Bauhaus-Universität Weimar
Fakultät Medien
Studiengang Mediensysteme

Entwicklung einer Umgebung zur Evaluierung von Plagiaterkennungsverfahren

Bachelorarbeit

Andreas Eiselt

1. Gutachter: Prof. Dr. Benno Stein
Betreuer: Martin Potthast

Datum der Abgabe: 12. Dezember 2010

Erklärung

Hiermit versichere ich, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Weimar, den 12. Dezember 2010

. . . . . .
Andreas Eiselt

Vorwort

Seit der Antike ist die Menschheit bestrebt, ihr Wissen zu konservieren und in schriftlicher Form festzuhalten. Sammlungen, wie die Bibliothek von Alexandria waren bereits im 3. Jahrhundert v. Chr. mit über 700.000 Schriftrollen eine Informationsquelle für unzählige Gelehrte. Heute existieren weltweit mehr als 130 Millionen verschiedene Bücher (Taycher, 2010), wobei der Öffentlichkeit laut einer Erhebung der Unesco (1998) allein in Europa über 2,5 Milliarden Exemplare in über 200.000 Bibliotheken zur Verfügung stehen. Darüber hinaus wird die Anzahl der über das *World Wide Web* (WWW) frei zugänglichen Dokumente auf 11 - 30 Milliarden geschätzt (Kunder, 2007; Gulli/Signorini, 2005).

Neben den positiven Aspekten, die sich aus dieser breiten Verfügbarkeit von Informationen ergeben, eröffnen sich jedoch auch Möglichkeiten zum Missbrauch. So greifen immer wieder einige Autoren beim Verfassen ihrer Texte auf Werke fremder Autoren zurück und übernehmen diese im Ganzen oder in Teilen, ohne die übernommenen Passagen adäquat zu kennzeichnen. Ein derartiges Vorgehen wird als *Plagiarismus* bezeichnet und wie folgt definiert:

> „Plagiarismus; das unrechtmäßige Nachahmen und Veröffentlichen eines von einem anderen geschaffenen künstlerischen oder wissenschaftlichen Werkes; Diebstahl geistigen Eigentums“ (Duden, 2003)

Derartiger Missbrauch ist kein Phänomen des Informationszeitalters[1]. Bereits in einer Studie von Bowers (1964) gaben 28% von 5.000 befragten Studenten an, mindestens einmal während ihrer Zeit an der Hochschule plagiiert zu haben. Dass das Thema „Plagiate“ an Bedeutung zugenommen hat, machen die Ergebnisse einer jüngeren Studie von McCabe (2005) deutlich, in welcher der Anteil bei 36% lag. Darüber hinaus zeigen Einzelfälle immer wieder, dass nicht nur durch Studierende, sondern auf allen akademischen Ebenen plagiiert wird (Rieble, 2010). Doch nicht nur an den Universitäten und Forschungseinrichtungen, sondern auch in der Industrie und Wirtschaft stellt die

[1] Der Beginn des Informationszeitalters wird etwa mit den 1980er Jahren angegeben und steht in Verbindung mit der immer größer werdenden Bedeutung kommunikativer Netzwerke.

freizügige und unkontrollierte Verwendung von fremden Texten ein schwerwiegendes Problem dar. Besonders betroffen sind hierbei die Bereiche des Marketings (Nitterhouse, 2003), des Journalismus (Clough, 2010) und der Arbeit von Drehbuchautoren (Schilling, 2006).

Die unüberschaubare Menge an Texten macht das Auffinden von Plagiaten und deren Quelle für betroffene Personen und Institutionen nahezu unmöglich. Um die Identifikation von Plagiaten zu erleichtern, beschäftigen sich Forscher im Bereich des Information Retrievals (IR) und der Computerlinguistik mit der Entwicklung von Algorithmen zur automatischen Plagiaterkennung.

Ich selbst begann mich im Sommer 2007 im Rahmen eines Projektes am Lehrstuhl für Webtechnologien & Informationssysteme der *Bauhaus-Universität Weimar* mit Plagiaterkennung zu beschäftigen. Im Jahr 2008 bekam ich die Möglichkeit, an einem Gemeinschaftsprojekt der *Bauhaus-Universität Weimar*, der *Universidad Politécnica de Valencia* (Spanien), der *University of the Aegean* (Griechenland) und der *Bar-Ilan University* (Israel) mitzuarbeiten. Ziel des Projektes war es, eine einheitliche Evaluierungsumgebung für Algorithmen zur Plagiaterkennung zu entwickeln, welche es Wissenschaftlern ermöglichen sollte, die Ergebnisse ihrer Forschung zu evaluieren und zu vergleichen.

Während der „*1st International Competition on Plagiarism Detection*“, welche im Rahmen des *SEPLN´09* Workshops *PAN*[2] veranstaltet wurde, kam die Evaluierungsumgebung ein Jahr später erstmals zum Einsatz. Nach einem intensiven Austausch mit den Wettbewerbsteilnehmern und einer weiteren Entwicklungsphase diente die Evaluierungsumgebung auch 2010 als Grundlage für die „*2nd International Competition on Plagiarism Detection*“. Diese fand während des *PAN 2010 LABs*[3] im Rahmen der *CLEF 2010* Konferenz statt. Die Entwicklung und Analyse des Frameworks, sowie die Auswertung der Ergebnisse sind Gegenstand der folgenden Arbeit.

[2] `http://www.uni-weimar.de/medien/webis/research/workshopseries/pan-09/`
[3] `http://www.uni-weimar.de/medien/webis/research/workshopseries/pan-10/`

Inhaltsverzeichnis

Vorwort **I**

Inhaltsverzeichnis **III**

1 Einleitung **1**

2 Plagiate und Plagiaterkennung **3**
2.1 Plagiatformen 3
2.2 Plagiaterkennung 5
2.2.1 Externe Verfahren 5
2.2.2 Intrinsische Verfahren 6

3 Korpuskonstruktion **8**
3.1 Verwendete Plagiatformen 8
3.2 Strategien zur Erstellung von Plagiaten 9
3.2.1 Künstliche Plagiate 12
3.2.1.1 Übersetzungsplagiate 13
3.2.1.2 Paraphrasenplagiate 14
3.2.2 Simulierte Plagiate 19
3.3 Variablen 22
3.4 Korpuslayout 24
3.5 Plagiatannotation 24
3.6 Konstruktionspipeline 28
3.6.1 Vorverarbeitung 29
3.6.2 Korpuskonstruktion 30
3.6.3 Nachbearbeitung 31

4 Evaluierungsmaße **33**
4.1 Klassische Leistungsmaße im Information Retrieval 33
4.2 Maß zur Quantifizierung der Plagiaterkennungsleistung 34

5 PAN Competition **38**

6 Zusammenfassung **54**

A Danksagung **III**

Abbildungsverzeichnis **IV**

Tabellenverzeichnis **V**

Literaturverzeichnis **VIII**

Kapitel 1

Einleitung

Seit über drei Jahrzehnten erforschen und entwickeln sowohl Wissenschaftler als auch kommerzielle Anbieter Modelle und Algorithmen zur automatischen Plagiaterkennung . Da es sich hierbei um empirische Forschung handelt, kann die Qualität von Algorithmen nur mit Hilfe von Experimenten bewertet werden. Die Wichtigkeit und Herausforderung der Entwicklung aussagekräftiger Experimente beschreiben Basili/Shull/Lanubile (2000) in ihrem Buch „*Using Experiments to Build a Body of Knowledge*“:

> „Experimentation in software engineering is necessary but difficult. Common wisdom, intuition, speculation, and proofs of concept are not reliable sources of credible knowledge.“

Das Ziel von Experimenten auf dem Forschungsgebiet der Plagiaterkennung ist festzustellen, ob und unter welchen Rahmenbedingungen ein Plagiaterkennungsalgorithmus funktioniert. Die Experimente können dabei unter zwei verschiedenen Bedingungen durchgeführt werden:

1. **In-Vivo**: Fallstudie in realer Umgebung. Ein Beispiel sind die Untersuchungen von Clough/Gaizauskas/Piao (2002) zum Text-Reuse in der britischen Presse.

2. **In-Vitro**: in kontrollierter Umgebung.

Eine methodische, kontrollierte, quantifizierbare, reproduzierbare und randomisierte Durchführung der Experimente ermöglicht es dabei, mit Hilfe von Messungen neue Phänomene zu identifizieren, Hypothesen zu testen oder neue Erkenntnisse in Bezug auf die Anwendung von Modellen und Methoden zu gewinnen.

Grundlage für In-Vitro Experimente ist eine Evaluierungsumgebung, bestehend aus einer Kollektion von Plagiatbeispielen, sowie geeigneten Maßen

zur Bewertung der Leistung der Algorithmen. Da bisher keine einheitliche Evaluierungsumgebung für Plagiaterkennungsverfahren existiert, sind Wissenschaftler bis heute darauf angewiesen, eigene Experimente und Methoden zu entwickeln, um ihre Forschungsergebnisse zu evaluieren. So ergab eine Auswertung von 104 Veröffentlichungen zur automatischen Plagiaterkennung, dass in 80% der Fälle ein eigener Korpus angelegt wurde und lediglich 35% ihren Ansatz mit anderen Algorithmen verglichen (Potthast et al., 2010b). Folglich sind die meisten Ergebnisse nicht vergleichbar, was für empirische Forschung allerdings von grundlegender Bedeutung ist. Dies kann unter anderem zu einem von Armstrong et al. (2009) beschriebenen Stagnationseffekt führen. Demnach kann ein Fortschritt in der Wissenschaft dadurch verhindert werden, dass man Algorithmen im Vergleich zu anderen Ansätzen nicht sinnvoll bewertet. In der Tat erscheint nur ein Vergleich der Forschungergebnisse mit aktuellen *State of the Art*-Ansätzen von Bedeutung.

Das Ziel dieser Arbeit besteht daher in der Entwicklung einer einheitlichen Evaluierungsumgebung, welche es erlaubt, Verfahren zur Plagiaterkennung hinsichtlich ihrer Erkennungsleistung zu bewerten und miteinander zu vergleichen. Dafür wird zunächst ein Korpus in Form einer Textsammlung erstellt, welcher eine repräsentative Menge an Plagiaten unterschiedlicher Formen enthält. Die Plagiate werden dabei sowohl automatisch generiert, als auch teilweise von Menschen simuliert. Weiterhin wird ein Evaluierungsmaß basierend auf dem Konzept von *Precision* und *Recall* entwickelt, um die Erkennungsleistung von Algorithmen zur Plagiatanalyse zu quantifizieren.

