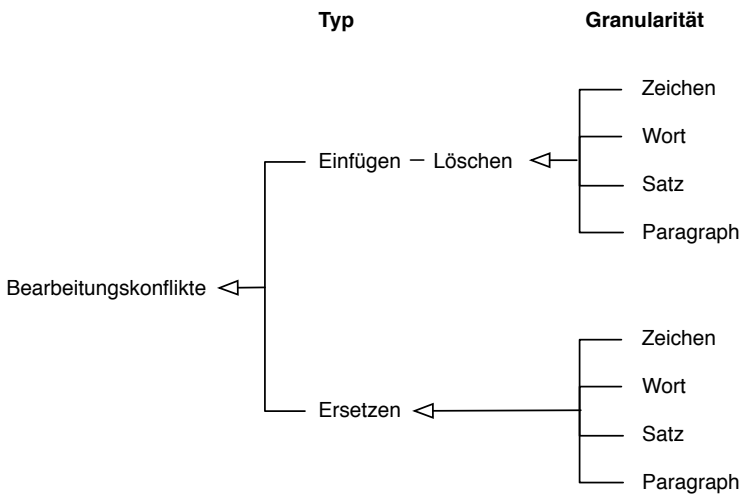

Abbildung 2.4 *: Taxonomie für lokale Bearbeitungskonflikte.*

Wikipedia zugrundeliegenden Datenbank vorliegen. Zugleich handelt es sich nicht länger um inhaltlich, konstruktive Konflikte. Administrationskonflikte äußern sich beispielsweise durch folgende Aktionen:

(i) Administratoren sperren und schalten wiederholt Benutzerkonten frei.

(ii) Administratoren löschen und fügen wiederholt Artikel ein.

Abbildung 2.5 stellt alle beobachteten Arten in einer Taxonomie zusammen.

Schlussbemerkungen Zusammengefasst verändern sich die Versionen eines Artikels stetig. Zugleich kann ein Bearbeitungskonflikt an einer bestimmten Position im Text ausgetragen werden. Versionen, die an der Meinungsverschiedenheit tatsächlich beteiligt sind, sind somit nicht mehr per Definition identisch. Dieser lässt sich somit global anhand von Versionsduplikaten nicht mehr geschlossen erkennen, wenn die Konfliktparteien nicht explizit die Zurücknahme-Funktion von Wikipedia verwenden. Anstatt ein andauerndes Wechselspiel zwischen zwei Duplikaten vorzufinden, sind nun viele verschiedene Versionen vom Konflikt betroffen. Sie sind nur ein oder zweimal Bestandteil eines solchen Konflikts. Ein rückblickender Vergleich mit Abbildung 2.3 zeigt, dass eine globale Erkennung einzig in Versionen drei und fünf ein Duplikat ermittelt. Ein automatisches Verfahren, welches ausschließlich globale Bearbeitungskonflikte erkennt, scheitert hier.

Die Vorteile einer Betrachtung lokaler Bearbeitungskonflikte liegen auf der Hand. Zum einen wird erstmalig die konfliktbezogene Textstelle herausgearbeitet, andererseits Bearbeitungen am Artikel toleriert. Somit sind lokale Bearbeitungskonflikte, die nur an einer

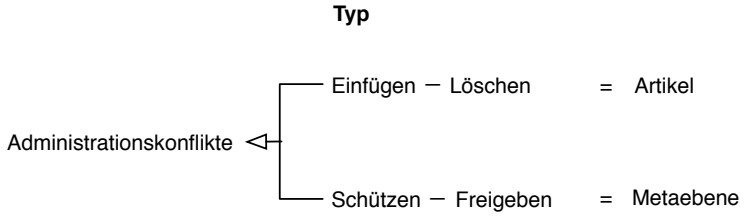

Abbildung 2.5 *: Taxonomie für Administrationskonflikte. Die Metaebene steht stellvertretend für Informationen, die in einer Datenbank abgelegt und nicht im Quelltext der Artikel verzeichnet sind.*

bestimmten Stelle im Text existieren weiterhin vollständig erfassbar. Dies wird durch Reduzieren der Betrachtung von Versionen erreicht, indem nun explizit Abschnitte betrachtet werden. In der Konsequenz ist der konfliktbezogene Abschnitt eines Bearbeitungskonflikts direkt anzugeben und eine Erkennung von mehreren Konflikten innerhalb einer Version wird möglich.

3 Automatische Erkennung von Bearbeitungskonflikten

In diesem Kapitel wird der *Edward*-Algorithmus vorgestellt. Die automatische Erkennung von Bearbeitungskonflikten geschieht dabei in zwei Schritten (vgl. Abbildung 3.1).

Im ersten Schritt werden die Versionen eines Wikipedia-Artikels mit einem sogenannten Differenzmodell repräsentiert. Dieses modelliert den Unterschied einer Version zu seiner Vorgängerversion. Ein Versionspaar wird im Ergebnis durch eine Menge von Textoperationen beschrieben, die eine Version in die nächste überführen. Die Versionsgeschichte eines Artikels wird somit durch eine sogenannte Textoperationsgeschichte beschrieben.

Im zweiten Schritt wird auf der Menge der Textoperationen ein Data-Mining durchgeführt. Ziel ist es, lokale Konflikt-Muster zu identifizieren. Erkannte Konflikt-Muster bilden schließlich die Menge automatisch entdeckter Bearbeitungskonflikte für einen Artikel.

Zunächst wird das Differenzmodell in Kapitel 3.1 diskutiert. Der Schritt der Mustererkennung wird anschließend in Kapitel 3.2 präsentiert. Ähnlichkeitsmetriken, die bei der Mustererkennung eingesetzt werden, diskutiert Kapitel 3.3, bevor abschließend in Kapitel 3.4 frühere Arbeiten vorgestellt werden.

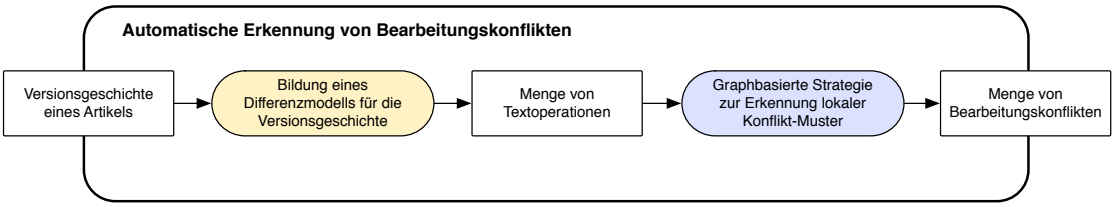

Abbildung 3.1 *: Die Erkennung von Bearbeitungskonflikten als Zwei-Schritt-Prozess.*

3.1 Bildung eines Differenzmodells für die Versionsgeschichte eines Artikels

Für die Erkennung lokaler Bearbeitungskonflikte interessieren vor allem die Abschnitte eines Artikels. Sie bilden die Menge der Textbereiche, die potentiell Bestandteil von Bearbeitungskonflikten sind. Dabei sind einzig jene von Bedeutung, die von Benutzern bearbeitet werden, sich also zwischen zwei Versionen ändern. Die Analyse der gesamten Versionsgeschichte bei der automatischen Erkennung von Bearbeitungskonflikten ist in der Konsequenz auf die Unterschiede von Versionspaaren beschränkbar, ohne einen Informationsverlust hinnehmen zu müssen. Die Komplexität der Analyse vollständiger Versionen wird dadurch gleichermaßen auf wenige Textbereiche pro Versionspaar reduziert.

Ein Differenzalgorithmus leistet die geforderte Differenzbildung. Er nimmt zwei Zeichenfolgen als Eingabe entgegen und liefert eine Menge von Textoperationen als Ausgabe zurück. Diese zeigen bei einer Gegenüberstellung die Unterschiede der beiden Zeichenfolgen auf. Sind beide Zeichenfolgen identisch, ist die Menge leer. Abbildung 3.2 zeigt das Ergebnis einer zeilenweisen Differenzberechnung mittels des Programms *GNU diff*. Als Eingabe dienten zwei Versionen eines Artikels, um die sich die zwei Benutzer von Wikipedia, Alice und Bob, streiten.

In diesem Kapitel werden zunächst Grundlagen der Differenzberechnung vorgestellt. Im Anschluss wird diskutiert, welches Verfahren zur Differenzberechnung der vollständigen Versionsgeschichte eines Artikels einzusetzen ist.

Die Differenz zwischen zwei Texten $A = a_1, a_2, \ldots, a_m$ und $B = b_1, b_2, \ldots, b_n$ soll ermittelt werden. Dabei sind a und b Symbole aus einem beliebigen Alphabet Σ, es gilt $m \leq n$. Zunächst wird eine dritte Zeichenkette $LCS = s_1, s_2, \ldots, s_l$ mit $l \leq m$ gesucht. Sie stellt die gemeinsame Teilsequenz von A und B dar, wobei l maximale Länge besitzt. Alle Zeichen von A und B, die nicht in der gemeinsamen Sequenz LCS enthalten sind, wurden entweder aus A gelöscht oder in B neu eingefügt. Auf diese Weise lassen sich Unterschiede zweier Texte ermitteln. Das Finden einer solchen Teilsequenz ist als *Longest-Common-Subsequence*-Problem (LCS) bekannt.

Das LCS-Problem gilt als Spezialfall des *Editierdistanz*-Problems. Hierbei wird eine minimale Anzahl an Textoperationen gesucht, die eine Zeichenkette A in eine Zeichenkette B überführen. Die Menge der elementaren Operationen wird auf Einfügen, Löschen und Ersetzen beschränkt – jedem Operationstyp werden Kosten zugesprochen. Die Edi-

Version von Alice (a.txt)

Die Wikipedia ist ein Projekt freiwilliger Autoren zum Aufbau einer Enzyklopädie und nichts anderes. Die Artikel sollen ausschließlich bedeutsames Wissen aus belegten und zuverlässigen Quellen enthalten. Der Name Wikipedia setzt sich zusammen aus wikiwiki, dem hawaiischen Wort für „schnell“, und „encyclopedia“, dem englischen Wort für „Enzyklopädie“.
Ein Wiki ist eine Webseite, deren Seiten jedermann leicht und ohne technische Vorkenntnisse direkt im Internetbrowser ändern kann.

Version von Bob (b.txt)

Die im März 2001 gegründete deutschsprachige Wikipedia ist ein Projekt freiwilliger Autoren zum Aufbau einer Enzyklopädie und nichts anderes. Die Artikel sollen bedeutsames Wissen aus belegten und zuverlässigen Quellen enthalten. Der Name Wikipedia setzt sich zusammen aus wikiwiki, dem hawaiischen Wort für „schnell“, und „encyclopedia“, dem englischen Wort für „Enzyklopädie“. Ein Wiki ist eine Webseite, deren Seiten jedermann leicht und ohne technische Vorkenntnisse direkt im Internetbrowser ändern kann.

```
Keksdose:Desktop hoppe$ diff a.txt b.txt
1,2c1,2
< Die Wikipedia ist ein Projekt freiwilliger Autoren zum Aufbau einer Enzyklopädie und nichts anderes.
< Die Artikel sollen ausschließlich bedeutsames Wissen aus belegten und zuverlässigen Quellen
enthalten.
---
> Die im März 2001 gegründete deutschsprachige Wikipedia ist ein Projekt freiwilliger Autoren zum Aufbau
einer Enzyklopädie und nichts anderes.
> Die Artikel sollen bedeutsames Wissen aus belegten und zuverlässigen Quellen enthalten.
```

***Abbildung 3.2** : Mittels* GNU diff *wird ein Versionspaar eines Artikels miteinander verglichen. Das Programm stellt dabei eine Ersetzung (c, „change“) in der ersten und zweiten Zeile fest. Jeder Satz entspricht einer Zeile im Dokument (hier zusammenhängend dargestellt). Eine Ersetzung äußert sich durch Löschen (<) und Einfügen (>) einer neuen Zeichenfolge an der gleichen Position. Im Text sind die eigentlichen Unterschiede farblich umrandet .*

tierdistanz D zwischen zwei Zeichenketten A, B wird schließlich als Summe der minimal benötigten Kosten definiert.

Eine Menge von Textoperationen, um eine Zeichenkette in eine andere überführen, existiert immer. Dies ist unabhängig davon, ob sich die Zeichenketten ähnlich sind oder nicht. Die Herausforderung besteht darin, eine Menge an Textoperationen zu finden, für die die Kosten minimal sind.

Wird die Ersetzen-Operation vernachlässigt und für Einfügen und Löschen Kosten von 1 veranschlagt, so lässt sich das LCS-Problem durch die Editierdistanz darstellen. Um A in B zu überführen, wird zunächst $LCS(A, B)$ bestimmt. Dies geschieht, indem $m - l$ Zeichen aus A entfernt werden. Anschließend werden zu der entstehenden Zeichenkette $n - l$ Zeichen von B hinzugefügt. Die Editierdistanz D ist schließlich durch $D(A, B) = m + n - 2l$ gegeben. Tabelle 3.1 verdeutlicht nochmals den Zusammenhang zwischen LCS und der Editierdistanz D anhand eines Beispiels.

Position		1 2 3 4 5 6 7
A	=	c b a b a c
B	=	a b c a b b a
LCS(A,B)	=	_ _ c b a b _ a c
		a b c _ a b b a _
	⇒	c a b a
D(A,B)	=	Pos.1: a löschen
		Pos.2: b löschen
		Pos.3: b einfügen
		Pos.6: b löschen
		Pos.7: c einfügen
	⇒	5 Operationen, D(A,B) = 5

Tabelle 3.1 *: Das Beispiel verdeutlicht für zwei Zeichenfolgen A und B den Zusammenhang zwischen der längsten gemeinsamen Teilsequenz (LCS) und dem Finden einer minimalen Menge an Textoperationen, um A in B zu überführen. Für die Operationen werden einheitlich Kosten von 1 veranschlagt. Grüne Buchstaben sind in A und B identisch, rote nicht.*

Versionspaare, die bislang vollständig durch ihren Text repräsentiert werden, sind somit gleichermaßen durch eine minimale Menge von Textoperationen zu beschreiben.

Die Zeitkomplexität zur Lösung des LCS-Problems beträgt im ungünstigsten Fall $O(mn)$. Da beide Texte vollständig miteinander verglichen werden müssen, ist eine Zunahme des Berechnungsaufwands bei sehr langen Texten festzustellen. Ein Vergleich eines sehr langen Textes mit einer kurzen Zeichenfolge ist ebenfalls nicht effizient lösbar (Hirschberg, 1975).

Bergroth u. a. (2000) bieten einen Überblick über bekannte Ansätze und Algorithmen, um das LCS-Problem zu lösen. Als effizienteste Methode gilt im Ergebnis das in Myers (1986) entwickelte Verfahren. Dieses kombiniert beide Herausforderungen, LCS sowie die Bestimmung der Editierdistanz, miteinander. Der Algorithmus[1] von Myers gehört zu der Klasse der Greedy-Algorithmen. Diese wählen in jedem Berechnungsschritt die Entscheidung, die zu dem Zeitpunkt der Wahl den maximalen Gewinn verspricht.

[1] Die aktuelle Implementation von *GNU diff* basiert auf dem Verfahren von Myers (1986).

Es handelt sich jeweils um die Wahl eines lokalen Optimums. Die schrittweise optimale Teillösung eines Problems führt nicht zwangsläufig zu einem globalen Optimum.

Myers verbessert auf diese Weise die Zeitkomplexität zur Lösung des LCS-Problems auf $O(D(m+n))$. In der Folge arbeitet dieser Algorithmus effizient, wenn beide Zeichenketten sehr ähnlich sind – D minimal wird. Dies wird für aufeinanderfolgende Versionen eines Wikipedia-Artikels angenommen. Im Gegensatz zu zwei zufällig gewählten Texten sind zwei benachbarte Versionen eines Artikels mit hoher Wahrscheinlichkeit sehr ähnlich. Der Kontext, in dem sie stehen, ändert sich nicht. Die Menge der Unterschiede ist klein, da Benutzer im Durchschnitt nur wenige Änderungen vornehmen. Es wird in der Folge angenommen, dass bei der Differenzberechnung der Versionsgeschichte die Editierdistanz D minimal wird und der vorgestellte Algorithmus von Myers die richtige Wahl darstellt.

Laufzeitanalyse zur Bildung des Differenzmodells Wie bereits gezeigt, beträgt der Aufwand der Differenzberechnung mittels des Verfahrens von Myers (1986) für zwei Texte $O(D(m+n))$. Die Editierdistanz D gibt die Anzahl benötigter Textoperationen an, um eine Version in die nachfolgende zu überführen. Sei R die Versionsgeschichte eines Artikels. Für $|R|$ Versionen eines Wikipedia-Artikels werden insgesamt $(|R|-1)$-mal Differenzen zwischen jeweils einer Version und der nächsten berechnet. Es ergibt sich schließlich eine Gesamtlaufzeit von $O(D(m+n)(|R|-1))$.

Eine im Rahmen der Evaluierung durchgeführte Erhebung für Artikel der englischen Wikipedia ergab $D = 3$.

Die Laufzeit der Differenzberechnung ist weiterhin von der gewählten Granularität der zu bildenden Textoperationen abhängig. Ein Differenzalgorithmus kann zwei Texte Zeichen für Zeichen, aber auch paragraphbasiert miteinander vergleichen. Die Granularität der Differenzberechnung nimmt somit Einfluss auf die entstehenden Differenzen. Je feiner die Granularität gewählt wird, desto genauer sind Unterschiede zu ermitteln. Der Berechnungsaufwand wächst durch Wahl einer feineren Granularität. Differenzalgorithmen arbeiten generell zeilenbasiert und vergleichen darum Texte effizient miteinander – es werden jedoch nur zeilenweise Differenzen berechnet.

Eine durchgeführte manuelle Sichtung von Bearbeitungskonflikten zeigt in diesem Zusammenhang, dass sich häufig um einzelne Wörter oder Satzteile gestritten wird. Eine zeilenweise Differenzberechnung ist hier zu ungenau. Eine Ausrichtung auf Sätze oder Paragraphen würde derartige Bearbeitungskonflikte ebenfalls nicht ausreichend genau erfassen. Eine Spezialisierung auf Wörter schließt die Erkennung längerer Zeichenfolgen indes nicht aus, so dass im Ergebnis Wörter Grundlage der Differenzberechnung bilden. Die Genauigkeit bei der Erfassung von Bearbeitungskonflikten steht im Vordergrund.

Der Algorithmus von Myers wird als Ergebnis der Betrachtungen beim *Edward*-Algorithmus zur Modellbildung eingesetzt. Für die vollständige Versionsgeschichte werden jeweils Differenzen zwischen aufeinanderfolgenden Versionspaaren berechnet. Die Versionsgeschichte wird schließlich durch eine Textoperationsgeschichte repräsentiert. Diese ist bei der automatischen Erkennung von Bearbeitungskonflikten im weiteren Verlauf auf lokale Konflikt-Muster zu untersuchen.

3.2 Der Erkennungs-Algorithmus *Edward*

Die grundlegende Funktionsweise des *Edward*-Algorithmus wurde bereits einleitend in Kapitel 3, siehe Abbildung Abbildung 3.1, motiviert. Der erste Schritt der Differenzbildung wurde im vorangegangen Kapitel diskutiert. Als Ergebnis wird die Versionsgeschichte durch eine Textoperationsgeschichte repräsentiert.

Die Textoperationsgeschichte eines Artikels wird nachfolgend vollständig auf lokale Konflikt-Muster untersucht. Diese beschreiben durch eine Folge lokaler Zurücknahmen, die sich auf einen Textabschnitt an einer bestimmten Position im Text beziehen, Bearbeitungskonflikte. Ziel der automatischen Erkennung von Bearbeitungskonflikten ist es, nach jedem neuen untersuchten Versionspaar alle für einen Wikipedia-Artikel erfassten Bearbeitungskonflikte angeben zu können. Kapitel 3.2.1 stellt die entsprechende Herangehensweise des *Edward*-Algorithmus vor. Diese basiert darauf, einen Graphen aufzustellen, bei dem Knoten Differenzoperationen repräsentieren und Kanten Operationen an denselben Textstellen in unterschiedlichen Versionen verknüpfen.

Neben der Aufgabe, eine große Menge an Textoperationen zu analysieren, ergibt sich bei der genaueren Betrachtung lokaler Bearbeitungskonflikte eine zweite.

Bei der, in einem späteren Kapitel vorgestellten, Dokumentation von Bearbeitungskonflikten wurde festgestellt, dass sich Benutzer für die Dauer dieser nicht ausschließlich um eine identische Zeichenfolge streiten. Der vom Streit betroffene Textabschnitt wird von den Benutzern stellenweise umgeschrieben. Dies geschieht einerseits, um direkte Zurücknahmen[2] zu vermeiden oder andererseits, um auf die andere Konfliktpartei inhaltlich zuzugehen. Weiterhin mag ein Benutzer unzufrieden mit seiner eigenen Formulierung sein und formuliert aus diesem Grund den betroffenen Abschnitt um.

[2] Aufgrund der „Three-Revert“ Richtlinie kann ein Benutzer, nachdem er dreimal innerhalb von 24 Stunden einen Artikel zurückgenommen hat, für maximal 24 Stunden gesperrt werden. Einer Zurücknahme wird in Wikipedia besondere Aufmerksamkeit geschenkt. Konfliktparteien ziehen es daher vor, unerkannt zu bleiben, da sonst ein Administrator auf die Kontroverse aufmerksam wird und durch entsprechende Konfliktvermeidungsmaßnahmen sie selbst für weitere Bearbeitungen sperren könnte.

Es wird somit die Hypothese aufgestellt, dass Bearbeitungskonflikte nicht ausschließlich auf identische, konfliktbezogene Textabschnitte begrenzt sind – sich der Text im Verlauf eines Konflikts ändert. Zur vollständigen Erfassung von Bearbeitungskonflikten ergibt sich schließlich zusätzlich die Aufgabe, für zwei gegensätzliche Textoperationen, deren Zeichenfolgen zu einem hohen Maße ähnlich sind, eine lokale Zurücknahme festzustellen. Um auch solche Bearbeitungskonflikte festzustellen, verwenden wir eine Ähnlichkeitsfunktion, die die inhaltliche Ähnlichkeit zwischen zwei Texten misst. Somit werden auch leichte Textveränderungen im Laufe eines Bearbeitungskonflikts erkennbar.

Zunächst jedoch folgt die Vorstellung der angesprochenen graphbasierten Strategie zur automatischen Erkennung von Bearbeitungskonflikten.

3.2.1 Graphbasierte Strategie zur Erkennung lokaler Konflikt-Muster

Zur automatischen Erkennung lokaler Bearbeitungskonflikte wird ein *Data-Mining* auf der Textoperationsgeschichte durchgeführt. Data-Mining beschreibt dabei eine Klasse von Verfahren, bei denen durch Analyse von Datensammlungen relevante Informationen gewonnen werden. Die Analyse beruht in diesem Fall auf einer Mustererkennung, bei der Übereinstimmungen mit lokalen Konflikt-Mustern in der Textoperationsgeschichte gesucht werden.