In der Arbeit werden deshalb in Kapitel 2 in Vorbereitung auf die Korpuskonstruktion die möglichen Formen von Plagiaten und bekannten Plagiaterkennungsalgorithmen vorgestellt. Hierauf aufbauend wird in Kapitel 3 die Auswahl der in den Korpus einfließenden Plagiatformen vorgestellt und begründet. Weiterhin werden die Struktur des Korpus, sowie Verfahren und Ansätze zur Erzeugung künstlicher Plagiate und ein Konzept zur kostengünstigeren manuellen Erstellung von Plagiaten mittels *Crowdsourcing* entwickelt. Als zweiter grundlegender Teil der Evaluierungsumgebung wird in Kapitel 4 ein neues Evaluierungsmaß zur Messung der Leistung von Plagiaterkennungsalgorithmen vorgestellt. Unter Anwendung dieser Maße, werden in Kapitel 5 die durch die Teilnehmer der „*2nd International Competition on Plagiarism Detection*“ erzielten Ergebnisse vorgestellt und ausgewertet. Abschließend wird die Arbeit in Kapitel 6 nocheinmal zusammengefasst, die im Laufe des Projekts gesammelten Erfahrungen ausgewertet und ein Ausblick auf mögliche weitere Entwicklungen der Evaluierungsumgebung gegeben.

Kapitel 2

Plagiate und Plagiaterkennung

Für eine aussagekräftige Bewertung der Leistung eines Algorithmus mit Hilfe von Experimenten, ist ein strukturierter Aufbau und ein repräsentativer Inhalt der zugrundeliegenden Testkollektion entscheidend. In Bezug auf die Evaluierung von Algorithmen zur Plagiaterkennung bedeutet dies, dass die Auswahl der in dem Korpus vorkommenden Plagiate die in der Realität möglichen Plagiatformen weitestgehend abdecken sollte. Dementsprechend wird in diesem Kapitel eine umfassende Übersicht über mögliche Formen des Textplagiats und darauf aufbauend Methoden zu deren Erkennung vorgestellt. Diese Übersicht bildet dann die Grundlage für die Auswahl der in den Korpus einfließenden Plagiatformen.

2.1 Plagiatformen

Ein Plagiat wird als „das unrechtmäßige Nachahmen und Veröffentlichen eines von einem anderen geschaffenen künstlerischen oder wissenschaftlichen Werkes“ (Duden, 2003) definiert. Neben dieser allgemeinen Definition können Plagiate in weitere Kategorien bezüglich möglicher vorgenommener Modifikationen, semantischer Zusammenhänge zum Ursprungstext und der Intention des Autors eingeteilt werden. Eine erste grobe Einteilung kann zunächst mit Blick auf die Intention des Autors vorgenommen werden. Geht man davon aus, dass nicht jedes Plagiat wissentlich entsteht, ergeben sich folgende Kategorien:

- **versehentliches Plagiat**: kann auf Grund fehlenden Wissens über Plagiate bzw. korrekte Zitierrichtlinien entstehen
- **unbeabsichtigtes Plagiat**: kann entstehen, wenn ein Autor durch Ideen und Wissen aus Quellen beeinflusst wurde und er diese Gedanken unbewusst so wiedergibt, dass sie als von ihm erdacht erscheinen

- **beabsichtigtes Plagiat**: beschreibt eine vom Autor bewusste vollständige oder teilweise Übernahme eines fremden Textes ohne angemessene Referenzierung des Ursprungstextes

Die Unterscheidung dieser drei Kategorien ist, bis auf die Erkennung fehlerhafter Zitate, für die Plagiaterkennung nicht von Bedeutung, da eine entsprechende Klassifizierung aufgrund mangelnden Wissens über mögliche Umstände und Intentionen des Autors nicht realisierbar ist.

Bezüglich möglicher vorgenommener Änderungen des Ursprungstextes, können desweiteren folgende Plagiatformen unterschieden werden:

- **Wort-für-Wort-Plagiat**: Teilen oder ein gesamter veröffentlichter Text wurde direkt übernommen.
- **Paraphrasenplagiat**: Bei dieser Form des Plagiats wurde der Ursprungstext teilweise umgeschrieben/paraphrasiert.

In Bezug auf semantische Bezüge zwischen Ursprungstext und Plagiat können weitere Unterscheidungen vorgenommen werden:

- **Übersetzungsplagiat**: Verwendung findet hier eine Übersetzung eines fremdsprachigen Textes.
- **Strukturplagiat**: Die Reihenfolge von Argumenten oder Gedanken wird aus einem fremden Werk übernommen und im eigenen Text mit eigenen Worten wiedergegeben.
- **Ideenplagiat**: Beschreibt die Wiederverwendung eines Gedankens aus einer Quelle ohne dabei den genauen Wortlaut oder die Form der Quelle zu übernehmen.

Eine Sonderstellung nehmen die beiden folgenden Plagiatformen ein:

- **Autorenschaftsplagiat**: Ein fremdes Werk wird vollständig und unter Angabe des eigenen Namens publiziert (Beispiel: Ghostwriting).
- **Selbstplagiat**: Beschreibt die Verwendung von Texten bereits selbst publizierter Arbeiten in Teilen oder im Ganzen.

2.2 Plagiaterkennung

Das grundlegende Ziel von Plagiaterkennungsalgorithmen ist es, in einem verdächtigen Dokument d_{plg} alle Textpassagen s_{plg} zu finden, deren Inhalt möglicherweise vom Textabschnitt s_{src} im Dokument d_{src} plagiiert wurde.

Entsprechende Algorithmen können zunächst dahingehend unterschieden werden, ob sie ein Dokument lokal oder global analysieren. Bei einer lokalen Analyse wird d_{plg} auf plagiierte Abschnitte s_{plg} untersucht, welche wiederum wenn möglich einer Textpassage s_{src} in einem Quelldokument d_{src} zugeordnet werden. Bei einer globalen Betrachtung, werden d_{plg} und d_{src} als Ganzes miteinander verglichen.

Bezüglich der Untersuchungsmethodik lassen sich im Wesentlichen zwei Ansätze unterscheiden. Steht eine Kollektion von möglichen Quelldokumenten für die zu untersuchenden Plagiate zur Verfügung, kommen externe Verfahren zum Einsatz. Steht diese Kollektion hingegen nicht zur Verfügung, finden sogenannte intrinsische Plagiaterkennungsalgorithmen Verwendung.

Eine Übersicht über die unterschiedlichen Plagiatvergehen und die entsprechenden Erkennungsmethoden wird in Abbildung 2.1 dargestellt.

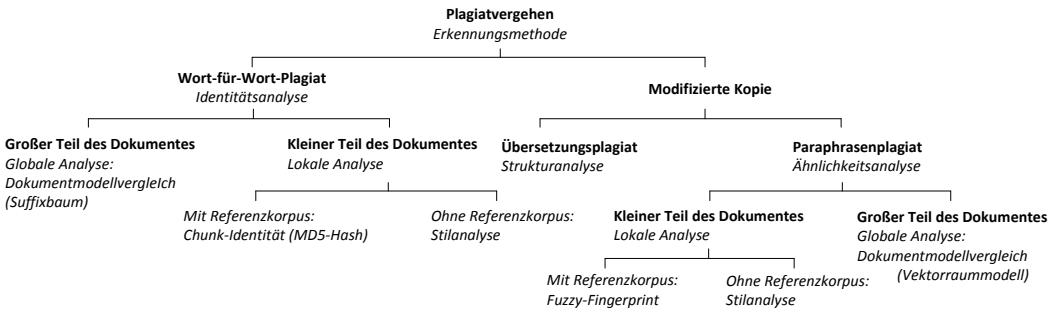


Abbildung 2.1: Taxonomie von Plagiatvergehen in Verbindung mit entsprechenden Erkennungsmethoden (nach Meyer Zu Eissen/Stein/Kulig (2007))

2.2.1 Externe Verfahren

Verfahren zur externen Plagiaterkennung untersuchen ein Dokument d_{plg} auf mögliche plagiierte Textpassagen s_{plg}. Ein Plagiat lässt sich hierbei als Quadrupel $s = \langle s_{plg}, d_{plg}, s_{src}, d_{src} \rangle$, bestehend aus der plagiierten Passage s_{plg} in d_{plg} und deren Quellpassage s_{src} in d_{src}, definieren. Grundlage für externe Plagiaterkennungsalgorithmen ist ein Referenzkorpus D_{ref}, welcher d_{src} und somit auch s_{src} enthalten kann.

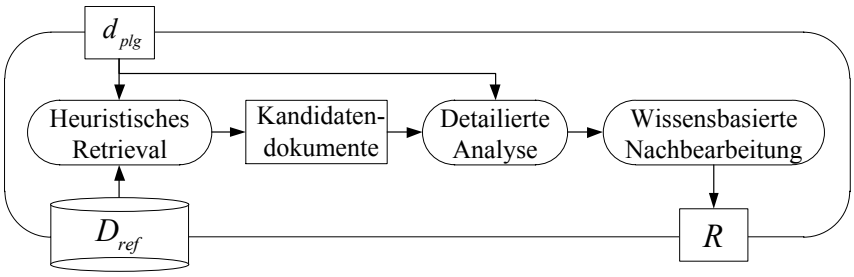


Abbildung 2.2: Schematische Darstellung des Prozesses der externen Plagiatanalyse (für die Bedeutung der Variablen, siehe Text) (nach Potthast et al. (2009))

Der Prozess der externen Plagiaterkennung kann, wie in Abbildung 2.2 dargestellt, in der Regel in drei Teile aufgeteilt werden (Stein/Meyer Zu Eissen/Potthast, 2007):

1. **Heuristisches Retrieval**: Der Referenzkorpus D_{ref} wird nach Textpassagen aus d_{plg} durchsucht. Das Ergebnis dieses Schrittes ist eine Menge von Kandidatendokumenten D'_{src}, welche möglicherweise Textpassagen aus d_{plg} enthalten. Üblicherweise sollte gelten, dass $|D'_{src}| \ll |D_{ref}|$, um den Aufwand für die folgenden Schritte zu minimieren.

2. **Detaillierte Analyse**: In diesem Schritt wird d_{plg} detailliert mit jedem möglichen Quelldokument $d'_{src} \in D'_{src}$ verglichen. Zwei Textpassagen r_{plg} und r_{src}, die eine hohe Ähnlichkeit aufweisen, werden als Plagiaterkennungsfall $r = \langle r_{plg}, d_{plg}, r_{src}, d'_{src} \rangle$ annotiert. Dabei steht r_{plg} für den von dem Algorithmus erkannten plagiierten Textabschnitt und r_{src} für dessen vom Algorithmus erkannten Ursprungstext. Idealerweise gilt $s_{plg} = r_{plg}$, $s_{src} = r_{src}$ und $d'_{src} = d_{src}$. Eine hohe Ähnlichkeit zwischen s_{plg} und s_{src} erhöht den Plagiatverdacht.

3. **Wissensbasierte Nachbearbeitung**: Um falsch positive Ergebnisse zu vermeiden, wird für alle $r \in R$ geprüft, ob diese mit ordnungsgemäßen Angaben wie Quellennachweis oder Zitatangaben versehen sind.

2.2.2 Intrinsische Verfahren

Unter der Voraussetzung, dass ein Dokument zum überwiegenden Teil von einem Autor geschrieben wurde, ist es mithilfe von Algorithmen zur intrinsischen Plagiaterkennung möglich, auch Plagiate zu erkennen, für die kein Quelldokument gefunden werden kann. Dies ist zum Beispiel relevant, wenn ein Autor aus Büchern plagiiert, die nicht in digitaler Textform zur Verfügung stehen.

Dementsprechend findet bei der intrinsischen Plagiaterkennung auch kein Vergleich von d_{plg} mit anderen Dokumenten statt. Vielmehr überprüfen intrinsische Verfahren, ob ein Textabschnitt aus d_{plg} vom selben Autor geschrieben wurde wie der überwiegende Rest des Dokumentes. Man spricht in diesem Fall von einem Ein-Klassen-Klassifizierungsproblem, bei welchem der Autor des Textes die Zielklasse darstellt, für die wiederum eine bestimmte Anzahl von Textbeispielen in d_{plg} vorkommt. Als Grundlage für die Klassifizierung dienen hierbei Merkmale, die mit Hilfe von Verfahren der Stilanalyse ermittelt werden (Stein/Lipka/Prettenhofer, 2010). Diese basieren nach Meyer Zu Eissen/Stein/Kulig (2007) auf folgenden semiotischen Charakteristika:

1. **Textstatistiken auf Zeichenebene**: Beispiele hierfür sind die Anzahl der Kommata, die Anzahl der Fragezeichen und die Wortlänge.

2. **Syntaktische Merkmale auf Satzebene**: Untersucht werden z.B. die Satzlänge und Statistiken über die Benutzung von Funktionswörtern.

3. **Wortart-Merkmale** zur Quantifizierung der Benutzung von Wortklassen: Auszugsweise ist die Anzahl der Adjektive und die der Pronomen zu nennen.

4. **Mengen geschlossener Wortklassen** um spezielle Wörter zu zählen: Beispiele hierfür sind die Anzahl der Stoppwörter und die der Fremdwörter.

5. **Strukturelle Merkmale**, die die Textorganisation beschreiben: Zu nennen sind hier beispielsweise Paragraph- und Kapitellänge.

Das Ziel ist es, in der Menge der Merkmalsausprägungen „Ausreißer“ zu finden, die nicht der Klasse des Autors angehören. Stellen die Merkmalsausprägungen eines Textfragments s_{plg} einen derartige Ausreißer dar, so wurde s_{plg} mit hoher Wahrscheinlichkeit nicht vom selben Autor verfasst, wie der Rest des Dokuments. Dies wiederum würde bedeuten, dass es sich bei s_{plg} um ein Plagiat handelt.

Kapitel 3

Korpuskonstruktion

Die Basis für den Korpus bilden Textdokumente verschiedener Länge, Autoren und Inhalte, welche sich textuell nicht überlappen. Mit dem Ziel die Leistung von Plagiaterkennungsalgorithmen zu testen, werden in diese Dokumente Plagiate unterschiedlicher Form und Ausprägung an zufälligen Stellen eingefügt. In Vorbereitung auf die eigentliche Konstruktion werden in Abschnitt 3.1 zunächst die zu verwendenden Plagiatformen, in Abschnitt 3.2 Strategien zu deren Erstellung und in Abschnitt 3.5 die Annotation der Plagiate im Korpus vorgestellt. Anschließend werden in Abschnitt 3.4 und Abschnitt 3.3 der Aufbau des Korpus sowie detailiert Eckdaten wie Dokument- und Plagiatlängen oder der prozentuale Anteil von Plagiaten in einem Dokument festgelegt. In Abschnitt 3.6 wird abschließend der Konstruktionsprozess in seinen einzelnen Stufen veranschaulicht.

3.1 Verwendete Plagiatformen

Für einen Korpus zur Evaluierung von Plagiaterkennungsalgorithmen ist die gezielte Auswahl der in ihm vorkommenden Plagiatformen von grundlegender Bedeutung, da sie die Aussagekraft der Ergebnisse in Bezug auf reale Szenarien bestimmt.

Die Schwierigkeit besteht zunächst darin, sich adäquate Quellen für große Mengen verschiedener Plagiatformen zu erschließen. Hierbei bieten sich drei Möglichkeiten an:

1. **Echte Plagiate**: Verwendung bereits bekannt gewordene Plagiate z.B. in studentischen Arbeiten, Büchern, Blogs, etc.

2. **Simulierte Plagiate**: Erstellung von Plagiaten durch Personen unter einer bestimmten Aufgabenstellung und ausschließlich zur Verwendung in dem geplanten Korpus.

3. **Künstliche Plagiate**: Erstellung von Plagiaten mit Hilfe von Algorithmen.

Jede dieser Möglichkeiten ist jedoch mit Einschränkungen oder Problemen verbunden. In Tabelle 3.1 (Seite 10) werden diese beschrieben und mögliche Lösungsansätze vorgestellt. Aufgrund der Probleme, die mit der Nutzung echter Plagiate verbunden wären, fließen in den Korpus nur simulierte und künstliche Plagiate ein.

Die zweite grundlegende Unterscheidung verschiedener Plagiatformen gründet sich auf die bereits in Abschnitt 2.1 vorgestellte Übersicht. Tabelle 3.2 greift diese auf und ordnet sie möglichen Formen der Erstellung (künstlich/simuliert) zu.

Eine weitere Unterscheidung von Plagiatformen kann im Hinblick auf deren Erkennung gemacht werden. Soll ein Plagiat nur mit Hilfe von intrinsischen Verfahren erkannt werden, so darf seine Quelle nicht im Korpus vorhanden sein. Soll ein Plagiat hingegen mit Verfahren der externen Plagiaterkennung gefunden werden, ist das Vorhandensein der Quelle im Korpus zwingend notwendig.

3.2 Strategien zur Erstellung von Plagiaten

Bei der Erstellung von Plagiaten wird eine Kopie eines Textabschnittes s_{src} aus einem Dokument d_{src} entnommen und an einer zufälligen Stelle in ein zufällig ausgewähltes Dokument d_{plg} eingefügt. Textabschnitt s_{src} wird dabei so gewählt, dass er mit einem Satzanfang beginnt und mit einem Satzende aufhört, um auffällige Textanomalien zu vermeiden. Wird s_{plg} in d_{plg} eingefügt, ersetzt es dort einen etwa gleichgroßen Textabschnitt s_{rpl} (siehe Abbildung 3.1). Dies ist nötig, da ohne ein Ersetzen von s_{rpl} Dokumente mit steigendem Plagiatanteil exponentiell an Größe zunehmen würden.

Bei der Erstellung von Plagiaten, welche ausschließlich von intrinsischen Verfahren erkannt werden, muss außerdem gelten, dass d_{plg} und d_{src} von zwei unterschiedlichen Autoren geschrieben wurde. Dies ist notwendig, da derartige Verfahren fremde Textabschnitte nur anhand von Unterschieden in der Stilistik erkennen können (siehe Abschnitt 2.2.2), welche bei zwei Texten des selben Autors mit hoher Wahrscheinlichkeit nicht auftreten[1]. Da es sich bei der Erkennung von intrinsischen Plagiaten nicht um eine Retrieval-Aufgabe handelt und somit auch kein Vergleich von Textabschnitten stattfindet, erscheint eine zusätzliche Veränderung des Plagiattextes s_{plg} nicht sinnvoll. Die dadurch möglicherweise erreichte höhere/verringerte Abweichung bestimmter

[1] Der Schreibstil eines Autors ändert sich in der Regel nicht innerhalb einer Schaffensperiode, sondern nur über längere Zeiträume.

	Problem	Lösungsansatz
Echte Plagiate	Die Verwendung und Weiterveröffentlichung realer, bekannt gewordener Plagiate ist sowohl rechtlich, als auch moralisch umstritten und bedarf sowohl der Zustimmung des entsprechenden Autors sowie des Plagiators. Diese Zustimmung ist jedoch auf Grund der Einordnung des Plagiats als Verstoß gegen ethische und wissenschaftliche Grundsätze eher unwahrscheinlich. Die gewünschte hohe Anzahl und Vielfalt an Plagiatformen erschwert eine Beschaffung zusätzlich.	*Simulierte Plagiate* Das Problem lässt sich nicht umgehen und würde die Erstellung eines großformatigen Korpus stark erschweren und verlangsamen.
Simulierte Plagiate	Die Simulierung von Plagiaten in Form einer gezielten Erstellung per Hand ist sehr zeit- und kostenintensiv. Außerdem ist es schwierig, ein entsprechend breites Spektrum an Plagiatformen sicherzustellen. Hierfür wäre eine große Anzahl von Autoren notwendig.	*Crowdsourcing* Aufgrund von Preisen, die pro Auftrag im Cent-Bereich liegen, erlaubt es der *Crowdsourcing*-Dienst *Amazon Mechanical Turk*, tausende Plagiatfälle durch ein Netzwerk von Personen und ohne den üblichen Kostenaufwand, zu erstellen.
Künstliche Plagiate	Für die Erstellung künstlicher Plagiate fehlen Untersuchungen, die Aufschluss darüber geben, wie oft und in welcher Ausprägung welche Plagiatformen in der Realität vorkommen. Auch sind Computer bisher nicht in der Lage sind, Texte vollständig semantisch zu erfassen, um sie beispielsweise umzuschreiben, wie es bei Plagiaten häufig Anwendung findet.	*Diversifikation & Annäherung* Anstatt reale Umstände nachzubilden, wird versucht, ein breites Spektrum an Möglichkeiten abzudecken. Mögliche Textmodifikationen werden mit Hilfe von Verfahren des *Information Retrievals* und des *Natural Language Processings* so realistisch wie möglich nachgebildet.