Die einzelnen Schritte der Mustererkennung verdeutlicht die in Abbildung 3.3 dargestellte Aktivität. Sie erweitert das Aktivitätsdiagramm des *Edward*-Algorithmus von Seite 18. Für jedes Versionspaar des Artikels sind folgende Schritte zu durchlaufen, wobei die zwei letztgenannten sich unter dem Begriff der Mustererkennung zusammenfassen lassen in diesem Kapitel diskutiert werden:

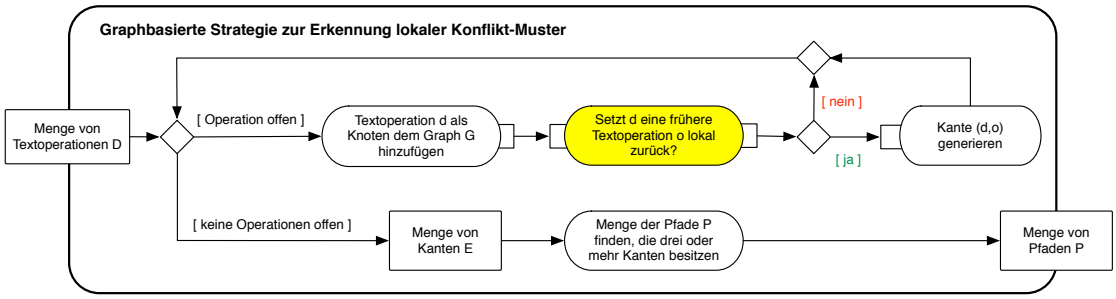

Abbildung 3.3 *: Der Prozess der Mustererkennung basierend auf einem gerichteten, azyklischen Graphen $G = (V, E)$. Jede Textoperation wird dem Graph als Knoten hinzugefügt. Es wird dabei geprüft, ob sie eine gegensätzliche Textoperation im Graph zurücknimmt. Ist dies der Fall, werden beide durch eine Kante verknüpft. Die Menge der Pfade mit drei oder mehr Kanten repräsentiert Bearbeitungskonflikte.*

(i) Bildung eines Differenzmodells für ein Versionspaar eines Artikels. Eine Menge von Textoperationen wird erzeugt. Diese werden als Knoten einem Graph hinzugefügt.

(ii) Jede Textoperation wird mit zuvor erzeugten Textoperationen anderer Versionspaare des Artikels verglichen. Es wird geprüft, ob die neue Textoperation eine vorherige zurücknimmt – sich also eine lokale Zurücknahme ergibt. Ist dies der Fall, werden beide Textoperationen durch eine gerichtete Kante verbunden.

(iii) Nach Verarbeitung aller Textoperationen des Versionspaares werden Pfade mit drei oder mehr Kanten gesucht. Die in solchen Pfaden enthaltenen Textoperationen stimmen mit dem lokalen Konflikt-Muster überein und repräsentieren somit die Menge der Bearbeitungskonflikte.

Durch Analyse weiterer Versionspaare eines Artikels sind Bearbeitungskonflikte sukzessiv erweiterbar. Die Analyse ist für einen Artikel beendet, sobald alle Versionspaare in zeitlich aufsteigender Reihenfolge verarbeitet wurden. Nach jedem Schritt sind durch die Menge der Pfade mit drei oder mehr Kanten alle Bearbeitungskonflikte festzustellen.

Eine Pseudocode-Darstellung des *Edward*-Algorithmus auf Seite 26 stellt insbesondere die graphbasierte Vorgehensweise bei der Erkennung von Bearbeitungskonflikten nochmals dar. Dabei wird der Begriff „offener“ Textoperationen eingeführt. Wenn eine für ein Versionspaar neu erzeugte Textoperation vom Algorithmus betrachtet wurde und keine lokale Zurücknahme festgestellt werden konnte, wird sie als offen bezeichnet. Offene Textoperationen sind mehrmals zu untersuchen, da sie im weiteren Verlauf der Analyse der Versionsgeschichte durch spätere Textoperationen zurückgenommen werden können. Es handelt sich also bei offenen Textoperationen um Operationen, die nicht an einem Bearbeitungskonflikt beteiligt und somit nicht mit einer anderen Textoperationen durch eine Kante verknüpft sind. Ausschließlich offene Textoperationen sind bei der Suche nach lokalen Zurücknahme relevant.

Im folgenden Kapitel 3.2.2 wird die Funktionsweise des *Edward*-Algorithmus mittels des vorgestellten Bearbeitungskonflikts zwischen Alice und Bob vertieft, bevor im Anschluss in Kapitel 3.2.3 Optimierungen für die Suche nach lokalen Zurücknahmen vorgestellt werden. Kapitel 3.2.4 schließt mit einer Laufzeitanalyse ab.

Algorithmus 1 : Graphbasierte Erkennung lokaler Bearbeitungskonflikte

Eingabe : Versionsgeschichte R eines Artikels, der untersucht werden soll
Ausgabe : Menge C der Pfade im Graph, die Bearbeitungungskonflikte beschreiben
Sei $G = (V, E)$ ein direkter, azyklischer Graph;
Sei O die Menge offener Textoperationen;
Sei S die Menge aller Startknoten für Pfade in G;
begin
  $V \leftarrow \emptyset$, $E \leftarrow \emptyset$, $O \leftarrow \emptyset$, $S \leftarrow \emptyset$, $C \leftarrow \emptyset$;
  `/** Suche nach Bearbeitungskonflikten **/`
  foreach *Versionsspaar* $(r_i, r_{i+1}) \in R$ **do**
    Berechne minimale Menge D an Textoperationen, um r_i nach r_{i+1} zu überführen;
    foreach *Textoperation* $d \in D$ **do**
      $V \leftarrow d \cup V$;
      `/** Aktualisierung des Graphen **/`
      foreach *offene Textoperation* $o \in O$ **do**
        if *Textoperation* d *gegensätzlich zu* o **then**
          $E \leftarrow (o, d) \cup E$;
          $O \leftarrow O \setminus o$;
          if $N_{G^-}(o) = \emptyset$ **then**
            $S \leftarrow o \cup S$;
          end
          Verlasse die innerste for-Schleife;
        end
      end
    end
    $O \leftarrow D \cup O$;
  end
  `/** Verfolgung relevanter Pfade in` G `**/`
  foreach $s \in S$ *mit* $|N_{G^+}(s)| \geq 4$ **do**
    Bilde die Menge aller Knoten des Pfades P mit dem Startknoten s;
    $C \leftarrow P \cup C$;
  end
  return C;
end

3.2.2 Anwendung des *Edward*-Algorithmus anhand eines Beispiels

Die automatische Erkennung von Bearbeitungskonflikten wird in diesem Kapitel anhand eines konkreten Benutzerkonflikts vorgeführt. Hierzu wird der in Kapitel 2.2 eingeführte und in Kapitel 2.2.2 erweiterte Benutzerkonflikt, bei dem sich Alice und Bob um die Aufnahme des Wortes „ausschließlich“ streiten, um konkrete Textoperationen ergänzt. Alle Textoperationen, die das Beispiel erweitern, sind in Tabelle 3.2 festgehalten.

Darstellung von Textoperationen Eine Textoperation wird als Tupel bestehend aus einer Zeichenfolge, einer Position im Text, an der die Zeichenfolge steht und der jeweiligen Operation – Hinzufügen (>), Löschen (<) – dargestellt:

```
( Zeichenfolge, Position, Operation )
```

Handelt es sich um eine Ersetzung, wird sie durch ein Paar Einfügen–Löschen wie folgt charakterisiert, wobei sich die Zeichenfolge unterscheidet:

```
{ ( Zeichenfolge, Position, < ), ( Zeichenfolge, Position, > ) }
```

Für das angeführte Beispiel aus Abbildung 3.2 von Seite 20 ergeben sich auf Wortebene als minimale Einheit folgende Differenzen:

(i) `( ausschließlich, 18, < )`

(ii) `( im März 2001 gegründete deutschsprachige, 2, > )`

Die Textoperation des Ersetzens wird hier nicht länger als solche erkannt, weil die Granularität feiner gewählt wurde. Die Differenzberechnung wird genauer. Die Ersetzungsoperation weicht den tatsächlichen Operationen Einfügen und Löschen.

Abbildung 3.4 stellt den Bearbeitungskonflikt zwischen Alice und Bob in einem Graph dar. Der Konflikt umfasst die Differenzen (1), (4), (9), (10) und (12). Durch die entsprechende Verknüpfung zweier Textoperationen unterschiedlicher Versionen mittels einer Kante bei Feststellung einer lokalen Zurücknahme ergibt sich schrittweise ein Pfad der Länge vier.

Vorbemerkung Im Folgenden wird die Vorgehensweise bei der automatischen Erkennung des Bearbeitungskonflikts zwischen Alice und Bob diskutiert. Dabei werden die Formalismen verwendet, die in der Pseudocode-Darstellung des *Edward*-Algorithmus auf Seite 26 eingeführt wurden.

Nr.	Version	Textoperationen	Symbol
(1)	1001 (Alice)	`( ausschließlich, 18, > )`	
(2)	1004 (Clair)	`( Wissen, 20, < )`	
(3)		`( Sachkenntnis, 20, > )`	
(4)	1007 (Bob)	`( ausschließlich, 18, < )`	
(5)	1008 (Clair)	`( bedeutsames, 19, < )`	
(6)		`( bedeutsame, 19, > )`	
(7)	1011 (Dave)	`( Webseite, 51, < )`	
(8)		`( Internetseite, 51, > )`	
(9)	1014 (Alice)	`( ausschließlich, 18, > )`	
(10)	1017 (Bob)	`( ausschließlich, 18, < )`	
(11)		`( Ausnahmen sind strittige Artikel, 65, > )`	
(12)	1208 (Alice)	`( ausschließlich, 18, > )`	

Tabelle 3.2 *: Erweiterung des Bearbeitungskonflikts zwischen Alice und Bob aus Kapitel 2.2. Zusätzlich zur Differenzdarstellung hilft eine spezifische Versionsnummer von Wikipedia, die Änderung eindeutig zu bestimmen. Jeder Textoperation wird ein Symbol zugeordnet, welches die Differenz in den folgenden Abbildungen referenziert. Die dazugehörige Nummer dient dem textlichen Verweis.*

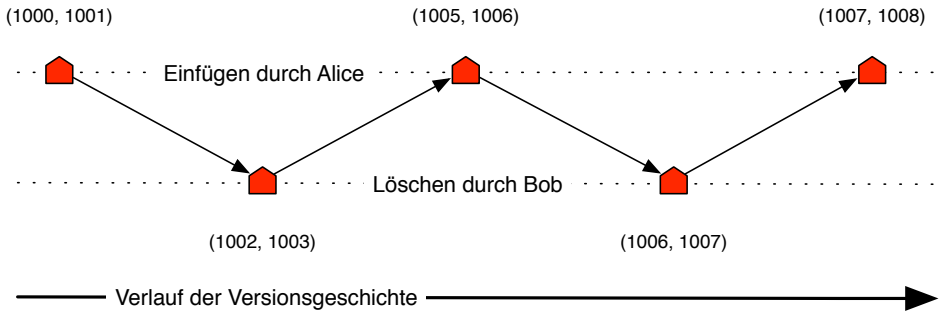

Abbildung 3.4 *: Ausgehend von dem Beispiel in Tabelle 3.2 wird der Bearbeitungskonflikt zwischen Alice und Bob mittels eines gerichteten Graphen $G = (V, E)$ dargestellt. Die roten Symbole verweisen dabei auf die Textoperation aus Tabelle 3.2, die im Wechsel von Alice vorgenommen und von Bob zurückgenommen wird.*

Für das Versionspaar $(1000, 1001)$ wird durch Differenzbildung eine Textoperation (1) erzeugt. Es soll sich um das erste Versionspaar handeln, das untersucht wird. Der Graph $G = (V, E)$ ist somit anfangs leer. Die durch Alice erzeugte Einfügeoperation (1) wird als Knoten d_1 dem Graph hinzugefügt ($V \leftarrow d \cup V$). Eine Aktualisierung stellt keine Zurücknahme fest, da die Menge der offenen Textoperationen O leer ist. d_1 wird anschließend O hinzugefügt ($O \leftarrow d_1 \cup O$), weil sie die einzige Textoperation für das Versionspaar war.

In der zweiten Runde wird die Ersetzung, repräsentiert durch (2) und (3), die durch Clair beim Übergang der Version 1001 zu 1002 generiert wurde, G durch zwei Knoten d_2 und d_3 hinzugefügt. Diese werden im Aktualisierungsschritt mit den Textoperationen in O verglichen. Dabei wird geprüft, ob sie jeweils zu einer Textoperation in O gegensätzlich sind. Dies ist nicht der Fall. Nachdem alle Textoperationen aus D untersucht wurden, werden auch d_2 und d_3 O hinzugefügt ($O \leftarrow \{d_2, d_3\} \cup O$).

In der dritten Runde werden die Differenzen des Versionspaares $(1002, 1003)$ gebildet. Bobs Löschoperation (4) wird entsprechend als Knoten d_4 G hinzugefügt und mit den offenen Einfügeoperationen aus O verglichen. Dabei wird festgestellt, dass die Textoperationen $d_4 \in V$ und $d_1 \in O$ eine Zurücknahme darstellen. Beide werden daher mittels einer Kante (d_1, d_4) verbunden. Da (1) nun durch eine Kante an einen potentiellen Konflikt gebunden ist, wird sie aus der Menge O entfernt ($O \leftarrow O \setminus d_1$), die Operation (4) von Bob dagegen hinzugefügt ($O \leftarrow d_4 \cup O$). Sie wird O hinzugefügt, da sie in späteren Versionen wieder zurückgenommen werden kann. Weiterhin gilt $N_{G^-}(d_1) = \emptyset$, so dass d_1 als Startknoten der Menge S hinzugefügt wird.

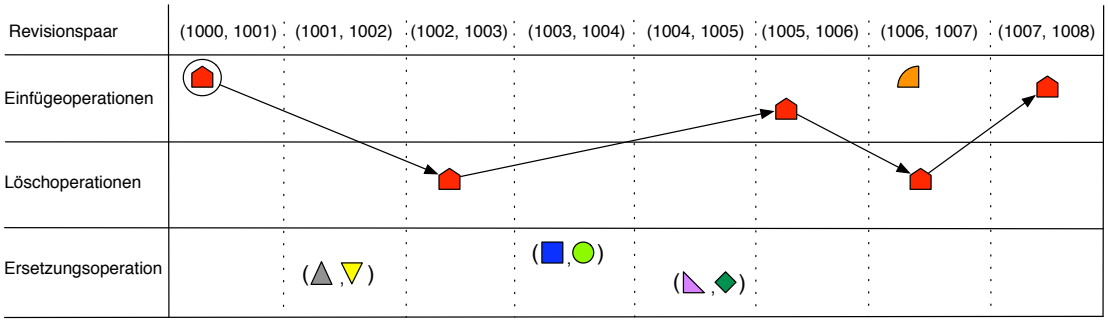

Abbildung 3.5 *: Graphische Darstellung des Bearbeitungskonflikts zwischen Alice und Bob. Jede Textoperation, die in Tabelle 3.2 aufgeführt ist, wird über ein entsprechendes Symbol referenziert. Textoperationen werden hinsichtlich ihres Operationstyps getrennt, wobei Ersetzungen als ein Paar Löschen–Einfügen dargestellt sind. Der Konflikt zwischen Alice und Bob ist durch eine Kantenfolge hervorgehoben. An ihr ist das lokale Konflikt-Muster nachzuvollziehen. Der Startknoten, von dem aus der Bearbeitungskonflikt verfolgt wird, ist umkreist.*

Im Folgenden werden die restlichen Versionspaare anhand der gleichen Vorgehensweise analysiert und jeweils die erzeugten Textoperationen dem Graph als Knoten hinzugefügt. Bei Feststellung weiterer Zurücknahmen – hier zwischen (4) und (9), (9) und (10) und (10) und (12) – wird jeweils eine Kante zwischen beiden Textoperationen generiert. In der Menge S sind alle Startknoten von Pfaden aus G enthalten. Der Startknoten $s \in S$ mit der Nachfolgermenge $N_{G^+} = 4$ beschreibt im Ergebnis einen Pfad der Länge vier. Dieser spiegelt zugleich den Bearbeitungskonflikt zwischen Alice und Bob wider.

Nach Verarbeitung aller in Tabelle 3.2 aufgeführten Versionspaare ergeben sich die in den Gleichungen 3.1 bis 3.5 aufgeführten Mengen V, E, O, S und C. Letzte umfasst die Menge aller Pfade, die mit dem lokalen Konflikt-Muster übereinstimmen. Abbildung 3.5 veranschaulicht alle im Graph aufgenommenen Textoperationen und hebt die Meinungsverschiedenheit zwischen Alice und Bob hervor.

$$V = \{d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9, d_{10}, d_{11}, d_{12}\} \tag{3.1}$$

$$E = \{(d_1, d_4), (d_4, d_9), (d_9, d_{10}), (d_{10}, d_{12})\} \tag{3.2}$$

$$O = \{d_2, d_3, d_5, d_6, d_7, d_8, d_{11}, d_{12}\} \tag{3.3}$$

$$S = \{d_1\} \tag{3.4}$$

$$C = \{\{(d_1, d_4), (d_4, d_9), (d_9, d_{10}), (d_{10}, d_{12})\}\} \tag{3.5}$$

3.2.3 Optimierung der Erkennung lokaler Zurücknahmen

Es wurde gezeigt, dass ausschließlich offene Textoperationen bei der Suche nach potentiellen lokalen Zurücknahmen in Frage kommen. Offene Textoperationen, die sich in der Menge O befinden, sollten nach einem Test auf Zurücknahme nicht verworfen werden. Sie können im weiteren Verlauf der Analyse der Versionsgeschichte durch eine gegensätzliche Textoperationen weiterhin zurückgenommen werden. Dies hat zur Folge, dass die Menge O kontinuierlich wächst. Die Anzahl der Vergleiche im Aktualisierungsschritt des Graphen nimmt daher im Verlauf eines Artikels stetig zu. Im ungünstigsten Fall existieren für die vollständige Versionsgeschichte keine gegensätzlichen Textoperationen. Im Ergebnis wird in jeder Aktualisierung des Graphen jede Textoperation in O auf Zurücknahme getestet. Dieser Umstand wirkt sich nachteilig auf die Effizienz des Verfahrens aus.

Eine Optimierung der der graphbasierten Strategie ist also vor allem durch die Minimierung der Suche nach einer potentiellen Zurücknahme erzielbar. Eine Partitionierungsstrategie leistet dies, indem sie die vollständige Menge O offener Textoperationen anhand gewählter Attribute in kleinere Partitionen zerlegt. O wird dabei anhand einer gewählten Attributmenge A partitioniert. Die Elemente der Partitionen werden in offenen Mengen

L gehalten. Eine neu hinzugefügte Textoperationen $d \in D$ wird somit ausschließlich mit den Textoperationen einer einzigen Partition verglichen. d wird folglich im ungünstigsten Fall nicht länger mit allen Elementen aus O verglichen, sondern nur mit einer Teilmenge aus O in einer Menge L_A.

Grundsätzlich werden offene Textoperationen in Hinblick auf ihren Operationstyp in getrennten Mengen $L_{A=\{\text{Operationstyp}\}}$ gehalten. Neue Löschoperationen werden somit direkt mit früheren Einfügeoperationen verglichen werden und umgekehrt. Vergleiche, die mit einer Wahrscheinlichkeit von Null zu einer Zurücknahme führen, werden vermieden. Eine günstige Attributmenge A zur Partitionierung offener Textoperationen ist:

$$A = \{\text{Operationstyp, Position im Text, Granularität der Zeichenfolge, }\}$$

Eine so gewählte Attributmenge stellt sicher, dass nur gegensätzliche Textoperationen miteinander verglichen werden, die an der gleichen Position im Text definiert sind und bezüglich ihrer Zeichenfolgen die gleiche Granularität aufweisen. Sind die Zeichenfolgen zweier Textoperationen in ihrer Granularität verschieden, ist ein Vergleich auf Gleichheit verzichtbar. Die dreielementige Attributmenge garantiert weiterhin, dass Textoperationen bei hinreichend starker Partitionierung in eine Vielzahl unterschiedlicher offener Mengen L_A eingefügt werden. Dies führt zu einer gleichmäßigeren Verteilung. Listen werden klein gehalten.

Zusätzlich werden bei einem Vergleich auf Zurücknahme zeitlich nahe liegende Textoperationen aus direkt vorangegangenen Versionen mit der neu hinzugefügten Textoperation zuerst verglichen. Diese Vorgehensweise dient nicht nur der korrekten zeitlichen Zusammenführung von Zurücknahmen für einen Bearbeitungskonflikt. Aufgrund der Tatsache, dass beinahe die Hälfte aller Bearbeitungskonflikte innerhalb von wenigen Tagen sich wieder auflösen, ist die Wahrscheinlichkeit, eine Zurücknahme zu finden, vor allem bei der Menge offener Textoperationen aus den zuletzt untersuchten Versionen am größten. Die Wahrscheinlichkeit nimmt danach stark ab. Die vorgeschlagene Minimierung der Vergleichsoperationen ist ebenso durch eine Begrenzung der Textoperationen, die in Partitionen gehalten werden, zu erreichen.

Es ist derzeit unklar, welche untere Schranke gewählt werden muss, damit keine Bearbeitungskonflikte bei der Erkennung unentdeckt bleiben. Eine offene Menge mit den zuvor genannten Eigenschaften Operationstyp, Granularität der Zeichenfolge und zeichengenauer Positionsangabe trennt hochgradig die Menge O. Eine Einschränkung auf eine zweistellige Elementanzahl sollte zu keinem Informationsverlust führen, da im gün-

stigsten Fall jede Textoperation in einer Partition eine Version der Versionsgeschichte repräsentiert.

Zusammenfassend hat die Optimierung das Ziel, Textoperationen im Graph eindeutig zu machen. Im optimalen Fall ist mit einer einzigen Vergleichsoperation eine Zurücknahme zu finden – falls eine existiert. Durch eine Zerlegung der Textoperationen wird die Zahl der Textoperationen, die auf lokale Zurücknahme im Aktualisierungsschritt des Graphen getestet werden müssen, deutlich reduziert. Nachteilig wirkt sich aus, dass durch eine Zerlegung der Menge O Zurücknahmen eventuell nicht erkannt werden. Das eine neue Differenz ausschließlich mit einer qualifizierten Menge verglichen wird, bedeutet nicht, dass die „passende“ Textoperation in Hinblick auf eine Zurücknahme auch in dieser Partition enthalten ist. Eine zu starke Trennung der Textoperationen ist daher zu vermeiden.

Die Zeichenfolge der Textoperation ist ebenfalls als Attribut zur Partitionierung heranzuziehen. Dabei ist die Zeichenfolge durch einen eindeutigen Zahlenwert zu repräsentieren. Dies kann durch den Einsatz einer Hash-Funktion erzielt werden, die für jede Zeichenfolge einen Hash-Wert berechnet. Dieser wird als Schlüssel, die Textoperation als Wert, in einer Hash-Tabelle hinterlegt. Da eine Hash-Funktion für identische Zeichenfolgen den gleichen Hash-Wert berechnet, ist die zur identischen Zeichenfolge gehörende Textoperation in konstanter Zeit durch eine einzige Schlüsselanfrage zu finden. In Verbindung mit der Versionsnummer sowie der Position im Text wird die Textoperation in der Versionsgeschichte eindeutig bestimmt. In der Konsequenz sind somit effizient lokale Zurücknahmen zu finden. Schließlich sind mit dieser Herangehensweise nur identische Zeichenfolgen zu identifizieren, die Zurücknahmen bilden.

3.2.4 Laufzeitanalyse des *Edward*-Algorithmus

Der Speicherbedarf des *Edward*-Algorithmus ist verallgemeinert für D Textoperationen, die durch eine einzige Differenzbildung generiert werden, mit $O(D(|R|-1))$ anzugeben. Alle Textoperationen, die durch die Differenzbildung von $(|R|-1)$ Versionspaaren in der Versionsgeschichte R erzeugt wurden, werden im Graph gespeichert.

Für $|R|-1$ Versionspaare ist jeweils die Differenz zu berechnen. Der zeitliche Aufwand der Differenzberechnung beträgt hierfür $O(D(m+n))$. Der Aufwand, um die vollständige Versionsgeschichte zu analysieren entspricht $O(D(m+n)(|R|-1)$. Dies wurde bereits in Kapitel 3.1 diskutiert.