Tabelle 3.1: Probleme und entsprechende Lösungsansätze bei der Verwendung der unterschiedlichen Plagiatformen (siehe dazu auch Abschnitt 3.2).

Plagiatform		**Bemerkung**
versehentliches Plagiat	$\{k, s\}^*$	aus der Sicht eines Computers keine Unterscheidung möglich; mögliche Unterklasse aller anderen Plagiatformen
unbeabsichtigtes Plagiat	$\{k, s\}^*$	
beabsichtigtes Plagiat	$\{k, s\}^*$	
Wort-für-Wort-Plagiat	$\{k\}$	
Paraphrasenplagiat	$\{k, s\}$	
Übersetzungsplagiat	$\{k\}$	
Strukturplagiat	$\{s\}$	künstliche Erstellung derzeit noch nicht möglich; teilweises Vorkommen jedoch bei simulierten Plagiaten
Ideenplagiat	$\{s\}$	
Autorenschaftsplagiat		Diese Form des Plagiats ist nicht vorgesehen (Forschungsgebiet *Authorship Attribution*)
Selbstplagiat	$\{k, s\}^*$	Aus Algorithmensicht eine mögliche Unterklasse jeder anderen Plagiatform

Tabelle 3.2: Liste der verschiedenen Plagiatformen, welche für den Korpus künstlich (k) und/oder simuliert (s) erstellt wurden. Eine Markierung durch einen Stern bedeutet, dass diese Plagiatform nicht explizit erstellt wurde, sondern in ihrem Vorkommen eine Unterklasse einer anderen Plagiatform bildet.

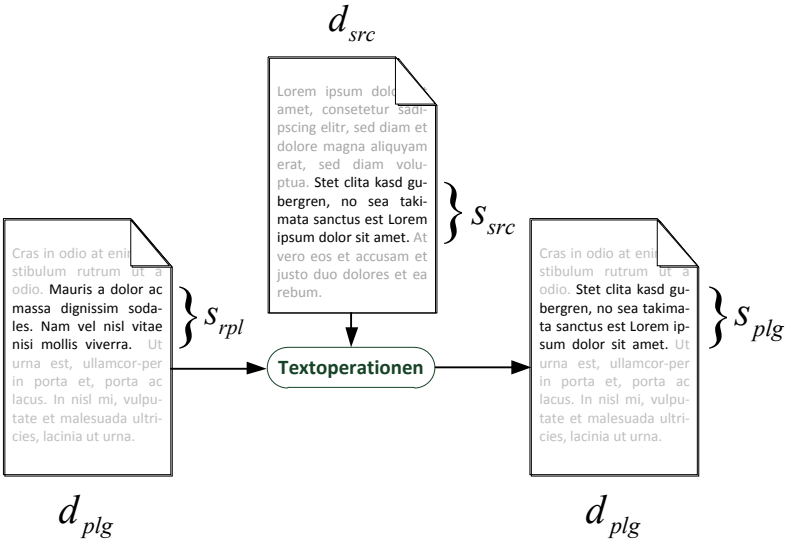


Abbildung 3.1: Schematische Darstellung des Vorgangs der Plagiaterstellung. Abschnitt s_{rpl} in d_{plg} wird durch s_{plg} ersetzt. s_{plg} wird mit Hilfe von Textoperationen von s_{src} abgeleitet, d.h. s_{plg} ist eine Übersetzung, Umformulierung oder Wort-für-Wort-Kopie von s_{src}.

Stilmerkmale von s_{plg}, in Bezug auf den restlichen Text von d_{plg}, steht mit dem notwendigen Aufwand in keinem Verhältnis zum Nutzen. Aufgrund der großen Anzahl von Autoren ist davon auszugehen, dass auch bei einer zufälligen Kombination von Autoren ein breites Spektrum an möglichen Merkmalsdifferenzen entsteht.

Bei Plagiaten, die von externen Plagiaterkennungsalgorithmen erkannt werden sollen, ist abhängig von der entsprechenden Plagiatform eine Modifikation von s_{plg} vor dem Einfügen in d_{plg} notwendig. Dadurch soll das Verhalten eines Plagiators (Person die plagiiert) simuliert werden. In Abschnitt 3.2.1 wird ein Ansatz vorgestellt, diese Modifikation mit Hilfe von Algorithmen durchzuführen. Der Vorteil liegt hierbei darin, dass große Mengen an Plagiaten nahezu ohne Zeit- und Kostenaufwand erstellt werden können. Abschnitt 3.2.2 stellt in Ergänzung dazu einen Ansatz zur Simulierung von Plagiaten durch Menschen vor. Der Vorteil ist hierbei eine wesentlich realistischere Modifikation der Texte.

3.2.1 Künstliche Plagiate

Wie bereits in Tabelle 3.2 dargestellt, sollen im Rahmen der Korpuskonstruktion Wort-für-Wort-, Übersetzungs- und Paraphrasenplagiate künstlich erzeugt werden. Während bei einem Wort-für-Wort-Plagiat s_{plg} lediglich eine unverän-

derte Kopie von s_{src} darstellt, erfordert die maschinelle Erstellung von Übersetzungen und Paraphrasierungen eine nicht triviale Umformung von s_{plg}. Das Forschungsgebiet, welches sich mit diesen Themen beschäftigt, nennt sich *Natural Language Processing* (NLP).

3.2.1.1 Übersetzungsplagiate

Für die maschinelle Übersetzung von Texten werden heutzutage vor allem statistische Modelle und hybride Systeme aus statistischen Modellen und regelbasierten Übersetzungssystemen verwendet (siehe hierzu weiterführend „Learning machine translation“ von Goutte/Cancedda/Dymetman (2009)). Aus einem Wettbewerb des *National Institute of Standards and Technology* (NIST) ging 2005 Googles statistisches Übersetzungssystem als das beste System für maschinelle Übersetzungen hervor. Messungen auf Basis des BLEU-Scores (Papineni et al., 2002), welcher die Ähnlichkeit zwischen einer maschinellen Übersetzung und einer durch Menschen angefertigten Referenzübersetzung des selben Textes misst zeigen, dass *Google Translate* bis heute führend ist (Amancio et al., 2008). Aufgrund der relativ hohen Ähnlichkeit der Übersetzungen im Vergleich zu menschlichen Übersetzungen, des hohen Bekanntheitsgrades des Dienstes und der damit verbundenen häufigen Nutzung (auch durch Plagiatoren), sowie der einfachen automatisierten Nutzung des Service über eine API (Google, 2010), wird der Dienst zur Übersetzung von Texten und somit zur Erstellung von Übersetzungsplagiaten verwendet. Tabelle 3.3 zeigt ein Beispiel hierfür.

Ursprungstext s_{src}	**Plagiattext** s_{plg}
Mein Vater war ein Mensch mit zwei Seelen. Die eine Seele unendlich weich, die andere tyrannisch, voll Uebermaß im Zorn, unfähig, sich zu beherrschen. Er besaß hervorragende Talente, die aber alle unentwickelt geblieben waren, der großen Armut wegen.	My father was a man with two souls. One soul infinitely soft, the other tyrannical, fully incapable of excess in anger to control themselves. He possessed great talent, but they were all still immature, the extreme poverty due.

Tabelle 3.3: Beispiel eines Übersetzungsplagiats. Der Originaltext stammt aus dem Buch „Mein Leben Und Streben“ von Karl May und wurde mit Hilfe von *Google Translate* ins Englische übersetzt.

3.2.1.2 Paraphrasenplagiate

Bei Paraphrasenplagiaten besteht das Ziel darin, den Text s_{plg} so zu verändern, dass er semantisch identisch zu s_{src} bleibt, während Syntax und Vokabular verändert werden. In der Praxis handelt es sich dabei in der Regel um Umformulierungen beziehungsweise Anpassungen des Textes, mit dem Ziel ein Plagiat zu verschleiern und eine Erkennung zu erschweren. Das folgende Beispiel veranschaulicht das Vorgehen:

> „Karl May entstammte einer armen Weberfamilie.“
>
> „Die Eltern von Karl May waren arme Weber.“
>
> „Karl May wurde in eine recht arme Weberfamilie hineingeboren.“

Alle drei Sätze sagen das Selbe aus, weichen jedoch syntaktisch und in Bezug auf ihr Vokabular voneinander ab. Neben der Erkennung solcher Paraphrasen beschäftigen sich Wissenschaftler im Bereich des NLP auch mit deren automatischer Erstellung. Die aktuelle Forschung auf dem Gebiet beschäftigt sich dabei vor allem mit statistischen Modellen, ähnlich derer, die zur maschinellen Übersetzung verwendet werden. Einen guten Überblick über entsprechende Ansätze liefern die Arbeiten von Androutsopoulos/Malakasiotis (2010) und Madnani/Dorr (2010).

Aufgrund der Komplexität dieser Ansätze sowie des fehlenden Zugriffs auf entsprechende, zum Training dieser Modelle notwendigen Korpora, wurde für die Erzeugung der Paraphrasenplagiate ein einfacherer Ansatz gewählt. Ziel hierbei ist es, das Verhalten eines Plagiators beim Verschleiern eines Plagiats durch die folgenden einfachen Textoperationen zu imitieren:

- Einfügen/Löschen/Ersetzen von Wörtern
- Vertauschen von Wörtern
- Vertauschen von Phrasen
- Vertauschen von Sätzen

Als Maß für den Grad der Veränderung von s_{plg} in Bezug auf s_{src} wird die von Kondrak (2005) vorgestellte n-Gramm-Distanz verwendet. Sie hat einen Wertebereich von $[0, 1]$ und ist so definiert, dass ein Wert von 0 keine Veränderung des Textes bedeutet. Ein Wert von 1 hingegen bedeutet, dass beide Texte keine Gemeinsamkeiten hinsichtlich des verwendeten Vokabulars sowie der Wortkombinationen haben. Nicht jede der Textoperationen ist dabei in der Lage, einen Wert von 1 zu erreichen, da besonders die Veränderung auf Phrasen- und Satzebene weniger n-Gramme verändern können, als solche auf Wortebene.