Jede Textoperation $d \in D$, die erzeugt wird, ist dabei auf Zurücknahme mit der Menge O offener Textoperationen zu testen. O wird durch die Menge D repräsentiert, die

wiederum $(|R| - 1)$-mal O hinzugefügt wird. Dies führt zu $D(|R| - 1)$ Elementen in O. Im ungünstigsten Fall, bei dem in keiner Runde eine passende Zurücknahme gefunden wird und jeweils alle offenen Textoperationen in einer einzigen Partition gehalten werden, ergibt sich für die Laufzeit des *Edward*-Algorithmus eine Zeitkomplexität von $O(D^3(m+n)(|R|-1)^2)$.

Sind lokale Zurücknahmen bei der Aktualisierung des Graphen jedoch in konstanter Zeit zu finden, so verringert sich die Laufzeit zu $O(D^2(m+n)(|R|-1))$.

3.3 Metriken zur Ähnlichkeitsberechnung von Zeichenfolgen

In Kapitel 2.2.2 wurde die Annahme aufgestellt, dass konfliktbezogene Zeichenfolgen sich während eines Bearbeitungskonflikts verändern. Hierzu ein Beispiel. Alice und Bob befinden sich weiterhin im Streit um das Wort „ausschließlich“. Weil Bob ihre Bearbeitung erneut entfernt hat, fügt Alice sie wieder hinzu. Dabei verschreibt sie sich und fügt „ausschliesslich“ dem Artikel hinzu. Bob löscht die inhaltlich übereinstimmende Formulierung ebenfalls.

Eine bisherige Ausrichtung auf identische Zeichenfolgen würde die beiden betroffenen Versionen, in denen „ausschliesslich“ zwischenzeitlich Gegenstand der Meinungsverschiedenheit war, nicht erfassen. Somit ist die Identifikation lokaler Zurücknahmen auf ähnliche Zeichenfolgen zu erweitern, um leichte Veränderungen des konfliktbezogenen Textabschnitts zu berücksichtigen.

Der Vergleich von Zeichenfolgen bei der Feststellung einer lokalen Zurücknahme wird aufwendiger. Dieser konnte aufgrund der vorausgesetzten Gleichheit effizient für Zeichenfolgen beliebiger Länge mittels Berechnung eines Hash-Wertes durchgeführt werden. Dem einfachen Zahlenvergleich steht nun ein explizite Ähnlichkeitsberechnung zweier Zeichenfolgen gegenüber. Die Berechnung wird dahingehend erschwert, als dass die Zeichenfolgen, auf denen die Textoperationen definiert sind, sich in ihrer Länge, also in ihrer Granularität, stark voneinander unterscheiden. Ziel bei der Ähnlichkeitsberechnung ist es, Zeichenfolgen der gleichen Granularität mit einem geeigneten Maß miteinander zu vergleichen. Die Zeichenfolgen der Textoperationen werden hierzu, entsprechend der Ebenen lokaler Bearbeitungskonflikte, in vier Klassen eingeteilt: Zeichen, Wort, Satz und Paragraph. Gehören zwei Zeichenfolgen der gleichen Klasse an, wird eine entsprechende Ähnlichkeitsmetrik gewählt. Unterscheiden sie sich dagegen, wird die Metrik, die der gröberen Granularitätsklasse zugewiesen ist, verwendet.

Vorgestellte Ähnlichkeitsmetriken	Granularitätsklasse	Referenz
Levenshtein Distanz (Levenshtein, 1966)	Zeichen	Kapitel 3.3.1
Jaro-Winkler Distanz (Winkler, 1999)	Wort, Satz	Kapitel 3.3.1
Kosinusähnlichkeit (Salton u. Lesk, 1968)	Satz, Paragraph	Kapitel 3.3.2
SoftTF-IDF (Cohen u. a., 2003)	Wort, Satz, Paragraph	Kapitel 3.3.2
Fuzzy Fingerprinting (Stein, 2005)	Paragraph	Kapitel 3.3.3

***Tabelle 3.3** : Metriken zur Ähnlichkeitsberechnung, die zur Bestimmung der Ähnlichkeit von Zeichenfolgen bestimmter Granularitätsklassen verwendet werden.*

Durch eine entsprechende Unterteilung der Zeichenfolgen in vier Klassen ist im Ergebnis ein für die Ähnlichkeitsberechnung günstiges Maß wählbar.

Tabelle 3.3 gibt einen Überblick über die in diesem Kapitel diskutierten Metriken.

Vorbemerkungen Damit zwei Textdokumente d und d' aus der Menge der realen[3] Dokumente D auf Ähnlichkeit überprüfbar sind, wird eine logische Sicht auf diese Dokumente benötigt. Diese wird durch ein Dokumentmodell $\mathbf{d}$, $\mathbf{d}'$ realisiert.

Eine Ähnlichkeitsfunktion $\varphi(\mathbf{d},\mathbf{d}')$ berechnet für zwei Dokumentmodelle $\mathbf{d}$, $\mathbf{d}'$ aus der Menge aller Dokumentrepräsentationen $\mathbf{D}$ einen reellen Zahlenwert aus dem Intervall $[0, 1]$. Dieser gibt die Ähnlichkeit der beiden Dokumente an. Ein Wert von 0 bedeutet keine Ähnlichkeit, ein Wert nahe 1 hohe Ähnlichkeit.

Die Ähnlichkeit ist weiterhin durch eine Distanzfunktion $\sigma(\mathbf{d},\mathbf{d}')$ anzugeben. Diese bildet im Gegensatz dazu auf den Bereich der natürlichen Zahlen ab – auf das offene Intervall $[0, \infty)$. Ein Wert nahe 0 steht für hohe Ähnlichkeit. Je größer die Distanz σ, desto verschiedener sind $\mathbf{d}$ und $\mathbf{d}'$. Wird die Funktion σ ebenfalls auf das Intervall $[0, 1]$ normiert, bildet sie das Inverse zur Ähnlichkeitsfunktion φ.

3.3.1 Zeichenbasiert

Damit sehr kurze, wenige Symbole lange Zeichenfolgen zweier Textoperationen effizient vergleichbar sind, wird auf zeichenbasierte Metriken zur Ähnlichkeitsberechnung zurückgegriffen. Jedes Symbol der Zeichenfolgen wird bei der Berechnung berücksichtigt. Sie werden vorrangig bei der Verknüpfung von Datensätzen eingesetzt. Da zeichenbasierte

[3] Ist hier von Dokumenten die Rede, ist zumeist eine Computerrepräsentation eines realen Dokuments gemeint. Ein Ursprungsdokument wie beispielsweise eine Internetseite ist als „real“ anzusehen.

Ähnlichkeitsmetriken Rechtschreibfehler bis zu einem gewissen Grad tolerieren, sind Duplikate vermeidbar. Auf die gleiche Weise sind schließlich ähnliche Zeichenfolgen mittels zeichenbasierter Metriken zu bestimmen.

Als einheitliches Modell zur Repräsentation der Dokumente eignet sich intuitiv die Interpretation als eine aus Symbolen bestehende Zeichenfolge.

Die Ähnlichkeit zweier Zeichenfolgen $\mathbf{d}$ und $\mathbf{d}'$ wird bei zeichenbasierten Metriken oftmals durch eine Distanzfunktion σ bestimmt. Besitzen $\mathbf{d}$ und $\mathbf{d}'$ die gleiche Länge, so wird ihre Distanz als die Anzahl aller Positionen angegeben, an denen beide Zeichenfolgen unterschiedliche Symbole besitzen. Diese einfache Distanzfunktion ist in der Literatur als Hamming-Abstand bekannt. Sie wird in der Regel nicht verwendet, da die geforderte Einschränkung auf Zeichenfolgen gleicher Länge für den Vergleich beliebiger Textdokumente zu restriktiv ist.

Bei der Erkennung von Bearbeitungskonflikten wird daher für den Vergleich von Zeichenfolgen der Klasse „Zeichen“ die Levenshtein Distanz eingesetzt. Diese stellt eine Verallgemeinerung des Hamming-Abstands dar. Die Beschränkung auf gleichlange Zeichenfolgen ist nicht länger gegeben. Für die Klassen „Wörter“ und „Sätze“ wird jedoch die Jaro-Winkler Distanz vorgezogen. Im Gegensatz zur Levenshtein Distanz toleriert diese geringe Positionswechsel identischer Symbole in beiden Zeichenfolgen.

Levenshtein Distanz Die *Levenshtein Distanz*, auch Editierdistanz genannt, ist eine Verallgemeinerung des Hamming-Abstands. Die Distanz σ wird definiert durch die minimale Anzahl an Zeichen, die benötigt werden, um $\mathbf{d}$ nach $\mathbf{d}'$ zu überführen, so dass $\mathbf{d} = \mathbf{d}'$ gilt. Analog zur Differenzberechnung, bei der Textoperationen einen Text in einen anderen überführen, wird zwischen Einfüge-, Lösch- und Ersetzungsoperationen unterschieden. Zusätzlich werden jeder Operation Kosten ω zugewiesen. Die Distanz berechnet sich somit aus der Summe der einzelnen Kosten für jede benötigte Operation, um eine Zeichenkette in eine andere zu überführen. Durch die Einbeziehung von Einfüge- und Löschoperationen wird die ursprüngliche Beschränkung des Hamming-Abstands, wobei nur Transpositionen erlaubt sind, auf gleich lange Zeichenfolgen aufgehoben (Levenshtein, 1966).

Ein Beispiel bezugnehmend auf Alice und Bob. Die Ähnlichkeit ist für die beiden Zeichenfolgen $\mathbf{d}$= {„ausschließlich“} und $\mathbf{d}'$= {„ausschliesslich“} zu bestimmen. Wird für das Einfügen oder Löschen ein Kostenfaktor von 1 und für das Ersetzen Kosten von 2 angesetzt, so beträgt die Distanz $\sigma(\mathbf{d},\mathbf{d}')= 3$. Hierbei ist *ß* durch *s* auszutauschen und an Position 11 ein weiteres *s* hinzuzufügen.

Zwei Zeichenketten d und d' sind nach der Definition ähnlich, wenn $\sigma(\mathbf{d},\mathbf{d'}) \leq e$ gilt. Die Variable e definiert in diesem Fall eine obere Schranke, die die maximale Anzahl an Transformationen angibt, damit zwei Zeichenfolgen als ähnlich gelten.

Da jede Position beim Vergleich von zwei Zeichenketten berücksichtigt wird, eignet sich die Levenshtein Distanz gut für kurze Textlängen. Sie wird darum bei der Erkennung von Bearbeitungskonflikten für kurze Zeichenfolgen mit einer Länge kleiner 5 eingesetzt. Es gilt $\sigma(a,b) \leq e = 1$. Sie dürfen sich folglich maximal um eine Postion voneinander unterscheiden, um sich für eine Zurücknahme zu qualifizieren. In Hinblick auf das aufgeführte Beispiel ist $\sigma = 3$. Die Zeichenfolgen gelten also nicht als ähnlich. Jedoch bestehen die Wörter aus mehr als vier Buchstaben und gehören demnach zur Klasse dieser.

Löst man die Distanzberechnung durch einen dynamischen Programmieransatz, so lässt sich $\sigma(\mathbf{d},\mathbf{d'})$ mit einer Zeitkomplexität von $O(|\mathbf{d}|\cdot|\mathbf{d'}|)$ berechnen.

Jaro-Winkler Distanz Ein weit verbreitetes zeichenbasiertes Ähnlichkeitsmaß stellt die *Jaro-Winkler Distanz* dar. Es handelt sich dabei um eine durch Winkler (1999) erweiterte Metrik von Jaro (1995), der *Jaro Distanz*. Ursprünglich wurde sie zum Verknüpfen von Datensätzen anhand von Vor- und Nachnamen zur Vermeidung von Duplikaten eingesetzt. Sie eignet sich darum gut die für den Vergleich von kurzen Zeichenfolgen und Wörtern. Es zeigt sich, dass die Metrik ferner ebenso gut auf Basis von Sätzen arbeitet.

Die Ähnlichkeitsfunktion $\varphi_{Jaro}(\mathbf{d},\mathbf{d'})$ berechnet sich wie folgt:

(i) Berechnung der Längen $|\mathbf{d}|$ und $|\mathbf{d'}|$.

(ii) Gesucht werden alle „gemeinsamen“ Symbole c der beiden Zeichenfolgen $\mathbf{d}$, $\mathbf{d'}$. Als gemeinsam gelten Symbole, für die $\mathbf{d}[i] = \mathbf{d'}[j]$ mit $|i-j| \leq \frac{1}{2} min(|\mathbf{d}|,|\mathbf{d'}|)$ gilt. Buchstaben werden in der Folge weiterhin in beiden Zeichenfolgen als übereinstimmend angesehen, solange sie sich in ihrer Position nicht zu sehr unterscheiden.

(iii) Gesucht wird die Menge der Ersetzungen t, um $\mathbf{d}$ nach $\mathbf{d'}$ zu überführen. Die Menge ergibt sich, in dem die „gemeinsamen“ Symbole in $\mathbf{d}$ mit denen in $\mathbf{d'}$ verglichen werden. Ist $\mathbf{d}[i] \neq \mathbf{d'}[i]$, zählt das Symbol als Ersetzung.

(iv) Die berechneten Parameter c und t werden schließlich in Gleichung 3.6 eingesetzt, die den Ähnlichkeitswert berechnet:

$$\varphi_{Jaro}(\mathbf{d},\mathbf{d'}) \quad = \frac{1}{3} \times \left(\frac{c}{|\mathbf{d}|} + \frac{t}{|\mathbf{d'}|} + \frac{c - \frac{t}{2}}{c} \right) \tag{3.6}$$

Winkler (1999) ergänzte die Metrik durch eine höhere Gewichtung gleicher Präfixe p. Diese sind insbesondere beim Vergleich von Namen relevant. Gleichung 3.7 beschreibt

die Ähnlichkeitsfunktion.

$$\varphi_{Jaro-Winkler}(\mathbf{d}, \mathbf{d}') \quad = \varphi_{Jaro}(\mathbf{d}, \mathbf{d}') + \frac{max(|p|,4)}{10} \times (1 - \varphi_{Jaro}(\mathbf{d}, \mathbf{d}')) \tag{3.7}$$

Im Gegensatz zu der vorgestellten Levenshtein Distanz ist die Jaro-Winkler Metrik weniger restriktiv, wenn es um den Vergleich identischer Symbole an unterschiedlichen Positionen geht. Dies ist auf die Definition „gemeinsamer“ Symbole zurückzuführen.

Bei der Diskussion der Levenshtein Distanz konnte gezeigt werden, dass diese die beiden Wörter „ausschließlich“ und „ausschliesslich“ nicht als ähnlich definiert. Mittels der Jaro-Winkler Distanz wird jedoch eine Ähnlichkeit von $\varphi = 0,96$ berechnet. Konkret gilt für das Beispiel $|\mathbf{d}|= 13$, $|\mathbf{d}'|= 15$, $c = \{$„ausschlielich“$\} = 13$ und $t = 0$.

Ein weiteres Beispiel bezogen auf den Vergleich von Sätzen. Diese sind:

(i) „Die Wikipedia ist ein Projekt freiwilliger Autoren zum Aufbau einer Enzyklopädie und nichts anderes.“

(ii) „Die Wikipedia ist nichts anderes als ein Projekt freiwilliger Autoren zum Aufbau einer Enzyklopädie.“

Es ist leicht zu erkennen, dass beide Sätze inhaltlich die gleiche Aussage besitzen. Bei einem Bearbeitungskonflikt wären beide somit ein relevantes Paar für eine Zurücknahme[4] und sollten als ähnlich erkannt werden.

Durch den strikten Positionsvergleich bei Editierdistanzen wie der Levenshtein Distanz kann allerdings keine Ähnlichkeit festgestellt werden. Es sind 38 Änderungen notwendig, um den ersten in den zweiten Satz zu überführen. Die Jaro-Winkler Distanz berücksichtigt im Gegensatz dazu die Umstellung von Wörtern, so dass weiterhin eine Ähnlichkeit von $\varphi = 0,93$ erzielt wird.

Es zeigt sich, dass die vorgestellte Metrik sehr gut zum Vergleich von ähnlichen Wörtern und Sätzen dient. Hierbei werden bei der Ähnlichkeitsberechnung Umstellungen von Wörtern einbezogen. Editierdistanzen leisten dies nicht. Dies ist vor allem bei der Erkennung von Bearbeitungskonflikten von Bedeutung, da Benutzer konfliktbezogene Zeichenfolgen im Verlauf eines Konflikts umschreiben können. Diese Änderungen müssen als ähnlich erkannt werden, um den Bearbeitungskonflikt vollständig erfassen zu können.

Ein Nachteil der Jaro-Winkler Metrik stellt indes die zeichenbasierte Berechnung dar. Die Metrik ist ähnlich wie die Levenshtein Distanz mit einer Zeitkomplexität von $O(|\mathbf{d}|\cdot|\mathbf{d}'|)$

[4] Angenommen, die Positionen der dazugehörigen Textoperationen sind identisch und der Operationstyp gegensätzlich.

nicht für lange Zeichenfolgen geeignet. Weiterhin ist die Tolerierung von Wortumstellungen durch die in Schritt (ii) aufgeführte Bedingung $|i-j| \leq \frac{1}{2}min(|\mathbf{d}|,|\mathbf{d'}|)$ beschränkt. In Ergebnis werden größere Positionswechsel im Text ebenfalls nicht erkannt.

3.3.2 Wortbasiert

Damit die Reihenfolge der Wörter keinen negativen Effekt auf die Ähnlichkeit ausübt, werden bei wortbasierten Ähnlichkeitsmaßen die einzelnen Wörter einer Zeichenfolge separat betrachtet.

Intuitiv berechnet sich die Ähnlichkeit zweier Zeichenfolgen durch einen Vergleich aller gemeinsamen Wörter mit der Anzahl aller Wörter aus beiden Zeichenfolgen.

Dies führt zu einer Definition eines Dokumentmodells für wortbasierte Verfahren. Ein Textdokument $d \in D = \{d_1, \ldots, d_n\}$ wird durch die endliche Menge an Worten, auch Terme genannt, $W = \{w_o, \ldots, w_m\}$ aller in D enthaltenen Dokumente repräsentiert. W bildet das Vokabular aller Dokumente. Für jedes Dokument $d_i \in D$ wird dem Wort w_j ein Gewicht $w_{i,j} \in \{0, 1\}$ zugewiesen. Intuitiv wird jedem Wort w_j ein Gewicht von 1 zugesprochen, wenn es im Dokument enthalten ist, ansonsten 0. Besitzt ein Wort in beiden Dokumentrepräsentationen ein Gewicht von 1, haben beide Textdokumente das Wort gemeinsam. Folglich beschreibt ein Vektor $\mathbf{d}_i = \{w_{i,1}, \ldots, w_{i,m}\}$ ein Textdokument. Das Modell wird dementsprechend nach Salton u. Lesk (1968) Vektorraummodell genannt. Textdokumente werden in einzelne Wörter unterteilt und durch Vektoren beschrieben.

Die obige Definition eines Dokumentmodells lässt alle im Dokument enthaltenen Wörter gleichwertig den Inhalt eines Dokuments beschreiben. Jedoch repräsentieren Wörter, die in vielen Dokumenten der Menge D vorkommen, ein Dokument inhaltlich schlechter als Wörter, die in sehr wenigen oder ausschließlich in einem Dokument auftreten. Zur Gewichtsanpassung wird häufig das sogenannte TF-IDF Schema verwendet. Die inverse Dokumenthäufigkeit (IDF) gibt dabei die Anzahl der Dokumente an, in denen ein bestimmter Term vorkommt. Je geringer der Wert ist, desto besser beschreibt das Wort ein Dokument. Die Termfrequenz (TF) stellt bezogen auf ein individuelles Dokument die absolute Häufigkeit eines Wortes dar und gilt als Indikator für die Wichtigkeit eines Wortes hinsichtlich der inhaltlichen Beschreibung eines Dokuments. Beide Metriken werden häufig kombiniert, um die Gewichte im Vektor $\mathbf{d}$ nach der Formel in Gleichung 3.8 zu bestimmen (Ferber, 2003).

$$w_{i,j} = TF \cdot IDF = h(i,j) \cdot \frac{1}{d(j)} = \frac{h(i,j)}{d(j)} \tag{3.8}$$

$h(i,j)$ bezeichnet dabei die Häufigkeit des Auftretens eines Terms w_j im Dokument d_i, $d(j)$ die Anzahl der Dokumente, die den Term w_j enthalten. Ein Gewicht $w_{i,j}$ erhält demnach einen positiven Wert aus $\mathbb{R}$.

Kosinusähnlichkeit Aufgrund der Tatsache, dass Textdokumente durch Vektoren dargestellt werden, ist die Ähnlichkeit $\varphi(\mathbf{d},\mathbf{d}')$ zweier Gewichtsvektoren durch den Kosinus berechenbar. Je geringer der Winkel ausfällt, desto ähnlicher sind sich die beiden Dokumente. Die Ähnlichkeitsfunktion φ wird entsprechend in Gleichung 3.9 formuliert.

$$\varphi(\mathbf{d},\mathbf{d}') = \frac{\langle \mathbf{d},\mathbf{d}' \rangle}{|\mathbf{d}| \times |\mathbf{d}'|} \tag{3.9}$$

Dieses Ähnlichkeitsmaß eignet sich für den Vergleich von Zeichenfolgen, die sich in der Granularitätsklasse „Satz“ oder „Paragraph“ befinden. Die Position der Wörter im Text ist für die Berechnung der Ähnlichkeit nicht von Bedeutung. Desweiteren ist die Bestimmung der Ähnlichkeit zweier Zeichenfolgen gegenüber zeichenbasierten Metriken effizienter. Dies folgt durch die Repräsentation von Texten durch Gewichtsvektoren. Die Berechnung des Kosinus genügt, um die Ähnlichkeit anzugeben. Ein Vergleich jedes einzelnen Zeichens entfällt.

Die Effizienz des Verfahrens nimmt mit steigender Dimensionalität der Gewichtsvektoren durch Verarbeitung langer Dokumente mit großem Vokabular ab.

Ein Nachteil der Kosinusähnlichkeit ergibt sich durch das Vektorraummodell. Dokumente mit ähnlichem Inhalt, aber anderem Vokabular bzw. häufigen Rechtschreibfehlern, werden mit hoher Wahrscheinlichkeit nicht als ähnlich angesehen. Um diesem entgegenzuwirken, können die Wörter, die das Vokabular bilden, auf ihre Stamm- respektive Grundformen reduziert und sogenannte Stoppworte aus dem Vokabular entfernt werden. Stoppworte sind beispielsweise Artikel oder Pronomen. Diese kommen häufig in einer Sprache vor und sind nicht für den Inhalt relevant. Ein Entfernen von Stoppworten aus dem Vokabular empfiehlt sich dagegen bei der automatischen Erkennung von Bearbeitungskonflikten nicht, da bei den kurzen Zeichenfolgen zu viele Informationen über den Inhalt verloren gehen.

SoftTF-IDF realisiert mit Jaro-Winkler Cohen u. a. (2003) stellen in ihrer Arbeit ein Verfahren namens *SoftTF-IDF* vor. Es basiert ebenfalls auf der Kosinusähnlichkeit sowie und beruht auf einer Gewichtung nach dem TF-IDF Schema. Wortähnlichkeiten werden jedoch zusätzlich durch eine zweite Metrik, der Jaro-Winkler Distanz, bestimmt. So werden auch Wörter, die ähnlich zueinander sind wie beispielsweise „ausschließlich“ und das falschgeschriebene „ausschliesslich“ als gemeinsames Wort zwischen Dokumenten erkannt.

3.3.3 Hashing-basiert

Jede der vorgestellten Ähnlichkeitsmetriken ist ineffizient bei einem Vergleich langer Textdokumente. Eine Hash-Funktion h, die eine beliebig lange Zeichenfolge auf eine natürliche Zahl abbildet, ist hier vorzuziehen. Der resultierende Hash-Wert wird auch als Fingerabdruck bezeichnet. Dieser beschreibt die Zeichenfolge eindeutig[5]. Zwei Texte $\mathbf{d}$ und $\mathbf{d}'$ sind identisch, wenn sie den gleichen Hash-Wert besitzen. Gleichung 3.10 stellt diesen Zusammenhang nochmals dar.

$$\mathbf{d} = \mathbf{d}' \Rightarrow h(\mathbf{d}) = h(\mathbf{d}') \tag{3.10}$$

Eine Ähnlichkeitsfunktion $\varphi(\mathbf{d}, \mathbf{d}')$ auf Basis von Hash-Funktionen bildet auf die Menge $\{0, 1\}$ ab. Sind beide Zeichenfolgen identisch, wird der Wert 1 angenommen. Anders ausgedrückt: Beim Auftreten einer Hash-Kollision sind zwei Texte identisch.

Hash-Werte sind zur Ähnlichkeitsbestimmung bei langen Texten und einer großen Dokumentensammlung vorzuziehen, da sie eindeutig ein Dokument beschreiben. Duplikate sind in konstanter Zeit durch einen einmaligen Zahlenvergleich effizient zu finden. Bekannte *Hashing*-Verfahren sind jedoch nicht für die Suche nach ähnlichen Textdokumenten geeignet. Bereits die Änderung eines Zeichens führt dazu, dass ein anderer Hash-Wert für ein Dokument generiert wird.

Fuzzy Fingerprinting Stein (2005) entwickelte eine Hash-Funktion, die beim Auftreten einer Hash-Kollision annimmt, dass zwei Dokumente hohe Ähnlichkeit aufweisen. Das Konzept kann gemäß Stein zur Suche nach ähnlichen Dokumenten verwendet werden. Somit sind auch Hash-Funktionen für die Bestimmung von ähnlichen Zeichenfolgen verwendbar.

Sei $\mathbf{d}$ eine Dokumentrepräsentation von d und d' ein zu d ähnliches Dokument. Eine Ähnlichkeitsfunktion φ bildet die Ähnlichkeit auf das Intervall $[0, 1]$[6] ab. So genannte „fuzzyfizierte“ Hash-Funktionen h_φ folgen der Bedingung in Gleichung 3.11:

$$h_\varphi(\mathbf{d}) = h_\varphi(\mathbf{d}') \Rightarrow \varphi(\mathbf{d}, \mathbf{d}') \geq 1 - \epsilon,\ mit\ 0 < \epsilon \ll 1 \tag{3.11}$$

Eine auf diese Art und Weise bestimmte Ähnlichkeit erlaubt es nicht, die Dokumente nach der Reihenfolge zu ordnen, da eine Hash-Funktion binär entscheidet. Das Dokument ist entweder ähnlich oder nicht ähnlich. Es existiert kein Grad der Ähnlichkeit. Dieser ist

[5] Angenommen, es wird eine ideale Hash-Funktion verwendet, die kollisionsresistent ist. Für zwei Zeichenfolgen wird nie der gleiche Hash-Wert berechnet.

[6] Maximale Ähnlichkeit entspricht 1, keine Ähnlichkeit entspricht 0.

bei der automatischen Erkennung von Bearbeitungskonflikten nicht notwendig, da durch zusätzliche lokale Informationen im Artikel die Zeichenfolge eindeutig bestimmt werden kann.

Es findet eine deutlich stärkere Abstraktion der Dokumente statt, so dass ein Dokument weniger präzise durch hashing-basierte Ansätze repräsentiert wird, als es durch das Vektorraummodell der Fall wäre. Detaillierte Informationen zum *Fuzzy Fingerprinting* (Realisierung und Aufbau) bietet Stein (2005) und die Zusammenhänge zur Ähnlichkeitssuche verdeutlichen Stein u. Potthast (2006).

Fuzzy Fingerprinting wird vor allem verwendet, um längere Zeichenfolgen, die der Klasse „Paragraph“ angehören, untereinander zu vergleichen. Hoppe (2006) zeigt in diesem Zusammenhang, dass bei entsprechender Parametrisierung der „unscharfen“ Hash-Funktion Textausschnitte aus Wikipedia-Artikeln mit einer Länge von 40 Wörtern in 97% der Fälle einen eindeutigen Hash-Wert besitzen. Dieser Wert wurde für die Bildung der Paragraphklasse, die Zeichenfolgen mit 40 oder mehr Wörtern enthält, übernommen. Desweiteren wird belegt, dass für einen Anfrageabschnitt 80% der relevanten Textabschnitte in Wikipedia gefunden werden, die zum angefragten Textabschnitt eine Ähnlichkeit $\varphi \geq 0.85$ aufweisen. Zeichenfolgen, die eine geringe Ähnlichkeit von $\varphi \leq 0.7$ aufweisen, werden dagegen deutlich schlechter ermittelt. Im Ergebnis ist bei der automatischen Erkennung von Bearbeitungskonflikten keine weitere Ähnlichkeitsberechnung anzuschließen, wenn mittels Fuzzy Fingerprinting zwei ähnliche Zeichenfolgen im Mustererkennungsschritt ermittelt werden.

3.4 Verwandte Arbeiten

Viégas u. a. (2004) entwickelten eine Anwendung namens *History Flow* zur visuellen Analyse von Versionsgeschichten aus Wikipedia. Die Entwicklung eines Artikels im Laufe der Zeit ist zu verfolgen und Unterschiede zwischen einzelnen Versionen sind visuell zu erfassen. In der Arbeit wurden spezielle Muster in Artikeln entdeckt, so dass eine erste Visualisierung eines Bearbeitungskonfliktes in Form eines Zick-Zack Musters, siehe Abbildung 3.6, erfolgt. Dieses Muster repräsentiert allerdings nur eine ganz bestimmte Art eines Konflikts: Einfügen und Löschen von längeren Textabschnitten.

Buriol u. a. (2006) beschäftigten sich mit der Repräsentation von Verknüpfungen (Hyperlinks) der Wikipedia-Artikel untereinander. In der Arbeit werden Zurücknahmen von Versionen automatisch ermittelt, in dem die Benutzerkommentare hinsichtlich des Wortes „revert“ untersucht werden. Dieses wird bei der Verwendung der Zurücknahme-Funktion

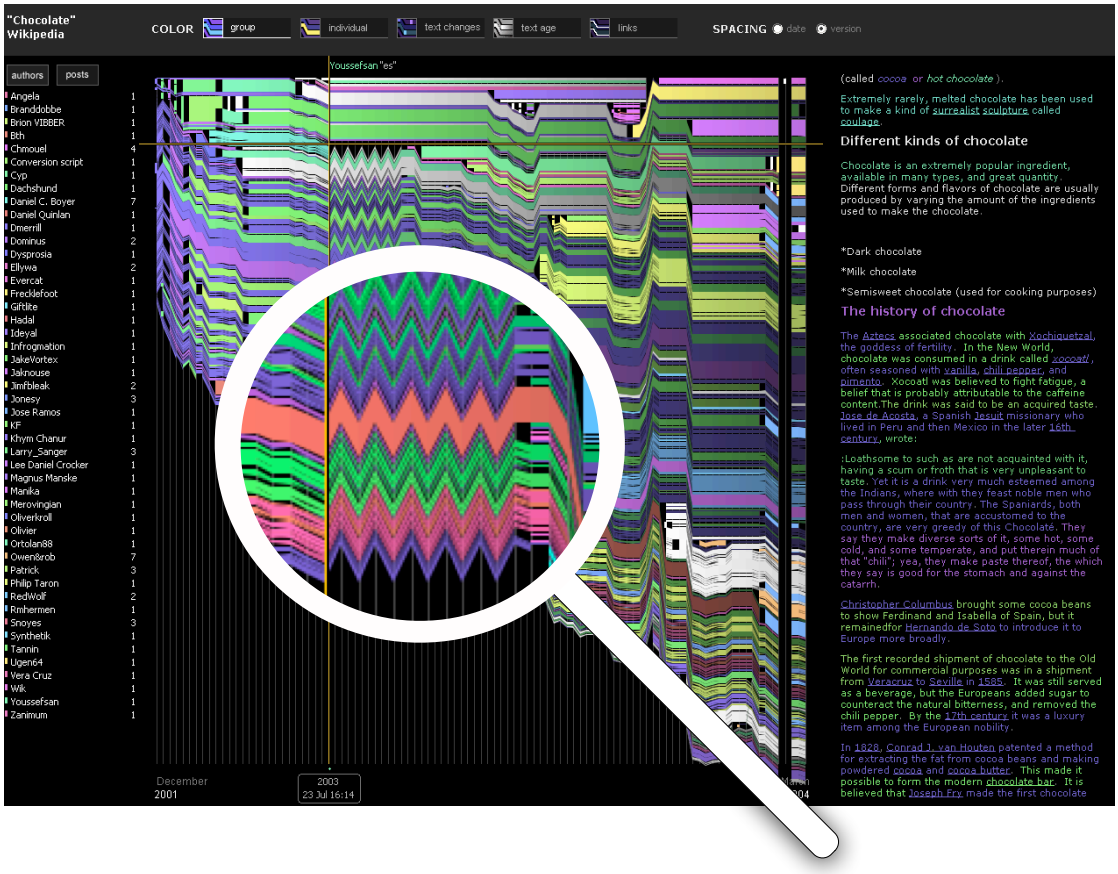

Abbildung 3.6 *: History Flow Visulization Tool. Das Zick-Zack Muster ist vergrößert hervorgehoben und repräsentiert einen Bearbeitungskonflikt vom Typ* Einfügen–Löschen *eines längeren Textabschnitts.*

von Wikipedia automatisch erzeugt. Bearbeitungskonflikte werden dabei in Bezug zu Zurücknahmen gebracht.

Die Autoren zeigen im Rahmen einer Analyse der Wikipedia, dass Zurücknahmen in den letzten Jahren im Vergleich zu „normalen“ Bearbeitungen am Artikel zugenommen haben. Sie sind in erster Linie Anzeichen für wachsenden Vandalismus an Artikeln, der mittels Zurücknahmen „repariert“wird. Ebenso bilden wiederholte Zurücknahmen an einem Artikel die Grundlage für Bearbeitungskonflikte und stellen ebenfalls zunehmend ein Problem in Wikipedia dar.

Konkret besaßen im Januar 2006 Zurücknahmen von Versionen einen Anteil von schätzungsweise 6% an allen Änderungen der englischen Wikipedia. 70% dieser wurden innerhalb einer Stunde durchgeführt.

Kittur u. a. (2007) analysierten Konflikte in Wikipedia. In diesem Zusammenhang wird eine Visualisierung von Zurücknahmen vorgestellt, bei der Benutzer in Beziehung gesetzt werden, die an einem Streit beteiligt sind. Die Kontroverse wird erneut anhand einer Kommentarauswertung nach Buriol u. a. (2006) erkannt. Zusätzlich wird eine Hash-Funktion verwendet, um Duplikate in der Versionsgeschichte zu ermitteln. Die Autoren unterscheiden explizit *identische* Zurücknahmen von *partiellen*, bei denen nur Teile früherer Änderungen zurückgenommen werden. Letztere werden auf Basis von Benutzerkommentaren ermittelt, nehmen nur 6% aller Zurücknahmen ein und seien damit vernachlässigbar. Die Autoren verwenden in beiden Fällen „globale“ Maße, die Kommentarauswertung sowie einen Hash-Wert für die gesamte Version, und analysieren nicht den Inhalt des Artikels. Es werden keine lokalen Bearbeitungskonflikte in der Weise erfasst, wie sie in der hier vorliegenden Ausarbeitung definiert sind.

Die Visualisierung stellt einen *Zurücknahmegraphen* dar, bei dem jeweils Benutzer (Knoten) durch Kanten (Zurücknahme) verbunden sind, die gegenseitig die Änderungen des Anderen zurückgenommen haben. Durch die Bildung von Gruppen ist es möglich, bestimmte Benutzergruppen zu bestimmen, die am Konflikt beteiligt sind.

Neben dem Zurücknahmegraphen wurde darüber hinaus ein lernbasiertes Verfahren vorgestellt, um das Konfliktpotential eines Artikels zu messen. Dies ist in Bezug auf diese Ausarbeitung interessant, denn so sind Artikelkandidaten, in denen Bearbeitungskonflikte mit hoher Wahrscheinlichkeit auftreten, mittels einer Klassifikation einzugrenzen.

4 Evaluierung des *Edward*-Algorithmus

Im Folgenden wird der in Kapitel 3 vorgestellte *Edward*-Algorithmus evaluiert. Dabei ergeben sich zwei Kategorien von Fragestellungen, wobei erstgenannte anhand von Experimenten auf dem Referenzkorpus sowie letztere anhand von einem Auszug der englischen Wikipedia beantwortet werden. In Bezug auf das Verfahren sind folgende Fragen von besonderem Interesse, die in Kapitel 4.3 diskutiert werden:

- Ist der *Edward*-Algorithmus in der Lage, Bearbeitungskonflikte zu entdecken?
- Wenn ja, wie fällt die Wahl der Parameter aus, um optimale Retrieval-Ergebnisse zu erzielen?
- Wie hoch ist der Anteil falsch entdeckter Bearbeitungskonflikte?
- Sind Meinungsverschiedenheiten, die ausschließlich ein Satzzeichen oder ein einzelnes, häufig im Text vorkommendes Wort betreffen, eindeutig erfassbar?
- Welche Attributmenge zur Partitionierung offener Textoperationen erzielt günstige Ergebnisse bei gleichzeitig geforderter starker Trennung dieser?

Desweiteren diskutiert eine vollständige Untersuchung der englischen Wikipedia in Kapitel 4.4 folgende Fragen allgemeinen Interesses:

- Wie viele Bearbeitungskonflikte existieren in Wikipedia?
- Sind eher viele oder nur wenige Benutzer in einen Streitfall involviert? Wie lange dauert dieser im Durchschnitt?
- Deckt ein Ansatz, der auf ähnlichen, lokalen Zurücknahmen beruht, tatsächlich eine größere Anzahl an Bearbeitungskonflikten auf als bisherige, globale Ansätze?
- Wie verteilen sich die Konflikte auf die in der Taxonomie vorgestellten Typen?
- Welche Art von Wikipedia-Artikeln ist vor allem Kontroversen ausgesetzt? Wie groß ist der Anteil betroffener Artikel an der Gesamtheit?

Zunächst wird der Referenzkorpus, in dem Bearbeitungskonflikte dokumentiert sind, vorgestellt. Es folgt eine Darstellung des Experimentverlaufs, an die eine Analyse des *Edward*-Algorithmus vor dem Hintergrund der oben genannten Fragestellungen anknüpft.

4.1 Referenzkorpus zur Durchführung der Experimente

Zur Evaluierung des *Edward*-Algorithmus wird eine maschinenlesbare Datensammlung von Bearbeitungskonflikten benötigt. Der englische Wikipedia-Artikel *Lamest Edit-Wars* beinhaltet dazu eine Kollektion zusammengetragener Bearbeitungskonflikte (Wikipedia, 2007n). Ein Konflikt wird ausschließlich in wenigen Sätzen inhaltlich beschrieben. Beteiligte Versionen, involvierte Benutzer und betroffene Textstellen werden nicht genannt. Gegenwärtig existiert weder eine Sammlung von Wikipedia-Artikeln, in denen diese explizit diskutiert sind, noch sind detaillierte Aufzeichnungen über Meinungsverschiedenheiten zwischen Benutzern vorhanden. Es ist daher notwendig, einen eigenen Referenzkorpus per Hand zu konstruieren, der den Ansprüchen einer automatischen Evaluierung genügt.

Da keine Bearbeitungskonflikte detailliert in Wikipedia schriftlich festgehalten sind, wurden bei der Suche nach Bearbeitungskonflikten insbesondere Wikipedia-Artikel untersucht, die sich mit dem Thema Benutzerkonflikte und deren Schlichtung beschäftigen. Tabelle 4.1 stellt diese zusammen.

Quelle	Konfliktfälle	Referenz
Verstöße gegen die Neutralität	5000	Wikipedia (2007f,c)
Mediation Cabal	800	Wikipedia (2007r)
Vermittlungsausschuss	750	Wikipedia (2007t,i)
Liste kontroverser Themen	670	Wikipedia (2007o)
Schiedsgericht	350	Wikipedia (2007s,u)
Lamest Edit Wars	300	Wikipedia (2007n)
Zurücksetzstatisitk	200	Karwath (2007)
Wikirage	100	Wood (2007)

***Tabelle 4.1** : Zusammenstellung von Seiten, die sich mit Benutzerkonflikten in Wikipedia beschäftigen. Bei den Konfliktfällen, die für jede Seite angegeben sind, handelt es sich um Schätzwerte. Es liegen jeweils keine genauen Angaben vor.*

Jeder auf diese Weise offengelegte Bearbeitungskonflikt wird durch eine XML-basierte Dokumentation beschrieben. Hierfür wurde eine Notation entworfen, die bei der manuellen sowie automatischen Aufzeichnung von Meinungsverschiedenheiten Anwendung findet. Im Anhang C ist das korrespondierende XML-Schema angefügt. Es definiert die logische Struktur der Dokumente. Für den Bearbeitungskonflikt zwischen Alice und Bob zeigt Abbildung 4.1 analog die XML-basierte Dokumentation. Relevante Informationen bei der Beschreibung eines Bearbeitungskonflikts fasst Tabelle 4.2 zusammen.

Der *Edward*-Algorithmus erwartet als Eingabe in einem *offline* Szenario die vollständige Versionsgeschichte eines Artikels. Diese ist in den von Wikipedia bereitgestellten Auszügen[1] enthalten. Der hier verwendete Auszug zur Gewinnung dieser trägt die Bezeichnung *enwiki-20060816-pages-meta-history* und liegt im XML-Format vor.

Der in dieser Ausarbeitung erstellte Referenzkorpus names *Wikipedia Edit-War Corpus WEBIS-EWC08-01* beinhaltet 24 Artikel. Für diese sind insgesamt 51 Bearbeitungskonflikte begleitend im XML-Format dokumentiert.

Der Referenzkorpus wurde anhand der Kriterien der Korpuslinguistik erstellt, um wissenschaftlichen Ansprüchen zu genügen. Der Begriff Korpuslinguistik umfasst nach Mehler (2005) die Gesamtheit aller Tätigkeiten, die darauf gerichtet sind, *authentisches* Textmaterial zu aggregieren, zusammen zu stellen und aufzubereiten. Eine Aufbereitung meint das Hinzufügen zusätzlicher Informationen beispielsweise durch Annotationen. Das Textmaterial soll öffentlich verfügbar gemacht werden, um den Korpus für wissenschaftliche Zwecke auswerten zu können.

Merkmal	Beschreibung
pageid	Artikel besitzen eine eindeutige Identifikationsnummer.
revid	Versionen verfügen ebenfalls über eine Identifikationsnummer.
position	Position der Zeichenfolge im Text.
token	Zeichenfolge, die vom Streit betroffen ist.
action	Operationstyp der an die Zeichenfolge gebundenen Textoperation.

***Tabelle 4.2** : Informationen, um Bearbeitungskonflikte XML-basiert zu dokumentieren. Diese sind den beteiligten Versionen eines Bearbeitungskonflikts zu entnehmen.*

[1] Der jeweils aktuellste Auszug der Versionsgeschichte ist unter `feed://download.wikimedia.org/enwiki/latest/enwiki-latest-pages-meta-current.xml.bz2-rss.xml` zu beziehen.

```
<?xml version="1.0" encoding="UTF-8"?>
<edit-war pageid="195216"
  type="insert-remove"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://www.uni-weimar.de/m/webis"
  xsi:schemaLocation="http://www.uni-weimar.de/m/webis
    http://www.uni-weimar.de/~hoppe/edward/edwardoc-schema.xsd">
  <description>
    Zwei Benutzer, Alice und Bob, streiten sich um ein wichtiges Wort.
  </description>
  <edits>
    <edit id="token_1">ausschließlich</edit>
  </edits>
  <revisions>
    <revision revid="1007" position="18"
      action="removed"  edit-id="token_1" />
    <revision revid="1014" position="18"
      action="inserted" edit-id="token_1" />
    <revision revid="1017" position="18"
      action="removed"  edit-id="token_1" />
    <revision revid="1208" position="18"
      action="inserted" edit-id="token_1" />
  </revisions>
  <discussion revid="7223771" />
</edit-war>
```

Abbildung 4.1 *: Beispiel einer XML-Dokumentation für einen Bearbeitungskonflikt.*

Es werden im Folgenden Kriterien besprochen, die bei der Zusammenstellung des Referenzkorpus berücksichtigt wurden.

Authentizität Artikel, die dem Wikipedia-Auszug entnommen wurden, sind unverändert. Eine Aufbereitung durch Annotationen für konfliktbezogene Textabschnitte in den Versionen eines Artikels wäre denkbar gewesen, doch ist sie aus mehreren Gründen nicht erwünscht. Zum einen stellt dies eine direkte, ungewollte Hilfestellung bei der Erkennung eines Bearbeitungskonfliktes durch ein Programm dar, an-

dererseits geht die Generalität des Korpus verloren. Durch die Authentizität bleibt gewährleistet, dass ein Erkennungsverfahren gleichermaßen in einem *offline* sowie *online* Szenario verwendet werden kann.

Erweiterbarkeit Der Referenzkorpus ist zukünftig durch weitere Bearbeitungskonflikte für die bereits erfassten Artikel erweiterbar. Dies wird durch die Trennung von Versionsgeschichte und begleitender Dokumentation erreicht.

Einfachheit und Maschinenlesbarkeit Dokumentationen von Bearbeitungskonflikten sind im XML-Format angelegt. Die Struktur wird mittels eines XML-Schemas festgelegt. Auf zusätzliche Metainformationen wie am Konflikt beteiligte Benutzer, Zeitstempel oder Kommentare der Teilnehmer zu den Bearbeitungen wurde verzichtet. Auf Wunsch sind diese automatisch zu generieren oder als *erweiterte* Dokumentation anzugeben.

Repräsentativität Es wurden bei der Zusammenstellung des Korpus alle bekannten Typen von Bearbeitungskonflikten berücksichtigt. Dabei wurde von der vorgestellten Taxonomie aus Kapitel 2.2.2 ausgegangen. Allerdings konnte nicht sichergestellt werden, ob die Verteilung der Bearbeitungskonflikte auf die einzelnen Klassen repräsentativ wiedergegeben wird.

4.2 Experimente

Experimente, die zur Beantwortung der Fragen dienen, die sich direkt auf den *Edward*-Algorithmus beziehen, basieren auf dem erstellten Referenzkorpus. Jeder enthaltene Wikipedia-Artikel im Korpus wird auf Bearbeitungskonflikte hin untersucht. Nachdem alle Versionspaare eines Artikels analysiert wurden, werden die automatisch entdeckten Bearbeitungskonflikte einzeln mit dem manuell dokumentierten Gegenstück, falls vorhanden, verglichen. Die per Hand erstellte Dokumentation beschreibt die Bearbeitungskonflikte im Referenzkorpus vollständig und dient darum als Referenz, um die Genauigkeit und Vollständigkeit der automatischen Erfassung zu beurteilen. Kapitel 4.2.1 stellt hierfür verwendete Gütemaße vor.

Gegenwärtig sind nur beschränkt eindeutige Aussagen über die Genauigkeit und Vollständigkeit automatisch erfasster Bearbeitungskonflikte zu treffen. Hierfür ist die vollständige Versionsgeschichte eines Artikels zu untersuchen, um richtige von falsch erkannten konfliktbezogenen Zeichenfolgen zu unterscheiden. Im Referenzkorpus befindet sich

jedoch kein Wikipedia-Artikel, für den dieses realisiert ist, so dass zusätzlich die Auswertung manuell auf Basis von Stichproben durchgeführt wird.