Um die Veränderung einzelner Wörter nicht überzugewichten und die Gewichtung von Änderungen auf Satz- und Phrasenebene zu erhöhen, werden bei der Berechnung der Distanz n-Gramme für $n = 2, ..., 5$ berücksichtigt.

Um starke Textanomalien durch eine willkürliche Anwendung der Textoperationen zu vermeiden, wird bei Operationen auf Wort- und Phrasenlevel eine Heuristik auf Basis von *Part-of-Speech*-Tags (PoS) angewendet. Beim sogenannten *Part-of-Speech Tagging* werden die Wörter eines Textes dafür zunächst ihrer entsprechenden Wortart (englisch: *part-of-speech*) zugeordnet. Das Beispiel in Tabelle 3.4 veranschaulicht dies. Für die automatische Zuordnung die-

PoS-Tags:	*NN*	*VBG*	*BEZ*	*DT*	*JJ*	*NN*	*IN*	
Text:	Content	scraping	is	a	known	phenomenon	of	...

Tabelle 3.4: Ein Textausschnitt mit den zugehörigen Wortarten (part-of-speech); Die PoS-Tags basieren auf dem *Penn Treebank Tag Set* (Marcus/Marcinkiewicz/Santorini, 1993). *NN* = Substantiv; *VBG* = Verb; *BEZ* = „is“; *DT* = Determinativ; *JJ* = Adjektiv; *IN* = Präposition.

ser Information wird ein auf statistischen Verfahren basierender PoS-Tagger verwendet (Mason, 1997). Die Grundlage für statistisches PoS-Tagging bilden große Korpora, in denen die Wörter bereits mit PoS-Tags versehen wurden. Ein mithilfe der Auftrittswahrscheinlichkeit jeder Kombination von „Wort - PoS-Tag“ sowie mit der Auftrittswahrscheinlichkeit aller möglichen n-Gramme von PoS-Tags trainiertes *Hidden Markov Models* (HMM) kann somit für jeden Satz die wahrscheinlichste Zuordnung von PoS-Tags berechnen.

Soll nun ein Wort in s_{plg} einfügt werden, so kann mit Hilfe der in der Datenbank des PoS-Taggers gespeicherten Auftrittswahrscheinlichkeit der PoS-Tag n-Gramme die wahrscheinlichste Position berechnet werden. Möchte man beispielsweise das Wort „illegal“ (PoS-Tag: *JJ*) in das Beispiel aus Tabelle 3.4 einfügen, so wird zunächst die Auftrittswahrscheinlichkeit für jedes möglicherweise entstehende Trigramm von PoS-Tags berechnet (siehe Tabelle 3.5): Wählt man die wahrscheinlichste Kombination („*JJ* ***JJ*** *NN*“), so ergibt sich folgender Text:

„Content scraping is a known **illegal** phenomenon of ...“

Als Quelle für einzufügende Wörter dient hierbei der durch s_{plg} in d_{plg} ersetzte Text s_{rpl}. Aus ihm kann zu diesem Zweck ein zufälliges Wort ausgewählt werden. Somit wird erreicht, dass das Vokabular von s_{plg} stückweise an den Umgebungstext angepasst werden kann.

Auf eine äquivalente Weise lassen sich auch Wörter löschen. Dafür wird zunächst die Wahrscheinlichkeit des Vorkommens aller möglichen PoS-Trigramme,

tri_{PoS}	$p(tri_{PoS})$	tri_{PoS}	$p(tri_{PoS})$
JJ *NN VBG*	0,001	*DT* ***JJ*** *JJ*	0,073
NN ***JJ*** *VBG*	0,002	*JJ* ***JJ*** *NN*	0,099
VBG ***JJ*** *BEZ*	0,010	*NN* ***JJ*** *IN*	0,000
BEZ ***JJ*** *DT*	0,006	*NN IN* ***JJ***	0,000

Tabelle 3.5: Die Wahrscheinlichkeit des Vorkommens aller möglichen PoS-Tag-Trigramme, die durch Einfügen von *JJ* („illegal“) entstehen können.

die durch Löschen eines Wortes entstehen können, berechnet (siehe Tabelle 3.6). Wird das Wort gelöscht, dessen Verschwinden das wahrscheinlichste

tri_{PoS}	$p(tri_{PoS})$	tri_{PoS}	$p(tri_{PoS})$
. VBG BEZ	0,081	*BEZ JJ NN*	0,007
NN BEZ DT	0,075	*DT NN IN*	0,120
VBG DT JJ	0,001		

Tabelle 3.6: Die Wahrscheinlichkeit des Vorkommens aller möglichen PoS-Tag-Trigramme, die durch Löschen eines Wortes entstehen können. Der Punkt muss am Beginn des Textes stehen, um auch die Wahrscheinlichkeit eines alternativen Satzbeginns berechnen zu können.

PoS-Trigramm erzeugt („known“; PoS-Tag: *JJ*), ergibt sich folgender Text:

„Content scraping is a phenomenon of ...“

Für das Ersetzten von Wörtern hingegen existieren zwei Ansätze. Zum einen kann s_{rpl} ebenfalls auf seine Wortarten hin analysiert und anschließend ein Wort der Klasse X aus s_{plg} mit einem Wort der selben Klasse (X) aus s_{rpl} ersetzt werden. Dieser Prozess gleicht das Plagiat bezüglich seines Vokabulars an den Umgebungstext in d_{plg} an. Ein Beispiel für einen Text, der durch die bisher beschriebenen Textoperationen des Einfügens/Löschens/Ersetztens von Wörtern modifiziert wurde, ist in Tabelle 3.7 zu sehen.

Eine weitere mögliche Quelle beim Ersetzten von Wörtern stellen mit diesen in einer semantischen Beziehung stehende Wörter dar. Hierzu zählen Synonyme (semantische Äquivalente), Hyponyme (Unterbegriffe), Hyperonyme (Oberbegriffe) und Antonyme (Gegenwörter). Als Quelle für diese Information dient das *WordNet* (Miller, 1995) - eine Datenbank, welche die semantische und lexikalische Beziehung zwischen Wörtern der englischen Sprache enthält. Ein Beispiel für einen Text, in dem alle Wörter, sofern vorhanden, durch semantisch verwandte Wörter ersetzt wurden, findet sich in Tabelle 3.8.

Ursprungstext s_{src}	**Ersetzter Text** s_{rpl}	**Plagiattext** s_{plg}
Content scraping is a phenomenon of copy and pasting material from Internet *websites*, affecting both established sites and blogs.	The goal of WordNet was to develop a system that would be consistent with the knowledge acquired over the years about how human beings process language.	*Knowledge* scraping is a *system* of copy and *develop* pasting material *with* Internet, affecting both established *beings* and blogs.

Tabelle 3.7: Beispiel für das Ergebnis des automatisierten Löschens/Einfügens/Ersetztens von Wörtern. Kursiv gedruckte Wörter in s_{src} wurden gelöscht. Kursiv gedruckte Wörter in s_{plg} wurden entweder ersetzt oder eingefügt.

Ursprungstext s_{src}	**Plagiattext** s_{plg}
Content scraping is a phenomenon of copy and pasting material from Internet websites, affecting both established sites and blogs.	*Contented* scraping is a *process* of *imitate* and pasting *substance* from *Cyberspace chatroom*, *impact* both *found tract* and *diary*.

Tabelle 3.8: Beispiel für das Ergebnis des automatisierten Ersetzens von Wörtern durch semantisch verwandte Wörter unter Verwendung des *WordNets*. Kursiv gedruckte Wörter wurden im Vergleich zu s_{src} verändert.

Das Vertauschen von Wörtern im Text wird mit einer ähnlichen Heuristik realisiert wie das Ersetzten von Worten auf Basis der PoS-Tags. Hierbei werden die Worte einer Wortart untereinander getauscht. Ein Beispiel hierfür findet sich in Tabelle 3.9.

Ursprungstext s_{src}	**Plagiattext** s_{plg}
Content scraping is a phenomenon of copy and pasting material from Internet websites, affecting both established sites and blogs.	*Material* scraping is a phenomenon *from* copy and pasting *content of* Internet *blogs*, affecting both established sites and *websites*.

Tabelle 3.9: Beispiel für das Ergebnis des automatisierten Vertauschens von Wörtern unter Beibehaltung der syntaktischen Struktur. Kursiv gedruckte Wörter wurden im Vergleich zu s_{src} vertauscht.

Bestimmte Phrasen können auf einem ähnlichen Prinzip vertauscht werden. Dafür werden zwei Textabschnitte in s_{plg} gesucht, die jeweils mit einem identischen Paar von PoS-Tag-Trigrammen beginnen und mit einem ebenfalls identischen Paar enden und sich nicht überlappen. Diese zwei Textabschnitte können anschließend getauscht werden, ohne das hierbei in der Regel der Eindruck eines intakten Textes verloren geht. Hierfür befindet sich ein Beispiel in der nachfolgenden Tabelle 3.10.

Ursprungstext s_{src}	**Plagiattext** s_{plg}
Content scraping is a phenomenon of copy and pasting material from Internet websites, affecting both established sites and blogs. Free online tools are becoming available to help identify plagiarism, and there is a range of approaches that attempt to limit online copying.	*Free online tools are becoming available to help identify plagiarism, and there* is a phenomenon of copy and pasting material from Internet websites, affecting both established sites and blogs. *Content scraping is that attempt to approaches a range of* limit online copying.

Tabelle 3.10: Beispiel für das Ergebnis des automatisierten Vertauschens von Phrasen. Kursiv gedruckte Abschnitte wurden im Vergleich zu s_{src} vertauscht.

Um Sätze in s_{plg} zu vertauschen, wird der Text an seinen Satzenden aufgesplittet und die entsprechenden Teile neu zusammengesetzt.