Vor einem Durchlauf des Experiments werden bestimmte Parameter festgelegt. Diese dienen der Einstellung der Retrieval-Qualität des *Edward*-Algorithmus. Kapitel 4.2.2 stellt diese vor und motiviert, durch welche Experimente ihr Einfluss auf die Retrieval-Qualität zu bewerten ist.

4.2.1 Gütemaße zur Bewertung des Verfahrens

Um die Retrieval-Eigenschaften einheitlich zu messen und verschiedene Systeme vergleichbar zu machen, werden die im Information Retrieval bekannten Gütemaße *Precision* und *Recall* bei der Evaluierung des *Edward*-Algorithmus herangezogen. Beide Gütemaße sind schematisch in Abbildung 4.2 dargestellt.

Precision ist ein Maß, um die Genauigkeit einer Antwort auf eine Anfrage zu messen. Eine Anfrage stellt hier die Suche nach einem konkreten Bearbeitungskonflikt in einem Wikipedia-Artikel dar. Die Antwort bildet die Versionsmenge des automatisch erfassten Bearbeitungskonflikts. In der Antwortmenge wird zur Bestimmung der Precision der Anteil *relevanter* Versionen bestimmt. Eine Version gilt als relevant, falls ein automatisch erkanntes Textoperationstupel mit dem Tupel der manuell dokumentierten Version übereinstimmt. Die Precision definiert sich nach Baeza-Yates u. Ribeiro-Neto (1999) wie folgt in Gleichung 4.1 und ist in Hinblick auf Bearbeitungskonflikte angepasst.

$$Precision = \frac{\text{Anzahl gefundener und relevanter Versionen}}{\text{Anzahl gefundener Versionen}} = \frac{|R \cap A|}{|R|} \tag{4.1}$$

Recall ist ein Maß, um die Vollständigkeit einer Antwort auf eine Anfrage zu messen. Vor diesem Hintergrund wird der Anteil relevanter Versionen in der Antwortmenge ins Verhältnis zu allen relevanten Versionen gesetzt. Die Menge aller relevanten Versionen wird durch die Versionen, die der dokumentierte Bearbeitungskonflikt einschließt, gebildet. Der Recall definiert sich nach Gleichung 4.2 in ähnlicher Weise wie die Precision.

$$Recall = \frac{\text{Anzahl gefundener und relevanter Versionen}}{\text{Anzahl relevanter Versionen}} = \frac{|R \cap A|}{|A|} \tag{4.2}$$

Zur Bewertung der Retrieval-Qualität des *Edward*-Algorithmus werden gemittelte Werte von Precision und Recall herangezogen. Sie werden jeweils getrennt nach Typ und Ebene von Bearbeitungskonflikten berechnet. Auf diese Weise sind für jede Art von Bearbeitungskonflikt individuelle Aussagen über die Retrieval-Qualität zu treffen.

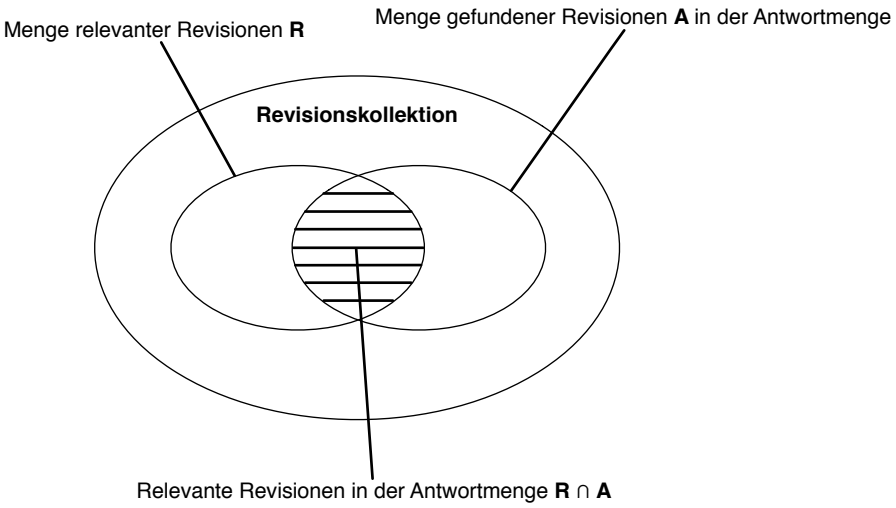

***Abbildung 4.2** : Schematische Darstellung der Gütemaße* Precision *und* Recall.

Precision und Recall werden ebenfalls verwendet, um die Retrieval-Qualität in Bezug auf die allgemeine Erkennungsleistung der dokumentierten Bearbeitungskonflikte anzugeben. Die Berechnung ist in diesem Fall davon abhängig, ob ein Bearbeitungskonflikt überhaupt gefunden wird oder nicht.

4.2.2 Allgemeine Parameter

Im Folgenden werden Parameter des *Edward*-Algorithmus beschrieben, die vor jedem Experiment festgelegt und für die Dauer der Analyse nicht verändert werden. Für jeden Parameter wird kurz motiviert, durch welche Experimente sein Einfluss auf die Retrieval-Qualität des *Edward*-Algorithmus zu messen ist.

Wahl geeigneter Metriken bzgl. der adaptiven Ähnlichkeitsstrategie Damit lokale Zurücknahmen nicht ausschließlich durch identische Zeichenfolgen beschränkt sind, werden zusätzlich ähnliche Zeichenfolgen berücksichtigt. Dies erfordert für jede Granularitätsstufe der Zeichenfolgen eine geeignete Metrik.

Um eine Aussage über die Wichtigkeit der Ähnlichkeitsmetriken auf die Retrieval-Qualität des *Edward*-Algorithmus vornehmen zu können, wurden die in Kapitel 3.3 vorgestellten Metriken eingesetzt. Hierbei ist vor allem der Vergleich der Ähnlichkeitsmetriken mit einer hash-basierten Erkennung interessant. Letztgenannte erfassen ausschließlich identische Zeichenfolgen. Der Vergleich erlaubt eine Aussage darüber, ob die in dieser Ausarbeitung aufgestellte Annahme, dass Benutzer wäh-

rend des Bearbeitungskonflikts auch konfliktbezogene Zeichenfolgen umformulieren, zutrifft.

Ähnlichkeitsschwellwert zur Identifikation lokaler Zurücknahmen Bezugnehmend zur adaptiven Ähnlichkeitsberechnung wird für jede Metrik ein eigener Schwellwert benötigt. Liegt ein berechneter Wert über diesem, gelten zwei Zeichenfolgen als ähnlich.

Eine Untersuchung der Ähnlichkeit von Zeichenfolgen unterstützt die Bewertung der Ähnlichkeitsmetriken, indem verschiedene Schwellwerte aus dem Intervall $[0, 1]$ bezüglich ihres Einflusses auf die Retrieval-Qualität analysiert werden.

Positionsauflösung der Textoperationen Eine Textoperation setzt sich aus einem Tupel, bestehend aus Zeichenfolge, Position im Text und Operationstyp, zusammen. In Kapitel 3.1 wurde erwähnt, dass insbesondere die Positionsangabe in ihrer Auflösung variierbar ist. Sie dient in erster Linie als lokale Information im Text. Die Auflösung ist zeichenbasiert, wortbasiert, satzbasiert oder paragraphbasiert einstellbar. Der Verzicht auf eine Positionsangabe ist ebenso möglich wie eine Abbildung der Position auf ein Vielfaches einer vorher zu definierenden Konstante.

Um den Einfluss der Positionsauflösung auf die Retrieval-Qualität angeben zu können, werden verschiedene Positionsangaben bei fest gewählten Ähnlichkeitsmetriken und Schwellwerten untersucht. Im Rahmen der Experimente werden u.a. zeichengenaue und satzgenaue Positionsangaben verglichen. Ferner wird auf die Positionsinformationen bei der automatischen Erkennung von Bearbeitungskonflikten gänzlich verzichtet, um ihre Relevanz zu bewerten.

Zur Bestimmung der Position für eine Textoperation wird folgende Gleichung verwendet:

$$\text{Position} = \lfloor \frac{1}{k} \cdot \text{Satzindex im Text} + 0,5 \rfloor \cdot k, \quad k \in \{0, 1, 3, 5, 10, 15, 20, 50, 75\}$$

Granularität der Differenzberechnung Bei der Differenzbildung für jedes Versionspaar ist wählbar, auf welcher Textebene der Algorithmus die Texte auf Differenzen untersuchen soll. Dabei ist immer die kleinste Einheit anzugeben. Zur Auswahl stehen eine zeichenbasierte, wortbasierte, satzbasierte oder paragraphbasierte Granularität. Kapitel 3.1 zeigte, dass die Effizienz des Verfahrens von der Differenzauflösung abhängt.

Um den Einfluss der Differenzgranularität auf die Retrieval-Qualität zu bewerten, werden wort-, satz- und paragraphbasierte Auflösungen untersucht. Auf eine zeichenbasierte Betrachtung wird aufgrund des hohen Berechnungsaufwands verzichtet.

Attributmenge zur Partitionierung offener Textoperationen Entsprechend den Ausführungen in Kapitel 3.2.1 sind offene Textoperationen hinsichtlich einer Effizienzsteigerung zu partitionieren. Zur Partitionierung stehen Attribute zur Auswahl, die sich aus dem Textoperationstupel ergeben und beliebig untereinander kombinierbar sind.

Zur Messung der Relevanz der Partitionierungsstrategie werden verschiedene Attributmengen miteinander verglichen. Es werden dabei die durchschnittlich durchgeführten Vergleichsoperationen gegenübergestellt, die im Aktualisierungsschritt des Graphen zur Ermittlung einer lokalen Zurücknahme je nach Partitionierung aufzuwenden sind.

Als Attributmenge A werden untersucht:

$$A = \{\text{Operationstyp}\}$$
$$A = \{\text{Operationstyp, Position}\}$$
$$A = \{\text{Operationstyp, Position, Zeichengranularität}\}$$

4.3 Analyse der Erkennung von Bearbeitungskonflikten

Die Evaluierung des *Edward*-Algorithmus mittels des Referenzkorpus hat ergeben, dass Bearbeitungskonflikte automatisch erkennbar sind. An Bearbeitungskonflikten beteiligte Versionen, die per Hand dokumentiert wurden und als Referenz dienten, sind mit einer Precision von $0,95$ bei einem Recall von $0,90$ automatisch zu erfassen. Die Angaben beziehen sich dabei auf die tatsächlich erkannten Bearbeitungskonflikte. Zwei der 51 dokumentierten Benutzerkonflikte werden nicht gefunden, da durch die Differenzberechnung andere Textoperationen ermittelt wurden als in der dokumentieren Referenz. Die Tabelle 4.3 gibt einen Überblick über das erzielte Retrieval-Ergebnis, welches mit den in Tabelle 4.4 aufgeführten Belegungen für die in Kapitel 4.2.2 aufgeführten Parameter zu erzielen.

Typ	Ebene	Bearbeitungskonflikte (#)	Precision	Recall
Einfügen–Löschen	Gesamt	28 von 29	0,97	0,86
	Zeichen	2 von 2	1,00	0,83
	Wörter	5 von 5	0,93	0,84
	Sätze	10 von 11	0,97	0,82
	Paragraphen	11 von 11	0,98	0,91
Ersetzen	Gesamt	21 von 22	0,93	0,96
	Zeichen	1 von 2	0,71	0,78
	Wörter	6 von 6	0,86	0,92
	Sätze	8 von 8	1,00	1,00
	Paragraphen	6 von 6	0,96	0,97
Alle		49 von 51	0,95	0,90

Tabelle 4.3 *: Anzahl festgestellter Bearbeitungskonflikte in der englischen Wikipedia für jede Ebene von Bearbeitungskonflikten. Precision und Recall beziehen sich ausschließlich auf erfasste Bearbeitungskonflikte. Die dritte Spalte gibt an, wie viele dieser in jeder Ebene erkannt werden.*

Parameter	Einstellungen
Differenz-Granularität	Wörter
Partitionierung	$A = \{\text{Operationstyp}, \text{Position}, \text{Zeichengranularität}\}$

Positionsauflösung

Ebene	Position
Zeichen	$k = 1$
Wörter, Sätze	$k = 5$
Paragraphen	$k = 7$

Ähnlichkeitsmetriken

Metrik	Klasse	Schwellwert
Levenshtein Distanz	Zeichen	$\sigma = 1$
SoftTF-IDF	Wörter, Sätze	$\varphi = 0,85$
Fuzzy Fingerprinting	Paragraphen	-

Tabelle 4.4 *: Optimale Einstellungen für die Parameter des* Edward-*Algorithmus zur Erreichung der Ergebnisse in Tabelle 4.3.*

Bei der Auswertung wird zunächst der Einfluss der einzelnen Parameter auf die Retrieval-Qualität des *Edward*-Algorithmus diskutiert. Abschließend werden die Ergebnisse zusammengefasst.

Diskussion der Ähnlichkeitsstrategie Zur Bestätigung der Annahme, dass eine Ausrichtung auf lokale Zurücknahmen und die Verwendung einer Ähnlichkeitsstrategie bei der Erfassung von Bearbeitungskonflikten mehr Meinungsverschiedenheiten erkennt, wurde der *Edward*-Algorithmus mit dem bisherigen globalen Ansatz verglichen. Abbildung 4.3 stellt die erzielten Ergebnisse der Verfahren schematisch dar. Im Ergebnis ist festzuhalten, dass die in dieser Ausarbeitung vorgeschlagene Herangehensweise die besten Ergebnisse erzielt. Ein globales Verfahren, welches ausschließlich identische Versionen berücksichtigt, zeichnet sich durch seine hohe Precision von insgesamt 0,99 aus, wobei jedoch der im Vergleich schlechteste Recall in Höhe von $0,72$ erzielt wird. Ein Verfahren, welches auf lokalen Zurücknahmen basiert und ebenfalls identische Zeichenfolgen betrachtet, erzielt ebenfalls bezüglich des Recalls ein schlechteres Ergebnis als der *Edward*-Algorithmus. Der Unterschied fällt hier jedoch deutlich geringer aus. Dies spricht dafür, dass es sich bei den konfliktbezogenen Zeichenfolgen fast ausschließlich um identische handelt, aber auch ähnliche vorhanden sind.

Diskussion der Ähnlichkeitsschwellwerte Die Wahl eines Ähnlichkeitsschwellwerts φ hat deutlich geringeren Einfluss auf die Retrieval-Qualität als zuvor angenommen. Abbildung 4.4 auf Seite 56 verdeutlicht, dass für fest gewählte Positionsauflösungen die Retrieval-Qualität durch Wahl von φ nur marginal beeinflussbar ist. Wie erwartet, führt eine hohe Belegung von φ zu einer besseren Precision bei einem schlechteren Recall. Entsprechend verhält es sich umgekehrt bei einem niedrig gewählten Schwellwert. Interessant ist, dass sich für die Ebenen der Zeichen und Paragraphen keine Veränderungen einstellen. Weiterhin hat die Ähnlichkeitsstrategie keine Relevanz hinsichtlich der Retrieval-Qualität bei zeichengenauer Positionsauflösung. Die konfliktbezogenen Zeichenfolgen sind hier identisch, so dass keine Auswirkungen auf die Retrieval-Qualität durch Variation von φ zu beobachten sind. Dies bestätigt die Ergebnisse von Kittur u. a. (2007), wonach 94% aller Zurücknahmen identische Versionen betrafen.

Bei einer geringen Ähnlichkeitsschwelle wäre zu erwarten gewesen, dass die Precision deutlicher abnimmt als es die Ergebnissen wiedergeben. Erst durch eine manuelle Stichprobe wird deutlich, dass die Zahl falsch erkannter Bearbeitungskonflikte bei Schwellwerten mit $\varphi \leq 0,6$ deutlich zunimmt.

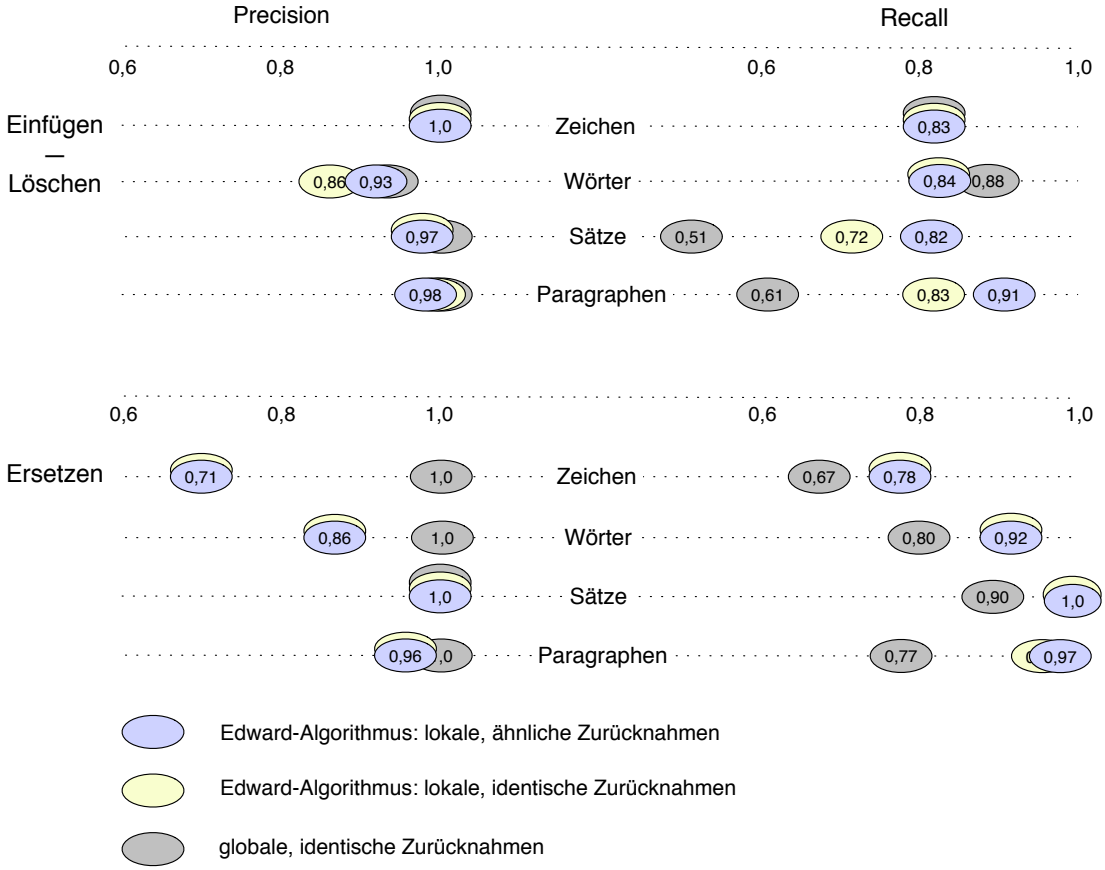

Abbildung 4.3 *: Vergleich dreier verschiedener Ansätze zur automatischen Erkennung von Bearbeitungskonflikten mittels Precision und Recall. Die Verfahren sind anhand ihrer Farbe voneinander getrennt. Liegen diese vertikal übereinander, so besitzen diese den gleichen Wert.*

Diskussion der Positionsauflösung Die Auflösung der Positionsangabe im Text besitzt den stärksten Einfluss auf die Retrieval-Qualität. Abbildung 4.5 auf Seite 57 belegt diese Annahme. Eine zeichengenaue oder satzgenaue Positionsauflösung maximiert hierbei die Precision, führt zugleich jedoch zu einem vergleichsweise schlechten Recall. In Hinblick auf den Recall ist eine sprunghafte Zunahme um etwa 0,2 bei Zeichen, Wörtern und Paragraphen zu verzeichnen, sobald für die Position $k \geq 3$ gilt. Die Precision wird in dem Zusammenhang nur minimal schlechter bzw. verändert sich nicht. Diese Tatsache belegt, dass bei Bearbeitungskonflikten konfliktbezogene Zeichenfolgen ihre Positionen im Text wechseln und somit nicht von identischen Versionen auszugehen ist. Eine zeichengenaue Positionsangabe schränkt die Menge offener Textoperationen zur Feststellung einer lokalen Zurückname zu stark ein, so dass für $3 \leq k \leq 10$ gute Ergebnisse zu erzielen sind.