Bis auf die Übersetzung von Texten, die nur zum Einsatz kommt, wenn d_{src} in einer anderen als der Hauptsprache des Korpus geschrieben wurde, werden alle bisher vorgestellten Methoden in Kombination auf s_{plg} angewendet. Dabei wird ein gewünschter Wert bezüglich der Eingangs erwähnten n-Gramm-Distanz vorgegeben und den einzelnen Verfahren ein zufälliger Anteil

zugewiesen. Anschließend werden die Textoperationen aufsteigend nach dem möglichen Einfluss auf die n-Gramm-Distanz ausgeführt:

1. Vertauschen von Sätzen
2. Vertauschen von Phrasen
3. Vertauschen von Wörtern
4. Einfügen/Löschen/Ersetzen von Wörtern

Dies hat zum Ziel, dass für den Fall, dass durch eine Stufe nicht der gewünschte Grad der Veränderung erreicht wird, die Differenz auf die restlichen Stufen verteilt wird.

3.2.2 Simulierte Plagiate

Ziel der hier vorgestellten Strategie ist es, Plagiate zu erstellen, die im Gegensatz zu den in Abschnitt 3.2.1 vorgestellten Textoperationen das menschliche Verhalten beim Verschleiern von Plagiaten, beziehungsweise der Umformulierung von Texten, wesentlich realistischer simulieren. Hierfür soll *Amazons Mechanical Turk* zum Einsatz kommen - ein Dienst, der in den letzten zwei Jahren einige Aufmerksamkeit auf sich gezogen hatte, da er es ermöglicht, Aufgaben, die von Computern nur schwer oder unzureichend gelöst werden können, zu sehr geringen Preisen durch Menschen lösen zu lassen.

Dies wird mithilfe von sogenanntem *Crowdsourcing* realisiert. Dabei können für den Menschen einfach zu lösende Aufgaben, wie z.B. die Bewertung der Relevanz eines Textes in Bezug auf ein Thema (TREC-Task) (Alonso/Mizzaro, 2009) oder dem Schreiben und Übersetzten von Texten (Ambati/Vogel/Carbonell, 2010), auf eine Masse von Freizeitarbeitern im Internet (der sogenannten *Crowd*) ausgelagert werden.

Über die Plattform von *Amazons Mechanical Turk* stellten wir den Nutzern 4000 Textabschnitte unterschiedlicher Länge mit der folgenden Aufgabenstellung zur Verfügung:

> „Rewrite the original text found below [*auf der entsprechenden Webseite*] so that the rewritten version has the same meaning as the original, but with a different wording and phrasing. Imagine a scholar copying a friend's homework just before class, or imagine a plagiarist willing to use the original text without proper citation.“

Voraussetzung für die Bearbeitung der Aufgabe war es, dass der Nutzer fließend Englisch sprechen kann. Außerdem wurde er darüber informiert, dass alle Ergebnisse geprüft werden würden. Insgesamt beteiligten sich 907 Personen an

Aufgabenstatistik				
Aufgaben pro User		**Arbeitszeit**	**Vergütung**	
Mittelwert	15	14 min	pro Aufgabe	0,06 - 0,5 US$
Standardabw.	20	21 min	abgelehnt	25%
Minimum	1	1 min		
Maximum	103	180 min		

Tabelle 3.11: Statistik der 4000 auf *Amazons Mechanical Turk* bearbeiteten Aufgaben

der Lösung dieser Aufgaben. Eine statistische Übersicht zur Lösung der Aufgaben durch die Teilnehmer wird in Tabelle 3.11 gegeben. Neben der eigentlichen Aufgabe konnten die Teilnehmer zusätzlich auf freiwilliger Basis Angaben zu ihrem Alter, Geschlecht, ihrer Ausbildung und Muttersprache machen. Des Weiteren wurden sie gefragt, ob sie professionelle Schreiber seien und ob sie jemals zuvor plagiiert hätten. Eine Auswertung der Umfrage ist in Tabelle 3.12 zu finden. Die Ergebnisse der Umfrage sind jedoch nur unter Vorbehalten zu

Teilnehmerstatistik					
Alter		**Geschlecht**		**Ausbildung**	
18, 19	10%	männlich	37%	High School	11%
20-29	37%	weiblich	39%	College	30%
30-39	16%			B.Sc.	17%
40-49	7%			M.Sc.	11%
50-59	4%			PhD.	2%
60-69	1%				
k.A.	25%	k.A.	24%	k.A.	29%
engl. Muttersprachler		**prof. Schreiber**		**bereits plagiiert**	
ja	62%	ja	10%	ja	16%
nein	14%	nein	66%	nein	60%
k.A.	23%	k.A.	24%	k.A.	25%

Tabelle 3.12: Überblick zu wesentlichen Merkmalen der 907 beteiligten Personen

interpretieren. Verschiedene Untersuchungen haben gezeigt, dass Teilnehmer wiederholt versuchen, das System auszunutzen und beispielsweise auf Fragen willkürlich antworten um somit die Bearbeitungszeit zu verringern (Kittur/Chi/Suh, 2008; Potthast, 2010).

Um derartige falsche Angaben schneller erkennen zu können, wurden verschiedene Kontrollmechanismen eingeführt. Unter anderem wurden mithilfe von JavaScript Messungen durchgeführt, die Aufschluss über die durch den Nutzer tatsächlich durchgeführten Arbeiten Aufschluss geben sollten. Folgende Messungen sollen dabei helfen folgende Fragen zu beantworten:

- **Unigramm Ähnlichkeit**: Wieviele Wörter verändert der Teilnehmer?
- ***n*-Gramm Ähnlichkeit** ($n = 5, 10$): Nimmt der Nutzer Änderungen an der Textstruktur vor?
- **Tastenanschläge**: Schreibt der Nutzer tatsächlich oder kopiert er fremden Text?
- **Bearbeitungszeit**: Ist der benötigte Zeitaufwand realistisch?

Weicht ein Benutzer stark von den durchschnittlichen Werten ab, ist die Wahrscheinlichkeit hoch, dass die Aufgabe nicht entsprechend der Aufgabenstellung bearbeitet wurde. Obwohl sich die Messungen als nützlicher Indikator für die Qualität erwiesen haben, ist eine manuelle Überprüfung der Ergebnisse nach wie vor erforderlich. So kopierten einige Nutzer den Text beispielsweise für die Bearbeitung aus dem Browser heraus und anschließend wieder zurück, wodurch sie sich teilweise nicht von Nutzern unterscheiden ließen, die einen fremden Text als Ergebnis verwendeten. Ein Beispiel für ein simuliertes Plagiat befindet sich in Tabelle 3.13. Zu Beginn wurde eine Reihe von Pilot-Experimenten durchgeführt, welche zum Ziel hatten, die Bezahlung pro Aufgabe in Abhängigkeit von der Textlänge und der benötigten Zeit zu ermitteln. Im Durchschnitt wurden hierbei 50 US-Cent für das Umschreiben von 500 Wörter gezahlt, was wiederum etwa eine halbe Stunde in Anspruch nahm. Dabei konnte beobachtet werden, dass sich eine Erhöhung/Verringerung der Bezahlung zwar proportional auf die Bearbeitungszeit, nicht jedoch auf die Qualität der Ergebnisse auswirkte. Diese Beobachtung stimmt mit früheren Untersuchungen überein (Mason/Watts, 2009; Potthast et al., 2010b).

Ursprungstext s_{src}	**Plagiattext** s_{plg}
The emigrants who sailed with Gilbert were better fitted for a crusade than a colony, and, disappointed at not at once finding mines of gold and silver, many deserted; and soon there were not enough sailors to man all the four ships. Accordingly, the Swallow was sent back to England with the sick; and with the remainder of the fleet, well supplied at St. John's with fish and other necessaries, Gilbert (August 20) sailed south as far as forty-four degrees north latitude.	The people who left their countries and sailed with Gilbert were more suited for fighting the crusades than for leading a settled life in the colonies. They were bitterly disappointed as it was not the America that they had expected. Since they did not immediately find gold and silver mines, many deserted. At one stage, there were not even enough man to help sail the four ships. So the Swallow was sent back to England carrying the sick. The other fleet was supplied with fish and the other necessities from St. John. On August 20, Gilbert had sailed as far as forty-four degrees to the north latitude.

Tabelle 3.13: Beispiel eines auf der Plattform von *Amazons Mechanical Turk* erstellten simulierten Plagiatfalls s. Der Ursprungstext ist ein Auszug aus „Abraham Lincoln; A History“ von Hay/Nicolay (2009).

3.3 Variablen

Bevor der Korpus erstellt werden kann, müssen einige Überlegungen bezüglich seiner Struktur und deren Bezug zur Realität angestellt werden. Dazu gehören die folgenden Punkte.

- **Sprache**: Welche soll die Grundsprache der im Korpus verwendeten Dokumente sein und aus welchen Sprachen sollen die Übersetzungsplagiate stammen?
- **Dokumentlängen**: Welche Textlänge sollen die im Korpus enthaltenen Dokumente besitzen?
- **Größe der Plagiatfälle**: Welche Textlänge sollen die eingebauten Plagiate haben?
- **Grad der Textmodifikation**: Wie hoch soll der Grad der Modifikation gemessen durch die n-Gramm-Distanz bei den künstlich erzeugten Paraphrasenplagiaten sein?
- **Plagiatanteil:** Wie hoch soll der Anteil der Plagiate pro Dokument sein?

- **Themenrelevanz zwischen d_{plg} und d_{src}:** Wie viele Plagiate sollen aus themenrelevanten Texten entnommen werden?

Als Grundsprache des Korpus wurde Englisch (*en*) gewählt, da die Weltsprache in Veröffentlichungen in Industrie und Forschungs führend ist. Als Ausgangssprache für Übersetzungsplagiate wurden Deutsch (*de*) und Spanisch (*es*) als zwei weitere, weit verbreitete Sprachen gewählt. Bei der Wahl der Dokumentlängen wird zwischen drei Kategorien unterschieden, die bei der Einteilung von Dokumenten hinsichtlich ihrer Länge sinnvoll erscheinen: kurze Texte (1-10 Seiten) wie etwa Hausarbeiten, Essays, Blogs oder wissenschaftliche Veröffentlichungen, mittlere Texte (10-100 Seiten) wie zum Beispiel Abschlussarbeiten, kurze Bücher oder Patentschriften und lange Texte (100-1000 Seiten) was vor allem Bücher betrifft. Eine ähnliche Einteilung findet bezüglich der Länge der einzelnen Plagiatfälle Anwendung. Dabei wird davon ausgegangen, dass ein Plagiator absatz- (50-150 Wörter), seiten- (300-500 Wörter) oder dokumentweise (3000-5000 Wörter) Texte aus einer Quelle übernimmt. Für den Grad der Modifikation eines Textes durch die in Abschnitt 3.2.1.2 beschriebene Heuristik werden „hoch“ und „niedrig“ als Kategorien festgelegt. Die ihnen jeweils zugrundeliegende n-Gramm-Distanz wurde durch Beobachtungen in Experimenten und in Abhängigkeit von der Länge des zu modifizierenden Textes ermittelt (siehe Tabelle 3.14). Die Abhängigkeit von der Länge des Textes gründet sich dabei auf der Überlegung, dass der Aufwand einer Modifikation für den Plagiator steigt, je länger der zu modifizierende Text wird und somit die Wahrscheinlichkeit sinkt, dass er diesen genauso stark modifiziert, wie einen kurzen Text. Im Hinblick auf den Anteil von Plagiaten in

	Absatz	**Seite**	**Dokument**
niedrig	$0,00-0,30$	$0,00-0,15$	$0,00-0,01$
hoch	$0,30-0,60$	$0,15-0,30$	$0,01-0,10$

Tabelle 3.14: Experimentell bestimmte Spanne der n-Gramm-Distanz für die Kategorien des Grades der Textmodifikation (niedrig/hoch) in Abhängigkeit von der Länge des Plagiatfalls.

einem Dokument wird davon ausgegangen, dass ein Plagiator tendenziell entweder einen hohen Plagiatanteil und einen geringen Eigenanteil oder aber genau umgekehrt verwendet. Für Texte, in denen Plagiatfälle intrinsisch erkannt werden sollen, wird der Plagiatanteil trotz dieser Annahme auf maximal 50% begrenzt, um das Funktionieren entsprechender Algorithmen zu ermöglichen. Um bei der Auswertung Unterschiede in der Bedeutung des heuristischen Retrievals im Prozess der externen Plagiaterkennung feststellen zu können, soll

bei der Wahl eines d_{src} im Hinblick auf die thematische Relevanz zu d_{plg} ein relevantes Dokument mit der gleichen Wahrscheinlichkeit ausgewählt werden, wie ein irrelevantes. Eine Zusammenfassung der Entscheidungen und der daraus resultierenden Statistik befindet sich in Tabelle 3.15.

Dokument- und Plagiatfallstatistik					
Plagiate pro Dokument			**Themenrelevanz zw. d_{src} und d_{plg}**		
wenig	(5%-20%)	45%	relevant		50%
mittel	(20%-50%)	15%	irrelevant		50%
viel	(50%-80%)	25%			
gesamt	(>80%)	15%			
Dokumentlänge			**Länge der Plagiatfälle**		
kurz	(1-10 pp.)	45%	kurz	(50-150 Wörter)	34%
mittel	(10-100 pp.)	15%	mittel	(300-500 Wörter)	33%
lang	(100-1000 pp.)	25%	lang	(3000-5000 Wörter)	33%

Tabelle 3.15: Dokument- und Plagiatfallstatistik: gibt den prozentualen Anteil der entsprechenden Dokument-/Plagiatfallkategorie an der Gesamtmenge der Dokumente/Plagiatfälle an.

3.4 Korpuslayout

Der Korpus ist in seinem Aufbau in zwei wesentlichen Teile unterteilt. D_{susp} enthält die Dokumente, die möglicherweise Plagiatfälle enthalten und D_{src} enthält alle Dokumente aus denen ein Plagiat stammen könnte. Dabei gilt dass $|D_{susp}| = |D_{src}|$ und $D_{susp} \cap D_{src} = \emptyset$. Die Menge der Dokumente, in denen tatsächlich Plagiate enthalten sind (D_{plg}) machen 50% von D_{susp} aus. Der im Vergleich zur gesamten Textmenge im Korpus somit deutlich unter 50% liegende Plagiatanteil soll verhindern, dass ein Algorithmus der alle oder zufällig Texte als Plagiate annotiert im Mittel in über 50% der Fälle richtig liegt. Die Verteilung der verschiedenen Plagiatformen in D_{plg} ist in Tabelle 3.16 aufgelistet.

3.5 Plagiatannotation

Um die Ergebnisse von Algorithmen zur Plagiaterkennung später auswerten zu können, ist eine entsprechende Annotation der Plagiate notwendig. Dabei geht es vor allem darum, die exakte Position sowie Metainformationen bezüglich

Intrinsische Erkennung	
Wort-für-Wort-Plagiate	30%
Externe Erkennung	
Wort-für-Wort-Plagiate	28%
Paraphrasenplagiate	28%
Übersetzungsplagiate	10%
Simulierte Plagiate	4%

Tabelle 3.16: Häufigkeitsverteilung der verschiedenen Plagiatformen in D_{plg}.

der eingefügten Plagiate und des Dokuments festzuhalten. Im Gegensatz zu den meisten linguistischen Korpora, in denen Metainformationen in der Regel direkt im Text abgelegt werden (Leech, 2005), sollen sie hier in einer separaten Datei abgelegt werden. Dies hat den Hintergrund, dass Informationen über die Position der Plagiate in der Phase der Erkennung durch einen Algorithmus nicht erwünscht sind und somit erst entfernt werden müssten. Anschließend würde dies die Zuordnung der durch den Algorithmus zurückgelieferten Plagiatpositionen zu den entsprechenden Textpositionen im annotierten Text jedoch deutlich erschweren.

Für die externe Speicherung dieser Metadaten kommt die *Extensible Markup Language* (XML) zum Einsatz. Hierbei handelt es sich neben der *JavaScript Object Notation* (JSON) und YAML um eine der am weitesten verbreiteten Auszeichnungssprachen im Internet. Im Gegensatz zu JSON und YAML wurde XML jedoch durch das *World Wide Web Consortium* standardisiert und lässt die Definition von anwendungsspezifischen Sprachen auf Basis von Schemasprachen wie *DTD* und *XML Schema* zu. Für die Speicherung von Dokument- und Plagiatmetadaten definieren wir folgendes XML-Schema (Listing 3.1).

Listing 3.1: XML Schema der Metadatei

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <xsd:element name="document" type="document" />

  <xsd:complexType name="document">
    <xsd:sequence>
      <xsd:element name="feature" type="feature" minOccurs
         ="0" maxOccurs="unbounded" />
    </xsd:sequence>
    <xsd:attribute name="reference" type="xsd:anyURI" use="
       required"/>
  </xsd:complexType>

  <xsd:complexType name="feature">
```

```
      <xsd:attribute name="name" type="xsd:string" use="
         required"/>
      <xsd:anyAttribute namespace="##any" processContents="
         skip" />
    </xsd:complexType>