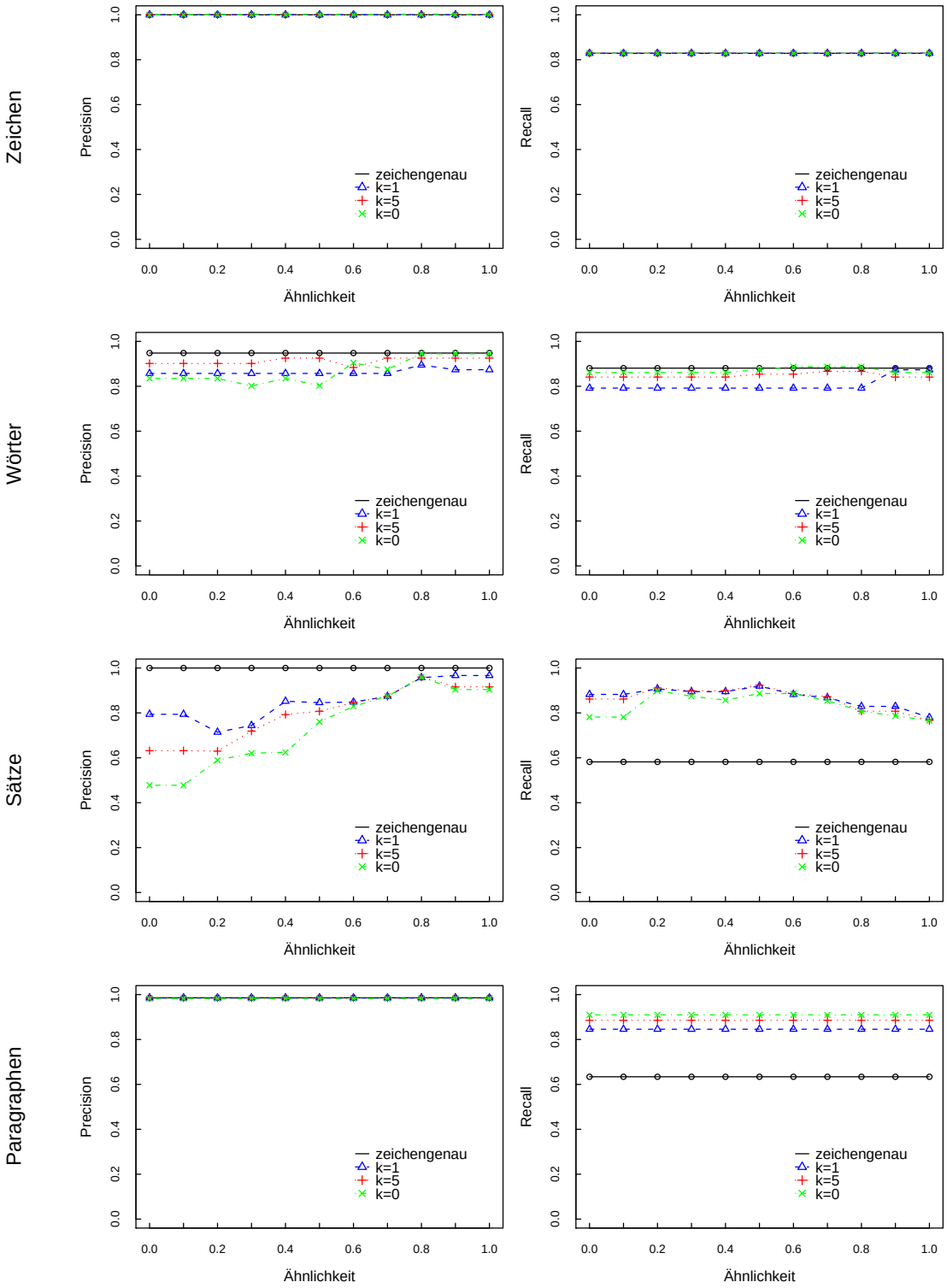

Abbildung 4.4 *: Die Abbildung zeigt den Einfluss der Ähnlichkeit auf die Retrieval-Qualität bei fest gewählter Positionsauflösung. Die Ergebnisse sind anhand der Ebenen von Bearbeitungskonflikten unterteilt.*

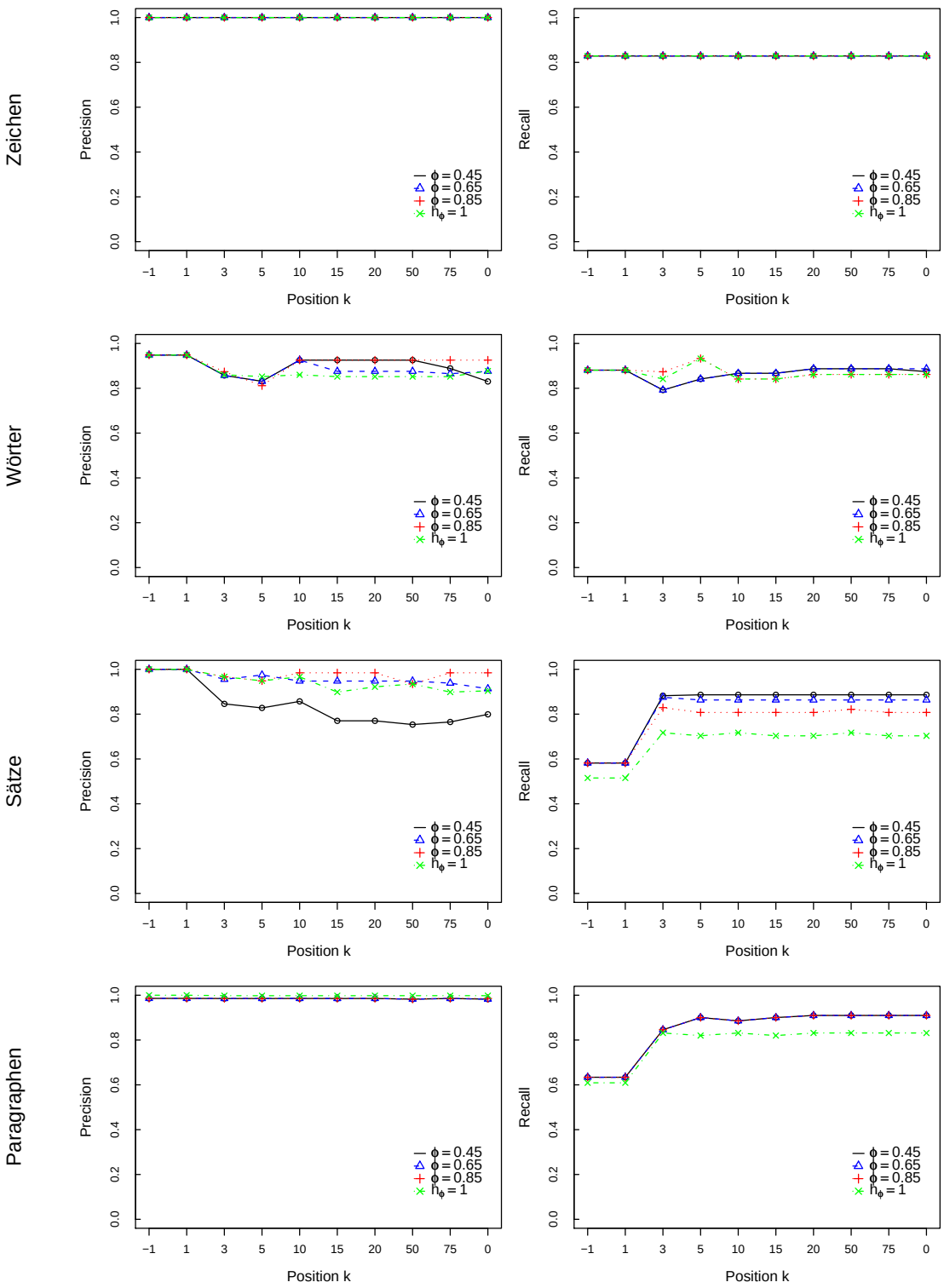

Abbildung 4.5 *: Die Abbildung zeigt den Einfluss der Positionsauflösung auf die Retrieval-Qualität bei fest gewählten Ähnlichkeitsschwellwerten. Eine Position von $k = -1$ bedeutet eine zeichengenaue Auflösung, $k = 0$ den Verzicht auf eine Positionsangabe.*

Wird auf eine Positionsangabe verzichtet, $k = 0$, so zeigen die Ergebnisse keine erkennbare Verschlechterung der Retrieval-Qualität. Auch hier belegt eine manuelle Stichprobe aller entdeckten Bearbeitungskonflikte für einen Artikel, dass bei dem Verzicht einer Positionsangabe in Kombination mit einer geringen Ähnlichkeitsschwelle deutlich mehr Bearbeitungskonflikte erfasst werden, die tatsächlich keine darstellen.

Interessant ist auch, dass vor allem bei der Erkennung von Bearbeitungskonflikten der Ebene Zeichen keine Verschlechterung der Retrieval-Qualität bei einer unschärferen Positionsauflösung eintritt. Vielmehr wäre zu erwarten gewesen, dass die Precision, wie es im Fall der Klasse Wörter der Fall ist, abnimmt. Es ist anzunehmen, dass konfliktbezogene Zeichenfolgen zwar ermittelt werden, aber die Zuordnung der Zeichenfolge im Text nicht korrekt ist. Bei häufig vorkommenden Wörtern im Text wie Artikelnamen oder Pronomen, für die für $k = 0$ die gleiche Position zugeordnet wird, ist dies mit hoher Wahrscheinlichkeit der Fall.

Diskussion der Granularität der Differenzberechnung Die erzielte Retrieval-Qualität des *Edward*-Algorithmus belegt, dass wortbasierte Differenzen gute Ergebnisse erzielen. In Anbetracht der Häufigkeit, in der einzelne Wörter oder Satzteile Bestandteil eines Bearbeitungskonflikts sind, sind diese gröberen Granularitäten vorzuziehen.

Eine Differenzberechnung auf Paragraphebene ist aus folgenden zwei Gründen nicht einzusetzen. Zum einen ist aufgrund der längeren Zeichenfolgen, die als Differenz zwischen Versionspaaren erzeugt werden, der konfliktbezogene Textabschnitt nicht mehr eindeutig beschrieben. Andererseits wächst die Anzahl falsch erkannter Bearbeitungskonflikte. Dies ist begründet in der wiederholten Umformulierung von Paragraphen in Wikipedia-Artikeln. Bei einer Differenzberechnung auf Paragraphebene wird bereits bei Änderung eines einzigen Zeichens der vollständige Paragraph als Differenz erkannt. Auf diese Weise erzeugte Differenzen erzielen anhand ihrer Länge hohe Ähnlichkeiten. Dies führt dazu, dass unter Verwendung der Ähnlichkeitsstrategie mit hoher Wahrscheinlichkeit diese fälschlicherweise als Bearbeitungskonflikte erkannt werden. Diese Bearbeitungen sind jedoch eindeutig von einem Bearbeitungskonflikt abzugrenzen, da es sich um normale Weiterentwicklungen des Artikels handelt. Aus diesem Grund sind Granularitätsstufen, die über die Wortebene hinaus gehen, nicht sinnvoll. Stichproben belegten diese Annahmen.

Desweiteren zeigte sich, dass die Granularität der Differenzberechnung für die Laufzeit des *Edward*-Algorithmus relevant ist. Im Ergebnis wird daher eine Differenzberechnung auf Paragraphebene vorangestellt, um anschließend für die erzeugten Differenzen wortbasierte zu ermitteln. Hiermit ist einerseits die Genauigkeit der erzeugten Differenzen

gegeben, andererseits wird die Komplexität einer fein aufgelösten Berechnung dieser auf kürzere Zeichenfolgen reduziert.

Weiterhin zeigte sich, dass der verwendete Differenzalgorithmus in zwei Fällen andere Operationstypen erkannte, als in der Referenz angegeben. Dies betraf die Ersetzungsoperation, die durch die Differenzberechnung als Einfügen und Löschen interpretiert wurde. Durch eine Beschränkung der vom Differenzalgorithmus unterstützten Operationstypen auf Einfügen und Löschen konnten auch die beiden zuvor nicht gefundenen Bearbeitungskonflikte korrekt erkannt werden.

Abschließend ist festzuhalten, dass durch eine wortbasierte Differenzberechnung Bearbeitungskonflikte, deren konfliktbezogene Zeichenfolge aus mehr als einem Wort besteht, bei Ersetzungen getrennt erfasst werden. Im ungünstigsten Fall wird für jedes Wort separat ein Bearbeitungskonflikt erkannt. Diese sind im Nachhinein bei Bedarf durch eine Aufbereitung zusammenzuführen. Dies hat keinen Einfluss auf die Erkennungsleistung.

Diskussion der Partitionierungsstrategie Bei der Partitionierung offener Textoperationen wird die Attributmenge $A = \{\text{Operationstyp}, \text{Position}, \text{Zeichengranularität}\}$ verwendet.

Abbildung 4.5 belegt, dass bei einer stärkeren Partitionierung die Zahl der Vergleiche beim Test auf lokale Zurücknahme bei gleichbleibenden Retrieval-Eigenschaften sinkt. Dies führt zu einer Verbesserung der Laufzeit. Der Unterschied wird deutlicher, je unschärfer die Positionsauflösung ausfällt und desto größer die Zahl relevanter offener Textoperationen wird. Weiterhin ist festzustellen, dass, je mehr Vergleichsoperationen im Mittel durchgeführt werden müssen, die Attributmenge, die zusätzlich anhand der Zeichengranularität partitioniert, deutlich effizienter ist. Hier entfallen unnötige Ähnlichkeitsberechnungen zwischen Zeichenfolgen verschiedener Granularitätsklassen.

Schlußfolgerungen

- Der *Edward*-Algorithmus liefert bessere Ergebnisse als ein Ansatz basierend auf globalen Zurücknahmen. Eine Beschränkung auf identische Zeichenfolgen bei zeichengenauer Positionsauflösung bei der Erkennung führt im Vergleich zu einem deutlich schlechteren Recall.
- Die Festlegung einer Positionsinformation für Textoperationen im Text besitzt einen starken Einfluss auf die Erkennungsqualität. Als günstige Wahl erwies sich $k \in \{3, \ldots, 10\}$.

- Der Ähnlichkeitsschwellwert φ hat nur geringen Einfluss auf die Retrieval-Qualität. Dieser sollte jedoch nicht zu niedrig gewählt werden, um die Zahl falsch erkannter Bearbeitungskonflikte zu minimieren. Als günstige Wahl erwies sich φ aus dem Intervall $(0,6;1]$ zu wählen.
- Die Einbeziehung ähnlicher, lokaler Zurücknahmen besitzt deutlich weniger Relevanz als angenommen. Ergebnisse in Abbildung 4.3 belegen, dass der *Edward*-Algorithmus bei Einschränkung auf identische Zurücknahmen beinahe die gleiche Retrieval-Qualität erreicht als der vorgeschlagene Ansatz auf Basis der Ähnlichkeitsstrategie.
- Solange Versionsgeschichten von Wikipedia-Artikeln nicht vollständig hinsichtlich Bearbeitungskonflikten dokumentiert sind, bleibt offen, wieviele Bearbeitungskonflikte nicht gefunden werden. Weiterhin ist unklar, wie groß der Anteil falsch entdeckter Konflikte ist.
- Jeder einzelne Bearbeitungskonflikt im Referenzkorpus ist zu erkennen, wenn die Differenzberechnung auf die Operationen Einfügen und Löschen beschränkt wird. Dadurch geht jedoch die direkte Zuordnung der sich ersetzenden Zeichenfolgen verloren.

Attributmenge A	Positions-auflösung	Vergleichsop. im Mittel	Laufzeit	Precision	Recall
{O, P, Z}	zeichengenau	1	100%	0,99	0,74
{O, P}	zeichengenau	1	100%	0,99	0,74
{O}	zeichengenau	14	108%	0,99	0,74
{O, P, Z}	$k = 5$	5	146%	0,95	0,90
{O, P}	$k = 5$	8	174%	0,95	0,90
{O}	$k = 5$	283	182%	0,95	0,90
{O, P, Z}	$k = 20$	35	221%	0,95	0,90
{O, P}	$k = 20$	74	358%	0,95	0,90
{O}	$k = 20$	444	364%	0,93	0,86

***Tabelle 4.5** : Vergleich verschiedener Attributmengen zur Partitionierung offener Textoperationen hinsichtlich ihres Einflusses auf die Laufzeit. Ein* O *in der Attributmenge steht für Operationstyp,* P *für Positionsangabe und* Z *für Zeichengranularität.*

4.4 Analyse der englischen Wikipedia in Hinblick auf Bearbeitungskonflikte

Zur Analyse der englischen Wikipedia wurde der Auszug *enwiki-20060816-pages-meta-history* verwendet, der ca. 1,6 Millionen Artikel enthält. Die Untersuchung wurde auf Basis des *Edward*-Algorithmus durchgeführt. In Tabellen dieses Kapitels werden auf diese Weise erkannte Bearbeitungskonflikte mit „lokal“ referenziert.

Bearbeitungskonflikte wurden bei 4,36% der 1.595.168 analysierten Artikel festgestellt. Bei den betroffenen Artikeln sind im Durchschnitt beinahe 10% der Versionen in der Versionsgeschichte Bestandteil eines Konflikts. Jede zehnte Änderung am Artikel ist demnach konfliktbezogen.

Bei der Analyse ließen sich zwei Gruppen von Bearbeitungskonflikten beobachten. Konflikte der ersten zeichnen sich durch eine kurze Konfliktdauer von maximal zehn Versionen aus. Sie bilden 85% aller erkannten Bearbeitungskonflikte, an denen im Mittel drei Benutzer beteiligt sind. Interessant ist hierbei die kurze Konfliktdauer. Ein Grund dafür ist in der in Kapitel 2.1.2 diskutierten *Three-Revert* Richtlinie zu sehen. Ein andere Ursache, die zu einer kurzen Konfliktdauer beiträgt, ist die Tatsache, dass mittels des *Edward*-Algorithmus wiederholter Vandalismus an Artikeln gleichermaßen erfasst wird. Dieser unterscheidet sich strukturell nicht von einem Bearbeitungskonflikt, ist in der Dauer aber deutlich kürzer. Stichproben zeigten jedoch keine erfassten Vandalismusfälle. Interessant ist ebenfalls, dass trotz der geringen Versionslänge drei Benutzer im Durchschnitt an einem Bearbeitungskonflikt beteiligt sind. Es bedarf jedoch oftmals eines unparteiischen Dritten, der den Streit zwischen zwei Benutzern schlichtet und darum eine gültige Version des Artikels wieder einstellt. Eine zusätzlich durchgeführte Untersuchung belegte diese Annahme: Mindestens ein Administrator greift in nahezu jeden zweiten Bearbeitungskonflikt kurzer Länge ein.

Weiterhin existieren Bearbeitungskonflikte, deren Versionslänge sich über zehn Versionen erstreckt. Acht Benutzer sind im Durchschnitt an einem solchen Konflikt beteiligt und Administratoren greifen hier nur in 8% der Fälle ein. Hierbei handelt es sich mit hoher Wahrscheinlichkeit um „versteckte“ Bearbeitungskonflikte. Die Konfliktparteien werden somit durch keinen unparteiischen Dritten „aufgehalten“, so dass sich in der Folge längere Bearbeitungskonflikte entwickeln. Insbesondere in dieser Konstellation ist eine automatische Erkennung von Bearbeitungskonflikten hilfreich, um frühzeitig Konflikte aufzudecken.

Typ	Ebene	Bearbeitungskonflikte	
		lokal	global
Einfügen–Löschen	Gesamt	220.521	91.939
	Zeichen	586	166
	Wörter	88.652	48.590
	Sätze	95.279	27.102
	Paragraphen	36.004	16.081
Ersetzen	Gesamt	139.535	111.671
	Zeichen	892	2.367
	Wörter	95.351	67.099
	Sätze	38.388	37.515
	Paragraphen	4.904	4.690
Alle		360.056	203.610

***Tabelle 4.6** : Anzahl festgestellter Bearbeitungskonflikte in der englischen Wikipedia für jede Ebene von Bearbeitungskonflikten.*

Um die Frage zu beantworten, ob die Einführung lokaler Zurücksetzungen tatsächlich eine größere Anzahl an Bearbeitungskonflikten aufdecken kann, wurde die englische Wikipedia zusätzlich in Hinblick auf globale Zurücknahmen untersucht. Dabei sind ausschließlich identische Zeichenfolgen an einem Bearbeitungskonflikt beteiligt. Die Positionsangabe der Textoperationen ist zeichengenau festgelegt. Es ist anzumerken, dass der gewählte Ansatz identische, lokale Zurücknahmen erkennt und nicht den gesamten Text als eine Einheit auffasst wie es Kapitel 2.2.1 diskutierte. Beide Verfahren sind auf diese Weise jedoch besser miteinander zu vergleichen. Tabelle 4.6 zeigt die aufgedeckten Bearbeitungskonflikte in einer Gegenüberstellung. Sie sind hinsichtlich Typ und Ebene lokaler Bearbeitungskonflikte unterteilt.

Die deutlich höhere Anzahl erkannter lokaler Bearbeitungskonflikte bestätigt die Schlussfolgerungen der Analyse des Referenzkorpus. Der lokale Ansatz ist dem globalen in fast jeder Hinsicht überlegen. Es werden insgesamt 77% mehr Bearbeitungskonflikte erkannt, welches einer Zunahme um 156.446 Konflikte entspricht. Einzig Bearbeitungskonflikte, die vom Typ Ersetzen und auf Basis von Zeichen definiert sind, werden beim globalen Ansatz häufiger erkannt.

Typ der Metrik	Seiten zur Erfassung
Versionen (#)	Artikel, Diskussion
Textlänge	Artikel, Diskussion
Verweise von anderen Artikel	Artikel, Diskussion
Verweise auf andere Artikel	Artikel, Diskussion
Anonyme Änderungen (#, %)	Artikel, Diskussion
Änderungen durch Administratoren (#, %)	Artikel, Diskussion
Geringe Änderungen (#, %)	Artikel, Diskussion
Zurücksetzungen (#)	Artikel

***Tabelle 4.7** : Ein Auszug relevanter Metriken, die von Wikipedia-Seiten gewonnen werden. Eine Raute # steht für die tatsächliche Anzahl, % für den prozentualen Anteil jeder Metrik.*

In Hinblick auf den Fragenkatalog in der Einleitung von Kapitel 4 ist zu beantworten, welche Art von Artikeln in Wikipedia vor allem von Bearbeitungskonflikten betroffen sind. Bei den betroffenen Artikeln handelt es sich überwiegend um *populäre* Artikel. Ein Artikel gilt als populär, sobald die Anzahl der Versionen und die der am Artikel beteiligten Benutzer deutlich über dem Durchschnitt liegen. Durchschnittlich setzt sich ein Artikel der englischen Wikipedia aus 25 Versionen und einer Textlänge von 2.000 Zeichen zusammen, die 14 verschiedene Benutzer gemeinsam verfasst haben.

Ein Artikel, in dem mindestens ein Bearbeitungskonflikt automatisch erkannt wurde, umfasst dagegen durchschnittlich 225 Versionen. Dabei schrieben 118 Benutzer einen 12.000 Zeichen umfassenden Text. Anhand dieser drei Eigenschaften ist ein Artikel mit Bearbeitungskonflikten eindeutig als *populär* zu bezeichnen.

Dass Bearbeitungskonflikte mehrheitlich bei populären Artikel anzutreffen sind, stimmt mit den Ergebnissen von Kittur u. a. (2007) überein. Um durch ein lernbasiertes Verfahren das Konfliktpotential eines Artikel zu messen, stellten die Autoren Artikelmerkmale zusammen. Darunter fallen Eigenschaften wie die Anzahl der Versionen, die Textlänge oder die Zahl der Autoren, die am Artikel beteiligt sind. Tabelle 4.7 listet die verwendeten Merkmale, die als Metriken dienen, auf. Kittur u. a. (2007) stellten bei der Evaluierung ihres Ansatzes bei steigender Versionsanzahl ein höheres Konfliktpotential fest. Das Auftreten von Bearbeitungskonflikten wird demnach bei stark frequentierten Artikeln wahrscheinlicher.

Aufgrund dieser Annahme wurde in der hier vorliegenden Ausarbeitung die Menge der Wikipedia-Artikel hinsichtlich der Kriterien *Versionen* und *Benutzer* eingeschränkt.

Korpusgröße	Einschränkung	Artikel mit Bearbeitungskonflikten	
		lokal	global
1.595.168	keine	69.515	43.166
862.209	$\geq$ 5 Versionen	69.515	43.166
158.048	$\geq$ 50 Versionen	56.261	35.978
71.268	$\geq$ 100 Versionen	40.763	27.861
23.054	$\geq$ 250 Versionen	19.553	15.366
787.858	$\geq$ 5 Benutzer	69.212	42.913
77.883	$\geq$ 50 Benutzer	40.930	27.284
30.686	$\geq$ 100 Benutzer	23.727	17.491
7.992	$\geq$ 250 Benutzer	7.738	6.837

Tabelle 4.8 *: Die Wikipedia-Artikel werden anhand der Metriken* Versionen *und* Benutzer *gefiltert. Eine eindeutige Zunahme des Anteils von Artikeln, bei denen Bearbeitungskonflikte auftreten, ist zu erkennen.*

Übereinstimmend zeigte sich, dass der Anteil von Artikeln mit Bearbeitungskonflikten bei steigender Versions- sowie Benutzerzahl zunimmt. Bereits ab 100 Versionen ist nahezu jeder zweite Artikel betroffen. Bei mehr als 250 Versionen, darunter fallen 23.054 Artikel, ist beinahe jeder Artikel von Bearbeitungskonflikten betroffen. Tabelle 4.8 stellt die Ergebnisse durch Einschränkung der Artikelmenge anhand der genannten Artikelmetriken dar. Bearbeitungskonflikte sind somit vor allem bei populären Artikeln anzutreffen.

Weiterhin interessiert, welchen Kategorien konfliktbetroffene Artikel zuzuordnen sind. Wikipedia-Artikel werden häufig in Kategorien unterteilt, die am Ende eines Artikels im Text verzeichnet sind. Vermutet wird, das in erster Linie bei kontroversen Themen der Politik und Gesellschaft Meinungsverschiedenheiten auftreten. Die Auswertung zeigte allerdings, dass vielmehr Artikel betroffen sind, die von Personen der Filmbranche und des Musikbusiness handeln. Deutlich zu erkennen ist die Ausrichtung auf den amerikanischen Markt. Tabelle 4.9 zeigt eine Übersicht häufig vorkommender Kategorien.

Neben der Auswertung der Kategorien interessiert, ob eine Liste kontroverser Themen automatisch mittels des *Edward*-Algorithmus generiert werden kann. Es ist bekannt, dass in der englischen Wikipedia eine Liste mit kontroversen Themen gepflegt und monatlich manuell aktualisiert wird (Wikipedia, 2007p). Ein Vergleich mit den dort aufgeführten

Kategorie (z = zusammengefasst)	Artikel
Schauspieler (z)	21.656
Musiker (z)	12.657
Living People	10.170
Pages for deletion	1.370
Roman Catholics (z)	1.222
Gay, lesbian and bisexual people (z)	974
{{disputed}}	939
People from New York (z)	898
Jewish Americans	637
People from California	494
Wikipedia Featured Articles	237

***Tabelle 4.9** : Zuordnung der 69.515 erfassten Artikel mit Bearbeitungskonflikten in Kategorien (ein Auszug). Jeder Artikel kann in mehrere Kategorien aufgenommen werden, so dass Überschneidungen wahrscheinlich sind. Für einige Kategorien wurde ein Oberbegriff gewählt, der diese zusammenfasst.*

Artikeln belegt, dass 76% von diesen Bearbeitungskonflikte aufweisen. In der Konsequenz ist zukünftig diese manuelle Aufstellung automatisch durch das in dieser Ausarbeitung vorgestellte Verfahren generierbar.

Zusammenfassend ist der prozentuale Anteil an Artikeln, die Bearbeitungskonflikte aufweisen, mit 4,36% gering. Jedoch sind vorzugsweise populäre Artikel davon betroffen. Hier beträgt der Anteil der Versionen, die an einer Kontroverse beteiligt sind 10% eines Artikels. Der längste, zusammenhängend erfasste Bearbeitungskonflikt konnte für den Artikel *Tots TV* dokumentiert werden, bei dem es sich um Kindersendung handelt. Über 279 Versionen hinweg streiten sich Benutzer, ob eine Puppe männlich oder weiblich ist. Die längsten Bearbeitungskonflikte sind in Tabelle 4.10 zusammengefasst. Weitere Beispiele finden sich im Anhang A.

Schlussbemerkungen

- Der *Edward*-Algorithmus erkennt 77% mehr Bearbeitungskonflikte als eine bisherige Erkennung auf Basis globaler Zurücknahmen. Im Ergebnis wurden 25.000 weitere Artikel identifiziert, die Bearbeitungskonflikte aufweisen.

- Die durchschnittliche Konfliktdauer eines Bearbeitungskonflikts ist mit fünf Versionen kurz. Die Ursache hierfür ist die strikte Einhaltung der *Three-Revert* Richtlinie und ein frühzeitiges Eingreifen von Administratoren.
- An der Mehrheit der Bearbeitungskonflikte sind im Durchschnitt drei Benutzer beteiligt. Dies schließt bei einer kurzen Konfliktdauer mit ein, dass ein Schlichter involviert ist.
- Weiterhin existieren viele Bearbeitungskonflikte, die eine Länge von mehr als einhundert Revisionen erreichen. Die Konfliktparteien hindern somit eindeutig andere Benutzer an einer vernünftigen Weiterentwicklung des Artikels. Ein Administrator wurde auf einen solchen Konflikt nicht aufmerksam.
- Bearbeitungskonflikte sind vor allem in populären Artikeln anzutreffen und betreffen einen Benutzer, der solche Artikel bearbeitet, mit hoher Wahrscheinlichkeit.

Seitennummer	Artikel	Bearbeitungskonflikte	
		Anteil	Versionslänge
1387517	Tots TV	78%	279
14229	Homeopathy	22%	254
1268106	Matt Leinart	49%	250
32565	Sildenafil	10%	224
232384	Qipao	63%	221
34289	Yasser Arafat	24%	194
155019	Hanging	42%	192
3199	Augusto Pinochet	9%	188
52269	Edward Heath	35%	177

***Tabelle 4.10** : Auflistung der Artikel mit den längsten, automatisch erkannten Bearbeitungskonflikten. Die Spalte „Anteil“ gibt den prozentualen Anteil der Versionen eines Artikels an, die an Bearbeitungskonflikten beteiligt sind.*

5 Zusammenfassung und Ausblick

Gegenstand dieser Ausarbeitung ist die automatische Erkennung von Bearbeitungskonflikten in Wikipedia.

Kapitel 2.1.1 diskutierte hierzu die Ursachen von Bearbeitungskonflikten. In diesem Zusammenhang wurde die Hypothese aufgestellt, dass der geringe Aufwand, um Bearbeitungen an Artikeln rückgängig zu machen, dazu führt, dass Benutzer bei Meinungsverschiedenheiten eher auf eine Diskussion verzichten und direkt Änderungen erneut im Artikeltext vornehmen. Die Versionsgeschichte ermöglicht es Benutzern weiterhin, gezielt frühere Versionen eines Artikels mit vernachlässigbarem Aufwand wieder einzustellen. Dies begünstigt Bearbeitungskonflikte in Wikis und insbesondere in Wikipedia.

Bei einer strukturellen Untersuchung von Bearbeitungskonflikten, die über die Arbeiten früherer Autoren weit hinaus geht, wurden in dieser Ausarbeitung Restriktionen bei der bisherigen globalen Betrachtung von Bearbeitungskonflikten festgestellt. Im Ergebnis sind Bearbeitungskonflikte in ihrer Struktur komplexer als bisher angenommen. Der Begriff des Bearbeitungskonflikts wurde in der Konsequenz neu gefasst, welches zur Definition sogenannter lokaler Bearbeitungskonflikte in Kapitel 2.2.2 führte. Fortan werden zwei Arten von Bearbeitungskonflikten, globale und lokale, unterschieden. Eine entsprechende Taxonomie fasst verschiedene, beobachtete Typen lokaler Bearbeitungskonflikte zusammen. In diesem Zusammenhang wurde die Hypothese aufgestellt, dass sich Benutzer nicht nur um identische Zeichenfolgen streiten, sondern vermehrt auch konfliktbezogene Textstellen im Verlauf des Bearbeitungskonflikts umformulieren.

Maßnahmen, die Wikipedia gegenwärtig unternimmt, um Bearbeitungskonflikte zu schlichten, setzen eine manuelle Sichtung von Artikeln voraus. Dies stellt bei über zwei Millionen Artikeln alleine in der englischen Wikipedia eine große Herausforderung dar. Ein automatisches Werkzeug würde die Arbeit der freiwilligen Helfer und Administratoren, die sich um die Schlichtung von Meinungsverschiedenheiten bemühen, mit hoher Wahrscheinlichkeit unterstützen. Aktuell sind keine Arbeiten bekannt, die ein solches Verfahren vorgestellt haben, so dass in dieser Ausarbeitung erstmalig eine automatische Erkennung von Bearbeitungskonflikten in Kapitel 3 auf Basis des *Edward*-Algorithmus

vorgestellt wurde. Dieser erkennt in einer Textoperationsgeschichte, die durch Differenzbildung aus der Versionsgeschichte eines Artikels erzeugt wird, Bearbeitungskonflikte auf lokaler Ebene. Bei der automatischen Erkennung dieser wurden ebenfalls ähnliche, lokale Zurücknahmen erfasst.

Zur Evaluierung des Verfahrens in Kapitel 4 wurde erstmalig ein Referenzkorpus per Hand konstruiert. Dieser stellt bei Experimenten die Referenz dar, an der die Erkennungsqualität des Algorithmus gemessen wird. Bei einer geeigneten Parametrisierung des *Edward*-Algorithmus wird eine Precision von $0,95$ und ein Recall von $0,90$ erreicht. Zwei der 51 dokumentieren Bearbeitungskonflikte konnten aufgrund einer falschen Zuweisung von Textoperationstypen durch die Differenzberechnung nicht erkannt werden.

Die Ergebnisse der Evaluierung des Referenzkorpus zeigen weiterhin, dass der Anteil ähnlicher, lokaler Zurücknahmen deutlich geringer ist als angenommen. Der Referenzkorpus ist aufgrund der geringen Größe jedoch nicht repräsentativ, so dass gegenwärtig keine eindeutigen Aussagen über das Verhältnis von ähnlichen zu identischen Zurücknahmen in Wikipedia getroffen werden kann.

Eine Analyse der englischen Wikipedia belegte abschließend, dass mittels des *Edward*-Algorithmus eine deutliche Steigerung bei der Erkennung von Bearbeitungskonflikten erzielt wird – 77% mehr Bearbeitungskonflikte sind gegenüber einer globalen Erkennung identifiziert worden. Hier zeigte sich, dass eine lokale Betrachtung einer globalen in jeder Hinsicht überlegen ist. Ebenso wie für die Analyse des Referenzkorpus ist jedoch unklar, wie stark der Einfluss ähnlicher, lokaler Zurücknahmen auf die gesamte Erkennungsleistung ist.

Ausblick

Bei der Betrachtung der Ergebnisse bleibt die Frage unbeantwortet, ob Bearbeitungskonflikte mittels der vorgestellten Herangehensweise tatsächlich vollständig erkannt werden. Die Ergebnisse, die sich auf die Vollständigkeit beziehen, wurden auf einem Referenzkorpus generiert, in dem pro Artikel nur wenige Bearbeitungskonflikte aus Teilbereichen der Versionsgeschichte dokumentiert sind. Eine vollständige Erfassung erfordert eine zeitaufwendige, manuelle Sichtung der gesamten Versionsgeschichte eines Artikels. In weiteren Arbeiten ist gleichwohl notwendig, die Menge der dokumentierten Bearbeitungskonflikte deutlich zu erweitern, um eindeutigere Aussagen über die Retrieval-Qualitäten treffen zu können. Insbesondere sind dabei Bearbeitungskonflikte vollständig für Artikel aus Wikipedia manuell zu beschreiben. Dient dies als Referenz zur Evaluierung des *Edward*-

Algorithmus, sind Schlussfolgerungen über die Offenlegung von Bearbeitungskonflikten hinsichtlich der vollständigen Erfassung beteiligter Versionen zu ziehen. Solange diese Evaluierungssituation nicht geschaffen ist, sind ebenso keine exakten Annahmen über die Genauigkeit der Ergebnisse zu machen. Anhand einer unvollständigen Dokumentation des Bearbeitungskonflikts, indem beispielsweise nur ein Teilbereich der Versionsgeschichte dokumentiert wurde, erkennt ein automatisches Verfahren mit hoher Wahrscheinlichkeit mehr relevante, beteiligte Versionen.

Im Zuge einer vollständigen Dokumentation ist der tatsächliche Anteil an ähnlichen Zurücknahmen zu bestimmen, um eine Aussage darüber zu treffen, ob der zusätzliche Aufwand der Ähnlichkeitsberechnung hinsichtlich einer besseren Erkennungsleistung lohnenswert ist.

Desweiteren bietet sich zukünftig die Möglichkeit, das bestehende System auf ein *online* Szenario zu adaptieren. Der *Edward*-Algorithmus würde in diesem Fall direkt auf die Datenbank von Wikipedia zugreifen. Jede neue Änderung in Wikipedia ist direkt zu analysieren, um frühzeitig Bearbeitungskonflikte identifizieren zu können. Aufgrund der großen Menge an Bearbeitungen, die täglich vorgenommen werden, sind spezielle Anforderungen an den Algorithmus zu stellen. Versionen eines Artikels müssen zum einen schrittweise verarbeitet werden. In der Konsequenz ist bei neuen Bearbeitungen keine vollständige Analyse der vorliegenden Versionsgeschichte eines Artikel durchzuführen. Diese besteht zum Teil aus bis zu 10.000 Versionen. Bei mehreren Bearbeitungen in Folge an einem Artikel dieser Größe wären zigtausende Versionen mehrfach unnötig zu analysieren.

Die erste Forderung hinsichtlich einer Adaption auf ein online Szenario ist bereits erfüllt. Eine Serialisierung eines sogenannten *Konflikt-Zustands* eines Artikels, der bei neu hinzugefügten Versionen einzulesen ist, ist dagegen bislang noch offen. Im Ergebnis ist ausschließlich die neue Version zu analysieren und der neue Zustand des Artikels wieder zu speichern. Alle Artikel in Wikipedia sind in der Konsequenz einmalig *offline* zu analysieren und dienen als Ausgangsbasis für neue Änderungen.

In der Motivation wurde bereits diskutiert, dass bislang Bearbeitungskonflikte ausschließlich in Wikipedia beobachtet wurden. Spannend ist zu sehen, ob solche auch in Quelltext-Verwaltungssystemen wie dem CVS existieren. Ein entsprechender Differenz-Algorithmus, der auf den Vergleich von Quelltexten spezialisiert ist, wird hierfür benötigt. Als Grundlage der Untersuchung dienen öffentlich zugängliche Repositories wie die der *Apache Software Foundation*.

Abschließend bleibt weiterhin offen, über welche Themen sich die Benutzer in Wikipedia streiten. Durch die Gewinnung der konkreten Zeichenfolgen, um die sich gestritten wird, wird eine umfangreiche Streitanalyse realisierbar. Kommunikationswissenschaftliche Studien sind in Anbetracht dessen denkbar. Ein interessanter Forschungsaspekt, zumal in Wikipedia Benutzer verschiedenster Kulturen aus der ganzen Welt zusammen kommen.

Literaturverzeichnis

[Adler u. de Alfaro 2007] Adler, B. T. ; Alfaro, Luca de: A content-driven reputation system for the wikipedia. In: *WWW '07: Proceedings of the 16th international conference on World Wide Web*. New York, NY, USA : ACM, 2007. – ISBN 978–1–59593–654–7, S. 261–270

[Amitay u. a. 2007] Amitay, Einat ; Yogev, Sivan ; Yom-Tov, Elad: Serial Sharers: Detecting Split Identities of Web Authors. (2007), July. http://einat.webir.org/SIGIR_PAN_workshop_2007.pdf

[Baeza-Yates u. Ribeiro-Neto 1999] Baeza-Yates, Ricardo ; Ribeiro-Neto, Berthier: *Modern Information Retrieval*. Addison Wesley, 1999. – ISBN 020139829X

[Bergroth u. a. 2000] Bergroth, L. ; Hakonen, H. ; Raita, T.: A Survey of Longest Common Subsequence Algorithms. In: *spire* 00 (2000), S. 39. http://dx.doi.org/http://doi.ieeecomputersociety.org/10.1109/SPIRE.2000.878178. – DOI http://doi.ieeecomputersociety.org/10.1109/SPIRE.2000.878178. ISBN 0–7695–0746–8

[Brandt 2006] Brandt, Daniel: *Plagiarism by Wikipedia editors*. http://www.wikipedia-watch.org/psamples.html. Version: 2006. – [Letzter Zugriff am 19.01.2008]

[Buriol u. a. 2006] Buriol, Luciana ; Castillo, Carlos ; Donato, Debora ; Leonardi, Stefano ; Millozzi, Stefano: Temporal Analysis of the Wikigraph. In: *Proceedings of the Web Intelligence Conference (WI 2006)*. Los Alamitos, CA, USA : IEEE Computer Society, December 2006. – ISBN 0–7695–2747–7, 45–51

[Cohen u. a. 2003] Cohen, William W. ; Ravikumar, Pradeep ; Fienberg, Stephen E.: A Comparison of String Distance Metrics for Name-Matching Tasks. In: Kambhampati, Subbarao (Hrsg.) ; Knoblock, Craig A. (Hrsg.) ; Kambhampati, Subbarao (Hrsg.) ; Knoblock, Craig A. (Hrsg.): *IIWeb*, 2003, 73–78

[Ferber 2003] Ferber, Reginald: *Information Retrieval: Suchmodelle und Data-Mining-Verfahren für Textsammlungen und das Web*. Heidelberg : dpunkt Verlag, 2003 http://information-retrieval.de/

[Free Software Foundation 2002] Free Software Foundation: *GNU Free Documentation Licence*. http://www.gnu.org/licenses/fdl.txt. Version: 2002. – [Letzter Zugriff am 23.10.2007]

[Griffith 2007] Griffith, Virgil: *WikiScanner*. http://wikiscanner.virgil.gr/. Version: 2007. – [Letzter Zugriff am 21.11.2007]

[Hirschberg 1975] Hirschberg, D. S.: A linear space algorithm for computing maximal common subsequences. In: *Commun. ACM* 18 (1975), Nr. 6, S. 341–343. http://dx.doi.org/http://doi.acm.org/10.1145/360825.360861. – DOI http://doi.acm.org/10.1145/360825.360861. – ISSN 0001–0782

[Hoppe 2006] Hoppe, Dennis: *Ähnlichkeitssuche anhand hashing-basierter Verfahren. Ein Anwendungsszenario*. Seminarausarbeitung an der Bauhaus-Universität Weimar, Dezember 2006

[Jaro 1995] Jaro, M. A.: Probabilistic Linkage of Large Public Health Data Files. In: *Statistics in Medicine* 14 (1995), S. 491–498

[Karwath 2007] Karwath, André: *Zurücksetzstatistik*. http://tools.wikimedia.de/~aka/cgi-bin/revstat.cgi. Version: 2007. – [Letzter Zugriff am 25.11.2007]

[Kittur u. a. 2007] Kittur, Aniket ; Suh, Bongwon ; Pendleton, Bryan A. ; Chi, Ed H.: He says, she says: conflict and coordination in Wikipedia. In: *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM, 2007. – ISBN 978–1–59593–593–9, S. 453–462

[Kleinz 2007] Kleinz, Torsten: *Politische Grabenkämpfe in der Wikipedia*. http://www.heise.de/newsticker/meldung/print/96265. Version: September 2007. – [Letzter Zugriff am 21.11.2007]

[Leppik 2004] Leppik, Peter: *How Authoritative is Wikipedia*. http://frozennorth.org/C2011481421/E652809545. Version: November 2004. – [Letzter Zugriff am 18.11.2007]

[Levenshtein 1966] Levenshtein, Vladimir: *Binary codes capable of correcting deletions, insertions, and reversals*. Soviet Physics Doklady 10 (1966): Seiten 707-710, 1966

[Mehler 2005] Mehler, Alexander: *Korpuslinguistik. Themenheft*. Bd. 2. Gesellschaft für linguistische Datenverarbeitung, 2005

[Möller 2006] Möller, Erik: *Die heimliche Medienrevolution*. 2. Auflage. Heise Zeitschriften Verlag, 2006. – 177 – 179 S.

[Myers 1986] Myers, E: An O(ND) difference algorithm and its variations. In: *Algorithmica* (1986), Jan. http://www.springerlink.com/index/L32685XQ65223882.pdf

[Neus 2001] Neus, Andreas: *Managing Information Quality in Virtual Communities of Practice*. 2001

[Nupedia 2001] Nupedia: *Nupedia: Editorial Policy Guidelines*. Online. Letzter Zugriff am 23. Oktober 2007 auf http://nupedia.8media.org/policy.shtml, 2001. – Version 4.04 vom 19. Juli 2001.

[Nupedia 2003] Nupedia: *Nupedia, the free encyclopedia*. Online. Letzter Zugriff am 23. Oktober 2007 auf http://web.archive.org/web/20030810153103/www.nupedia.com, 2003. – Archivierte Internetseite.

[Salton u. Lesk 1968] Salton, G. ; Lesk, M. E.: Computer Evaluation of Indexing and Text Processing. In: *J. ACM* 15 (1968), Nr. 1, S. 8–36. http://dx.doi.org/http://doi.acm.org/10.1145/321439.321441. – DOI http://doi.acm.org/10.1145/321439.321441. – ISSN 0004–5411

[Stein 2005] Stein, Benno: Fuzzy-Fingerprints for Text-Based Information Retrieval. In: *5th International Conference on Knowledge Management (I-KNOW 05), Graz. Journal of Universal Computer Science* (2005), S. 572–579

[Stein u. Potthast 2006] Stein, Benno ; Potthast, Martin: *Hashing-basierte Indizierung: Anwendungsszenarienm Theorie und Methoden*. Bauhaus Universität Weimar, 2006

[Viégas u. a. 2004] Viégas, Fernanda B. ; Wattenberg, Martin ; Dave, Kushal: Studying cooperation and conflict between authors with history flow visualizations. In: *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM, 2004. – ISBN 1–58113–702–8, S. 575–582

[Wales 2001] Wales, Jimmy: *Statement of principles — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/User:Jimbo_Wales/Statement_of_principles. Version: 10 2001. – [Letzter Zugriff am 20.11.2007]

[Wiegand 2007] Wiegand, Dorothee: Entdeckungsreise - Digitale Enzyklopädien erklären die Welt. In: *c't - Magazin für Computertechnik, Heise Zeitschriften Verlag.* 06 (2007), S. 136–145

[Wikimedia 2003] Wikimedia: *Edit wars — Wikipedia, The Free Encyclopedia.* http://meta.wikimedia.org/w/index.php?title=Edit_wars&oldid=20957. Version: 2003. – [Letzter Zugriff am 19.01.2008]

[Wikipedia 2005] Wikipedia: *Talk:Avril Lavigne/Archive 1.* http://en.wikipedia.org/wiki/Talk:Avril_Lavigne/Archive_1. Version: 2005. – [Letzter Zugriff am 24.11.2007]

[Wikipedia 2006] Wikipedia: *Cyclone Larry: Discussion — Wikipedia, The Free Encyclopedia.* http://en.wikipedia.org/wiki/Talk:Cyclone_Larry/Archive_2. Version: 2006. – [Letzter Zugriff am 24.11.2007]

[Wikipedia 2007a] Wikipedia: *Cyclone Larry — Wikipedia, The Free Encyclopedia.* http://en.wikipedia.org/wiki/Cyclone_Larry. Version: 2007. – [Letzter Zugriff am 19.01.2008]

[Wikipedia 2007b] Wikipedia: *John Seigenthaler Sr.: Wikipedia biography controversy — Wikipedia, The Free Encyclopedia.* http://en.wikipedia.org/wiki/John_Seigenthaler_Sr._Wikipedia_biography_controversy. Version: 2007. – [Letzter Zugriff am 21.11.2007]

[Wikipedia 2007c] Wikipedia: *Kategorie:Wikipedia:Neutralität — Wikipedia, die freie Enzyklopädie.* http://de.wikipedia.org/wiki/Kategorie:Wikipedia:Neutralit%C3%A4t. Version: 2007. – [Letzter Zugriff am 25.11.2007]

[Wikipedia 2007d] Wikipedia: *Neutraler Standpunkt — Wikipedia, die freie Enzyklopädie.* http://de.wikipedia.org/wiki/Wikipedia:Neutraler_Standpunkt. Version: 2007. – [Letzter Zugriff am 21.11.2007]

[Wikipedia 2007e] Wikipedia: *Neutrality Project — Wikipedia, The Free Encyclopedia.* http://en.wikipedia.org/wiki/Wikipedia:Neutrality_Project. Version: 2007. – [Letzter Zugriff am 20.11.2007]

[Wikipedia 2007f] Wikipedia: *NPOV disputes — Wikipedia, The Free Encyclopedia.* Online unter . Zuletzt abgerufen am 20. November 2007. http://en.wikipedia.org/wiki/Category:NPOV_disputes. Version: 2007. – [Letzter Zugriff am 20.11.2007]

[Wikipedia 2007g] Wikipedia: *Vermittlungsausschuss — Wikipedia, die freie Enyklopädie*. http://de.wikipedia.org/wiki/Wikipedia:Vermittlungsausschuss. Version: 2007. – [Letzter Zugriff am 21.11.2007]

[Wikipedia 2007h] Wikipedia: *Vermittlungsausschuss/Archiv — Wikipedia, die freie Enzyklopädie*. http://de.wikipedia.org/wiki/Wikipedia:Vermittlungsausschuss/Archiv. Version: 2007. – [Letzter Zugriff am 21.11.2007]

[Wikipedia 2007i] Wikipedia: *Vermittlungsausschuss/Archiv — Wikipedia, die freie Enzyklopädie*. http://de.wikipedia.org/wiki/Wikipedia:Vermittlungsausschuss/Archiv. Version: 2007. – [Letzter Zugriff am 21.11.2007]

[Wikipedia 2007j] Wikipedia: *Wie schreibe ich einen guten Artikel — Wikipedia, die freie Enzyklopädie*. http://de.wikipedia.org/wiki/Wikipedia:Wie_schreibe_ich_gute_Artikel. Version: 2007. – [Letzter Zugriff am 20.11.2007]

[Wikipedia 2007k] Wikipedia: *Wikiality — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Wikiality. Version: 2007. – [Letzter Zugriff am 21.11.2007]

[Wikipedia 2007l] Wikipedia: *Wikipedia Statistics – Tables – All languages — Wikipedia, The Free Encyclopedia*. http://stats.wikimedia.org/EN/TablesWikipediaZZ.htm. Version: 2007. – [Letzter Zugriff am 23.10.2007; Daten entnommen aus September 2006]

[Wikipedia 2007m] Wikipedia: *Wikipedia:Administrators' noticeboard/3RR — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Wikipedia:Administrators%27_noticeboard/3RR. Version: 2007. – [Letzter Zugriff am 28.01.2008]

[Wikipedia 2007n] Wikipedia: *Wikipedia:Lamest edit wars — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Lamest_edit_wars. Version: 2007. – [Letzter Zugriff am 25.11.2007]

[Wikipedia 2007o] Wikipedia: *Wikipedia:List of controversial issues — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Wikipedia:List_of_controversial_issues. Version: 2007. – [Letzter Zugriff am 25.11.2007]

[Wikipedia 2007p] Wikipedia: *Wikipedia:List of controversial issues — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Wikipedia:List_of_controversial_issues. Version: 2007. – [Letzter Zugriff am 25.11.2007]

[Wikipedia 2007q] Wikipedia: *Wikipedia:Mediation — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Wikipedia:Mediation. Version: 2007. – [Letzter Zugriff am 21.11.2007]

[Wikipedia 2007r] Wikipedia: *Wikipedia:Mediation:Cabal Cases — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Category:Wikipedia_Medcab_closed_case. Version: 2007. – [Letzter Zugriff am 25.11.2007]

[Wikipedia 2007s] Wikipedia: *Wikipedia:Requests for arbitration - Completed requests — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Wikipedia:Requests_for_arbitration/Completed_requests. Version: 2007. – [Letzter Zugriff am 21.11.2007]

[Wikipedia 2007t] Wikipedia: *Wikipedia:Requests for mediation/Archives — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Wikipedia:Requests_for_mediation. Version: 2007. – [Letzter Zugriff am 25.11.2007]

[Wikipedia 2007u] Wikipedia: *Wikipedia:Schiedsgericht — Wikipedia, die freie Enzyklopädie*. http://de.wikipedia.org/wiki/Wikipedia:Schiedsgericht. Version: 2007. – [Letzter Zugriff am 21.11.2007]

[Wikipedia 2007v] Wikipedia: *Wikipedia:Three Revert Rule — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Wikipedia:Three-revertrule. Version: 2007. – [Letzter Zugriff am 08.12.2007]

[Winkler 1999] Winkler, W.: *The state of record linkage and current research problems*. citeseer.ist.psu.edu/article/winkler99state.html. Version: 1999

[Wood 2007] Wood, Craig: *Wikirage*. http://www.wikirage.com/. Version: 2007. – [Letzter Zugriff am 25.11.2007]

A Beispiele für Bearbeitungskonflikte

Die folgenden Beispiele bieten einen kleinen Überblick über Bearbeitungskonflikte.

Nikolaus Kopernikus Ein sehr umstrittener – im Sinne von Streiten – Artikel ist „Nikolaus Kopernikus“[1]. Die „Wikipedianer“ streiten sich bereits seit Jahren um die Nationalität des Astronomen. Das besondere an diesem Streit ist, dass er immer wieder auflebt – erste Anzeichen des Konfliktes sind in der englischen Wikipedia bereits 2002 zu erkennen[2] – und gleich in mehreren Sprachen[3] vorkommt; ein internationaler Bearbeitungskonflikt. Der Konflikt beschäftigt sich mit dem Sachverhalt, ob der Astronom deutscher oder polnischer Herkunft sei. Die deutsche Wikipedia tendierte nach einem zwischenzeitlichen Konsens dazu, Kopernikus als Europäer anzusehen. Ende November 2007 sind keine direkten Aussagen im Sinne von „Kopernikus war ein...“ zu finden. Dem Konflikt wird im Artikel über Kopernikus selbst ein ganzes Unterkapitel gewidmet. Das englische Derivat ist stärker frequentiert als alle anderen regionalen Ableger. Es wurden lange Diskussionen über die Nationalität geführt. Der Artikel musste des öfteren vor Editierungen geschützt werden, da die Bearbeitungskriege überhand nahmen. Zuletzt vom 23. September bis zum 9. November 2007. Hier erhielt der Konsens „Europäer“ die größte Zustimmung. In der polnischen Wikipedia scheint man sich am wenigsten Aufhebens um die Nationalität zu machen, hier ist Kopernikus einfach Pole. Die wichtigsten Fakten sind als Steckbrief in Tabelle A.1 festgehalten.

Abbildung A.1 *: Entwicklung des Streits um den Wirbelsturm Larry. Abgebildet ist jeweils der Informationskasten zu Beginn des Artikels. Rechts ist der aktuelle Konsens zu sehen.*

[1] vgl. Nicolaus Copernicus im Englischen; Nicolas Copernic im Französischen
[2] Es existierte eine eigene Seite, die sich mit der Nationalität von Kopernikus beschäftigte.
[3] Der Konflikt wurde in der deutschen, englischen und französischen Wikipedia beobachtet

Wirbelsturm Larry Der tropische Wirbelsturm *Larry* (Wikipedia, 2007a) erreicht im März 2006 das australische Festland. Sind Ereignisse von allgemeinem Interesse, ist bei Wikipedia eine zeitnahe, intensive Entwicklung von Artikeln zu beobachten. Der Artikel wurde am 19. März 2006 erstellt und unterlag bereits nach 3 Tagen mehr als 500 Änderungen. Es wurde sehr stark zwischen den Autoren diskutiert, ob dieser Wirbelsturm der Kategorie 3, 4 oder 5 angehörte. Auslöser dieses Streits waren unterschiedliche Skalen. In Amerika findet die Saffir-Simpson-Hurrikan-Skala Anwendung, in Australien wird allerdings eine eigene Metrik verwendet. Die Wirbelstürme werden jeweils anhand ihrer Windgeschwindigkeiten eingeordnet, welches zu den Unterschieden in der Einstufung führte. Anstatt über die Fortschritte und Folgen der Zerstörung zu berichten, stritten sich die Autoren über die Einstufung des Wirbelsturm.

Die Diskussion blieb überwiegend sachlich, der eigentliche Bearbeitungskonflikt viel weniger stark aus als der dazugehörige Gedankenaustausch. Zwischendurch echauffiert sich ein Benutzer mit den Worten „*Why are we acting so childish? I like the way it looks right now, listing the category on both scales in the infobox. Australians use Wikipedia too, [..]*“ (Wikipedia, 2006). Letztendlich entsprach diese Meinung dem Konsens: Beide Kategorien wurden im Artikel aufgeführt mit dem Hinweis der jeweiligen zugrunde liegenden Skala – siehe Abbildung A.2.

Die wichtigsten Fakten sind als Steckbrief in Tabelle A.2 festgehalten, wobei der Artikel nicht in dem englischen Wikipedia Auszug enthalten ist, so dass Angaben geschätzt sind.

Steckbrief	Nikolaus Kopernikus
ID	323592
Revisionen	2.292
Benutzer	801
Bearbeitungskonflikte	237
Anteil Konflikte am Artikel	29%
Längster Konflikt (Versionen)	40

***Tabelle A.1** : Steckbrief für den Artikel* Nikolaus Kopernikus *aus der englischen Wikipedia. Stand: August 2006.*

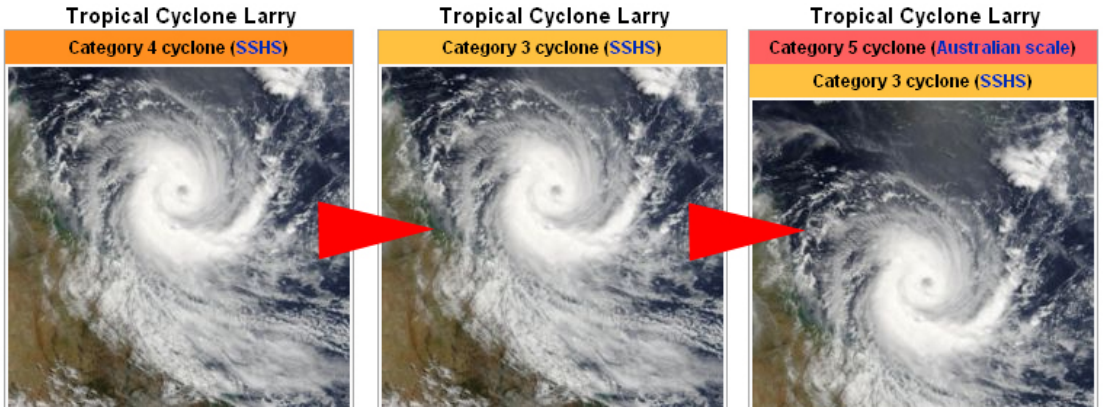

Abbildung A.2 *: Entwicklung des Streits um den Wirbelsturm Larry. Abgebildet ist jeweils der Informationskasten zu Beginn des Artikels. Rechts ist der aktuelle Konsens zu sehen.*

Steckbrief	Cyclone Larry
ID	4440812
Versionen	900
Benutzer	70
Bearbeitungskonflikte	10
Anteil Konflikte am Artikel	40%
Längster Konflikt (Versionen)	40

Tabelle A.2 *: Steckbrief für den Artikel* Nikolaus Kopernicus *aus der englischen Wikipedia. Stand: August 2006.*

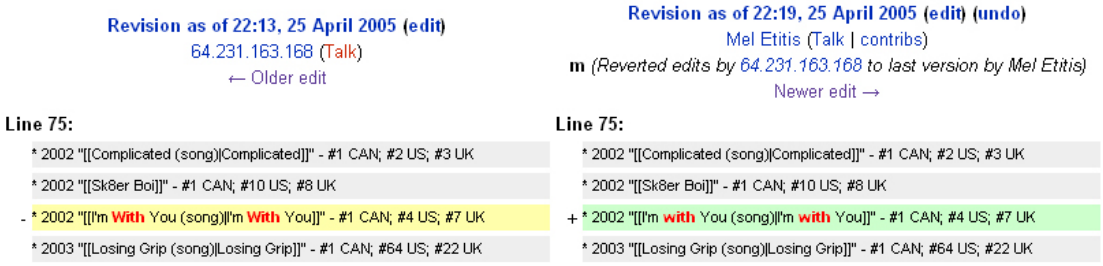

***Abbildung A.3** : Bearbeitungskonflikt hinsichtlich der Schreibweise des Wortes „with“.*

Avril Lavigne Die Sängerin veröffentlichte 2002 ihr Album *Let Go*, auf dem sich unter anderem das Lied „I'm With You“ befand. Das Wort „With“ wurde Bestandteil des Bearbeitungskonfliktes (siehe Abbildung A.3). Die Benutzer stritten sich darüber, ob das erwähnte Wort groß- oder kleingeschrieben werden sollte Wikipedia (2005). Dies ist wiederum typisches Beispiel dafür, dass bei Wikipedia oftmals auch um einzelne Buchstaben oder Zahlen gestritten wird. Im selben Artikel wurde sich ebenso – allerdings nur sehr kurz – darüber gestritten, ob bei der Auflistung der Titel eines Albums die Minutenangabe jedes Liedes in Klammern geschrieben werden sollte oder nicht. Solche Konflikte lassen die sachliche Grundlage der Diskussion vermissen.

Die wichtigsten Fakten sind als Steckbrief in Tabelle A.3 festgehalten.

Steckbrief	Avril Lavigne
ID	165507
Versionen	2.935
Benutzer	1204
Bearbeitungskonflikte	120
Anteil Konflikte am Artikel	16,5%
Längster Konflikt (Versionen)	35

***Tabelle A.3** : Steckbrief für den Artikel* Nikolaus Kopernicus *aus der englischen Wikipedia. Stand: August 2006.*

B Weitere Herausforderungen für Wikipedia

In Wiegand (2007) wurde im März 2007 Wikipedia mit etablierten kommerziellen Enzyklopädien verglichen. Herausgestellt wurde, dass in dem freien Nachschlagewerk nicht mehr Fehler zu finden seien als in alternativen Lexika wie Brockhaus oder Bertelsmann. Trotzdem gebe es viele Artikel mit mangelhafter Qualität, so dass ein Blick in die Revisionshistorie und das Durchlesen der dazugehörigen Diskussionsseite unabdingbar seien. Die Wikipedia-Gemeinde nahm diese Nachricht positiv auf, gleichwohl es aktuell immer noch zentrale Problemfelder in Wikipedia gibt, die durch die Philosophie „anyone can edit“ und dem Verhalten Einzelner in der Gemeinschaft hervorgerufen werden. Im Folgenden werden diese Herausforderungen durch Beispiele verdeutlicht, die letztendlich immer die allgemeine Qualität der Artikel in Frage stellen.

B.1 Neutralität von Artikeln

Der Neutralitätsgedanke[1] gehört zu einem der essentiellen Grundsätze, die vom Wikipedia-Gründer Wales selbst für das erfolgreiche Bestehen von Wikipedia 2001 definiert wurden (Wales, 2001). Die Gemeinschaft von Wikipedia achtet seither entschieden darauf, dass Artikel aus einem neutralen Standpunkt heraus geschrieben sind. Aus dem englischen Sprachraum abgeleitet wird ein Verstoß gegen diese Richtlinie in Kommentaren von Benutzern in der Regel mit der Abkürzung *pov*[2] oder *npov*[3] gekennzeichnet.

Im Gegensatz zu klassischen Nachschlagewerken besteht bei Wikipedia stets die Gefahr, dass Artikel nicht aus einer sachlich-neutralen Sicht geschrieben sind, so dass die Einhaltung dieser Richtlinie unabdingbare Voraussetzung für „gute Artikel“ Wikipedia (2007j) ist. In der englischen Wikipedia haben sich einige sogenannte *Wiki-Projekte* darauf fokussiert, problematisch eingestufte Artikel auf Grundlage der Richtlinien zu überar-

[1] Offizielle Richtlinie von Wikipedia. Siehe Wikipedia (2007d).
[2] engl. für „point of view“
[3] engl. für „neutral point of view“

beiten Wikipedia (2007e). Diese (freiwilligen) Einrichtungen sind erforderlich, schließlich ist die Liste der kontroversen Themen[4] lang Wikipedia (2007p).

Die englischen Archive[5] bezüglich der Verstöße gegen die Neutralität reichen bis August 2006 zurück. Seit Februar 2007 wurden über 4.000 neue Verletzungen der Richtlinie aufgenommen, welches einem Zuwachs von mehr als 120 Fällen pro Woche entspricht (Wikipedia, 2007f). Die aufgezählten Vergehen erscheinen im Vergleich zu zwei Millionen Artikeln der gesamten englischen Wikipedia verschwindend gering, allerdings gehören viele der betroffenen Seiten zu wichtigeren Themen der Politik und Gesellschaft.

Es besteht die Gefahr, dass wenige Autoren bestimmte Themen dominieren, da sie die Majorität der Editierungen stellen. Die Gefahr von Verzerrungen der Fakten ist hier besonders evident. Amitay u. a. (2007) zeigten, dass in der englischen Wikipedia 1.000 Autoren über 1.000 Bearbeitungen vorgenommen haben, wenige von ihnen sogar mehr als 10.000. Diese kleine Gruppe von Benutzern könnte somit – beabsichtigt oder nicht – eine Verzerrung der Informationen hervorrufen. Bei Bearbeitungskonflikten kommt es immer wieder vor, dass Autoren sich zusätzliche Benutzerkonten anlegen, um ihre Beiträge weniger offensichtlich in einem Konflikt durchzusetzen. Benutzer mit mehreren Identitäten werden *Sockenpuppen* genannt, dessen Bezeichnung sich auf einen Bauchredner zurückführen lässt, der sich mit seiner Handpuppe „unterhält“.

Um anonymen Autoren ein Gesicht zu geben, wurde die Software *WikiScanner* entwickelt. Sie ordnet IP-Adressen Organisationen zu. So wurde unter anderem der Hessische Landesverband der CDU bei Manipulationen in Wikipedia ertappt (Griffith, 2007), (Kleinz, 2007).

B.2 Glaubwürdigkeit von Artikeln

Durch den Umstand, dass jeder Teilnehmer jeden Artikel ohne großen Aufwand verändern kann, ist es kritisch zu sehen, inwieweit Wikipedia die Glaubwürdigkeit von Artikeln aufrecht erhalten kann. Der Anspruch einer Enzyklopädie ist kein geringer. Viégas u. a. (2004) zeigten bereits, dass destruktive Textänderungen durch Vandalismus[6] in der Hälfte der analysierten Fälle verhältnismäßig schnell rückgängig gemacht werden, so dass diese nur wenige Minuten als aktuelle Revision dem Betrachter eines Artikels dargeboten

[4] Als kontroverses Thema werden unter anderem Artikel eingestuft, die politisch konfliktgeladen sind oder ethnische Gruppen diffamieren.

[5] Betroffene Artikel können durch Hinzufügen von Textbausteinen wie *{POV}* automatisch Kategorien zugeordnet werden, die problematische Seiten listen.

[6] Die Autoren definierten Vandalismus zum einen anhand des Vorhandenseins von Vulgärausdrücken und zum anderen durch starke Unterschiede in der Textlänge von über 90% zwischen zwei Revisionen.

werden. Im Gegensatz dazu blieben laut Möller (2006) bewußt eingestreute Informationen über einen deutlich längeren Zeitraum unentdeckt. Leppik (2004) untersuchte 2004, ob dem Inhalt der Wikipedia vertraut werden kann. Der Autor fügte im Verlauf von mehreren Tagen fünf subtile Änderungen in Wikipedia ein. Keine von ihnen wurde revidiert, obwohl sie falsche – leicht nachprüfbare – Informationen enthielten. Jede seiner Änderungen blieb mindestens 20 Stunden aktiv, die längste 5 Tage, bevor er sie selbst zurücknahm.

In Brandt (2006) wurden Missbräuche in Wikipedia zusammengetragen. So werden unter anderem zahlreiche Plagiatsvorwürfe gelistet, die in biographischen Artikeln zu finden sind. Ein weiterer Fall, der für Medieninteresse gesorgt hat, ist die *Seigenthaler-Kontroverse* (Wikipedia, 2007b). Ein anonymer Wikipedia-Autor stellte – als Scherz gedacht – eine Beziehung zwischen dem Journalisten Seigenthaler und den Mordanschlägen des ehemaligen amerikanischen Präsidenten John F. Kennedy her. Sein Eintrag verblieb mehrere Monate in Wikipedia.

In seiner satirischen Fernsehsendung rief Steven Colbert seine Zuschauer im Juli 2006 dazu auf, den Artikel *Elephant* so umzuschreiben, dass die Population des Afrikanischen Elefanten sich in den letzten sechs Monaten verdreifacht habe. Er spielte auf seine Wortschöpfung *Wikiality* an, dessen Konzept sei: „Together we can create a reality that we all agree on – the reality we just agreed on.“ (Wikipedia, 2007k)

B.3 Schlußbemerkungen

An den gezeigten Beispielen wird erkennbar, dass Wikipedia mit einer Flut von Missbräuchen konfrontiert wird. Forschungen gehen gegenwärtig den Weg eines aktiven Unterbindens des möglichen Missbrauchs. Adler u. de Alfaro (2007) beschreiben zu diesem Zweck ein Reputationssystem für Wikipedia, in dem jeder Benutzer anhand seiner eigenen Beiträge sein Ansehen und somit seine Glaubwürdigkeit steigern kann. Besucher eines Artikels werden nicht länger mit der aktuellsten Version eines Artikels konfrontiert, sondern mit einer von der Gemeinschaft als „stabil“ markierten Version. Dies dient vor allem der Bekämpfung von Vandalismus.

C XML-Schema Repräsentation für Bearbeitungskonflikte

Das gelistete XML-Schema dient der Definition der XML-Dokumentstruktur zur Dokumentation von Bearbeitungskonflikten in Wikipedia. Es ist in seinem Aufbau einfach gehalten, so dass Benutzer von Wikipedia per Hand Meinungsverschiedenheiten dokumentieren können. Die vorgestellte XML-Dokumentation kann beispielsweise bei Vermittlungsausschüssen Grundlage der Urteilsbildung seitens der Mediatoren sein.

```
1  <!-- W3C Schema describing Edit-Wars in Wikipedia -->
2  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
3    targetNamespace="http://www.uni-weimar.de/m/webis"
4    xmlns:ewd="http://www.uni-weimar.de/m/webis"
5    elementFormDefault="qualified" attributeFormDefault="unqualified">
6    <xs:annotation>
7      <xs:documentation xml:lang="EN">
8        Description language for Edit-Wars in Wikipedia
9          Bauhaus University Weimar
10           Faculty of Media
11           Chair of Web Technology and Information Systems
12       Author: Dennis Hoppe
13       dennis dot hoppe (at) medien dot uni-weimar dot de
14     </xs:documentation>
15   </xs:annotation>
16   <!-- action attribute -->
17   <xs:simpleType name="actionType">
18     <xs:restriction base="xs:string">
19       <xs:enumeration value="inserted" />
20       <xs:enumeration value="removed" />
21       <xs:enumeration value="replaced" />
22     </xs:restriction>
23   </xs:simpleType>
24   <!-- type of edit-war attribute -->
25   <xs:simpleType name="editWarTypeAttributeType">
26     <xs:restriction base="xs:string">
27       <xs:enumeration value="insert-remove" />
28       <xs:enumeration value="move" />
29       <xs:enumeration value="replace" />
```

```
30      </xs:restriction>
31    </xs:simpleType>
32    <!-- complextype of revisions-element -->
33    <xs:complexType name="involvedRevisionsType">
34      <xs:sequence>
35        <xs:element name="revision" type="ewd:revisionType"
36          maxOccurs="unbounded" />
37      </xs:sequence>
38    </xs:complexType>
39    <!-- revision type -->
40    <xs:complexType name="revisionType">
41      <xs:attribute name="section" type="xs:string" use="optional" />
42      <xs:attribute name="revid" type="xs:long" use="required" />
43      <xs:attribute name="position" type="xs:int" use="required" />
44      <xs:attribute name="action" type="ewd:actionType"
45        use="required" />
46      <xs:attribute name="edit-id" type="xs:IDREFS" use="required" />
47      <xs:attribute name="with" type="xs:IDREFS" use="optional" />
48    </xs:complexType>
49    <!-- edits -->
50    <xs:complexType name="involvedEditsType">
51      <xs:sequence>
52        <xs:element name="edit" type="ewd:editType"
53          maxOccurs="unbounded" />
54      </xs:sequence>
55    </xs:complexType>
56    <!-- describe edit-war element -->
57    <xs:complexType name="editWarType">
58      <xs:sequence>
59        <xs:element name="description" type="xs:string"
60          minOccurs="0" />
61        <xs:element name="edits" type="ewd:involvedEditsType" />
62        <xs:element name="revisions"
63          type="ewd:involvedRevisionsType" />
64        <xs:element name="discussion" type="ewd:discussionType"
65          minOccurs="0" />
66      </xs:sequence>
67      <xs:attribute name="type" type="ewd:editWarTypeAttributeType"
68        use="required" />
69      <xs:attribute name="pageid" type="xs:long" use="required" />
70    </xs:complexType>
71    <!-- involved edits -->
72    <xs:complexType name="editType">
73      <xs:simpleContent>
74        <xs:extension base="xs:string">
75          <xs:attribute name="id" type="xs:ID" use="required" />
76        </xs:extension>
77      </xs:simpleContent>
```

```
78    </xs:complexType>
79    <!-- optional discussion element -->
80    <xs:complexType name="discussionType">
81      <xs:attribute name="revid" type="xs:long" use="required" />
82    </xs:complexType>
83    <!-- root element -->
84    <xs:element name="edit-war" type="ewd:editWarType" />
85  </xs:schema>
```