  </xsd:schema>
```

Um Plagiatfälle in einer Datei «*filename*.txt»zu annotieren, wird eine gleichnamige «*filename*.xml»angelegt. Diese wiederum verweist im *reference*-Attribut des Wurzelelements (*document*) auf die Textdatei («*filename*.txt»), für welche sie die Plagiatfälle annotiert. Für die eigentliche Annotation wird das *feature*-Tag verwendet. Die dabei möglichen Attribute und deren Werte sind in Tabelle 3.17 aufgeführt. Auch die Metainformationen zu dem Textdokument werden als *feature*-Tag abgelegt. Die Liste der Attribute findet sich in Tabelle 3.18.

Attributliste zur Plagiatannotation		
Attribut	**Werte**	**Beschreibung**
name	plagiarism	Identifiziert *feature*-Tag als Plagiatannotation
type	{artifical,simulated, translated,detected}	Typ des annotierten Plagiats
this_offset	*Integer*	Buchstaben-Offset von s_{plg} in d_{plg}
this_length	*Integer*	Länge von s_{plg} in Buchstaben
this_language	{en}	Sprache in der s_{plg} verfasst ist
source_reference*	*Dateiname*	Dateiname von d_{src}
source_offset*	*Integer*	Buchstaben-Offset von s_{src} in d_{src}
source_length*	*Integer*	Länge von s_{src} in Buchstaben
source_language*	{en,de,es}	Sprache in der s_{src} verfasst ist
obfuscation**	{none,low,high}	Stärke der Modifikationen gemessen anhand der n-Gramm Distanz
obfuscator_version**	{2009,2010}	Modifikation von s_{plg} mit (2010) oder ohne (2009) Verwendung einer Heuristik zur Erhaltung der Textstruktur.
intra_cluster**	{true,false}	Befinden sich d_{plg} und d_{src} im selben thematischen Cluster.

* Attribut wird bei Plagiatfällen für intrinsische Verfahren nicht verwendet.
** Attribut wird nur bei Plagiatfällen vom Typ „*artificial*“ verwendet.

Tabelle 3.17: Liste der Attribute bei der Annotation von Plagiatfällen

Attributliste für Dokumentmetadaten		
Attribut	**Wert**	**Beschreibung**
name	about	Identifiziert *feature*-Tag Metainformation des zugehörigen Dokuments
authors	*String*	Komma-separierte Liste der Autoren
title	*String*	Titel/Überschrift des Dokuments
cluster	*Integer*	Nummer des Clusters dem das Dokument angehört

Tabelle 3.18: Liste der Möglichen Attribute bei der Annotation von Metadaten des Textdokuments

3.6 Konstruktionspipeline

Die Basis für den Korpus bilden Bücher aus dem *Project Gutenberg*[2], welches derzeit ca. 33.000 Dokumente verteilt auf über 60 Sprachen zur Verfügung stellt. Der überwiegende Teil der Bücher enthält Erzähltext, aber auch Wissenschafts- und Gesetzestexte. Die Besonderheit dieses Projektes besteht darin, dass ein Großteil der Dokumente sowohl unter der *Public Domain* Lizenz veröffentlicht wird, als auch in reinem Textformat zur Verfügung steht. Ersteres ist Voraussetzung dafür, dass die Dokumente während der Erstellung des Korpus modifiziert und anschließend im Rahmen der Evaluierungsumgebung veröffentlicht werden dürfen. Letzteres verhindert vor allem ungewollte Textanomalien, die bei der Konvertierung von propritären Formaten ins Textformat auftreten können.

Für die in den Korpus einfließenden Sprachen (*en*, *de*, *es*) ergeben sich zunächst folgende Verteilung:

Die in den folgenden Unterkapiteln vorgestellte Konstruktionspipeline besteht, wie in Abbildung 3.2 dargestellt, aus drei wesentlichen Schritten, in denen die Dokumente des *Project Gutenberg* so aufbereitet werden, dass sie im Anschluss den in Abschnitt 3.3 festgelegten Vorgaben entsprechen.

[2] `http://www.gutenberg.org`

	en	*de*	*es*
Anzahl der Dokumente:	20.702	595	238
Größe des Korpus (MB):	4.300	173	91

Tabelle 3.19: Statistik über die Verteilung der Dokumente über die verwendeten Sprachen

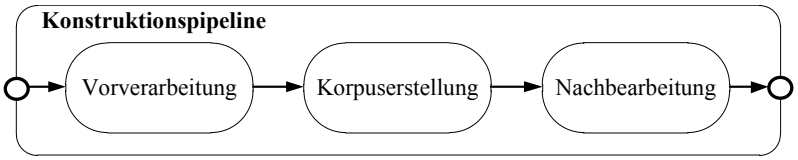


Abbildung 3.2: Korpuspipeline

3.6.1 Vorverarbeitung

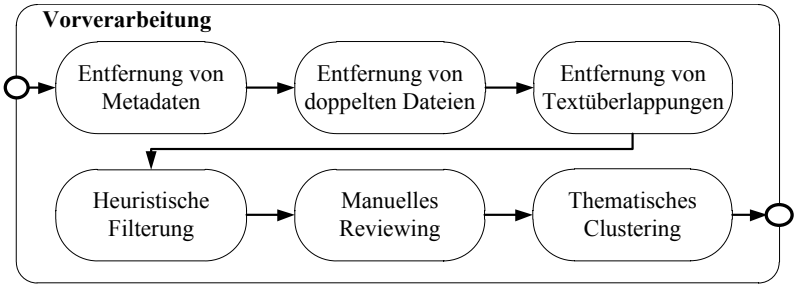


Abbildung 3.3: Schematische Darstellung der einzelnen Etappen der Vorverarbeitung als Teil der Konstruktionspipeline

Die Vorverarbeitung besteht im wesentlichen aus sechs Schritten (siehe Abbildung 3.3). Im ersten Schritt werden alle Metadaten, wie Hinweise auf das *Project Gutenberg*, Kommentare der Ersteller des Dokumentes, sowie Texte, die nicht Teil des Buches sind, mit Hilfe von Heuristiken entfernt. Dies ist aus lizenzrechtlichen Gründen notwendig, da eine Veröffentlichung einer modifizierten Kopie der Dokumente nur erlaubt ist, wenn vorher sämtliche Verweise auf das Projekt Gutenberg entfernt wurden:

> „If you strip the Project Gutenberg license and all references to Project Gutenberg from the ebook, you are left with a public domain ebook. You can do anything you want with that.“ (URL: `http://www.gutenberg.org/wiki/Gutenberg:The_Project_Gutenberg_License`; Stand: 3. Januar 2009)

Im zweiten Schritt werden Dokumente entfernt, die in der Kollektion doppelt vorkommen. Dies kann beispielsweise der Fall sein, wenn von einem Buch mehrere Auflagen oder Ausgaben vorhanden sind. Im Fall der Bibel sind dies immerhin 223 verschiedene Versionen. Da diese Form der Textkopie auch als Plagiat aufgefasst werden kann, sie sich jedoch einer Kontrolle hinsichtlich des Aufbaus des Korpus sowie der Evaluierung entzieht, werden alle entsprechenden Texte im Rahmen der Vorverarbeitung entfernt. Hierfür werden zunächst alle Bücher des selben Autors auf Basis des Kosinus-Ähnlichkeitsmaßes miteinander verglichen. Liefert dies einen überdurchschnittlich hohen Wert ($> 0,5$), wird angenommen, dass es sich bei den Dokumenten um Kopien handelt. Eines der beiden Dokumente wird daraufhin entfernt. Auf diese Weise konnten 934 Dokumente aussortiert werden.

Um auszuschließen, dass der Korpus weitere ungewollte mehrfach vorkommende Texte enthält, wurde zusätzlich ein invertierter Index aller im Korpus vorkommenden Wort-n-Gramme erstellt. Dabei wurde n schrittweise von 64 bis auf 16 gesenkt, um zu beobachten ab welcher n-Gramm-Länge anstatt von Textkopien, allgemein gebräuchliche Formulierungen erfasst werden. Hierbei wurde $n = 16$ als kleinster Wert ermittelt, bei dem fast ausschließlich ungewollte Textkopien erfasst wurden. Alle 16-Gramme die im Korpus mehr als einmal vorkamen, wurden im dritten Schritt der Vorverarbeitung zunächst ermittelt und anschließend entfernt. Um den umgebenden Text nicht zu zerstören, wurde die Entfernung auf Satzebene durchgeführt. Auf diese Weise konnten ca. 300 MB an unerwünschten Textkopien entfernt werden.

Im nächsten Schritt wurden mit Hilfe von Heuristiken 3307 Dokumente gefiltert, die nur zu geringen Teilen aus Fließtext bestanden. 2305 weitere Dateien konnten bei einer manuellen Begutachtung aussortiert werden, da sie aus unterschiedlichen Gründen nicht für die Entnahme oder das Einfügen von Plagiaten geeignet erschienen.

In Vorbereitung auf die Erstellung des Korpus wurden alle Dokumente bezüglich ihrer Kosinus-Ähnlichkeit in n Cluster aufgeteilt. Experimente ergaben dabei eine bei einem Wert von $n = 30$ optimale Verteilung der Dokumente auf verschiedene thematische Kategorien (unter anderem: Kochen, Regierung (US), Religion, Krieg, Natur, Schiff- und Seefahrt, Geschichten, Biologie, Chemie, etc.). Dieser Schritt schafft die Grundlage für die Auswahl themenverwandter Dokumente d_{src} und d_{plg} bei der Erstellung entsprechender Plagiatfälle.

3.6.2 Korpuskonstruktion

Abbildung 3.4 stellt den Prozess der Korpuskonstruktion schematisch dar. Da es sich bei den Dokumenten des Projekts Gutenberg fast ausschließlich um Tex-

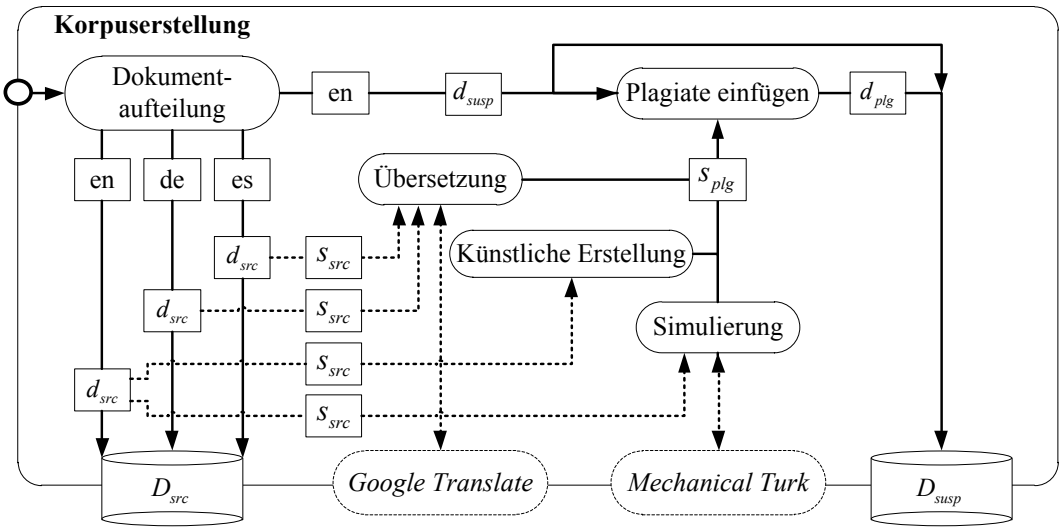


Abbildung 3.4: Schematische Darstellung der Vorverarbeitung als Teil der Konstruktionspipeline

te in Buchlänge (> 100 Seiten) handelt, der Korpus jedoch einen bestimmten Anteil seiner Dokumente in unterschiedlichen Dokumentlängenkategorien aufweisen soll (siehe Tabelle 3.15), werden die Dokumente entsprechend den Vorgaben auf Absatzebene geteilt und auf die entsprechenden Dokumentlängenkategorien verteilt. Anschließend werden die fremdsprachigen Dokumente, sowie 50% der englischen Dokumente D_{src} zugeordnet. Weitere 25% der englischen Dokumente werden den Vorgaben des Korpuslayouts entsprechend in D_{susp} verschoben, ohne dass ein Plagiat eingefügt wird. In die restlichen 25% der englischen Dokumente werden mithilfe der in Abschnitt 3.2 vorgestellten Strategien zur Erzeugung von Plagiaten und auf Grundlage der in Abschnitt 3.3 festgelegten Variablen die Plagiate s_{plg} erstellt, in d_{plg} eingefügt und annotiert (siehe Abschnitt 3.5).

3.6.3 Nachbearbeitung

Im Rahmen der Nachbearbeitung wird die Struktur des Korpus auf das Dateisystem abgebildet. Zu diesem Zweck werden die zwei Ordner *source-document* und *suspicious-document* angelegt, in die jeweils die Dokumente aus D_{src} und D_{susp} abgelegt werden. Den Dokumenten wird dabei in zufälliger Reihenfolge eine chronologischer Name zugewiesen. Dabei werden ebenfalls die entsprechenden Referenzen in den XML-Dateien angepasst.

Um eine Evaluierung auf dem Korpus durchführen zu können, wird er in zwei verschiedene Sets aufgeteilt. Das sogenannte Test-Set enthält neben den Textdateien auch die XML-Dateien, jedoch ohne die entsprechenden Plagia-

tannotationen (siehe Tabelle 3.20). Die XML-Dateien dienen in diesem Fall lediglich als Platzhalter für Annotationen durch Plagiaterkennungsalgorithmen. Das Evaluierungs-Set stellt gewissermaßen eine Ergänzung des Test-Sets dar, da es nur die XML-Dateien jedoch mit den entsprechenden Plagiatannotationen enthält. Dabei ist zu beachten, dass sich z.B. „suspicious-document00001.xml“ aus dem Evaluierungs-Set auf die Plagiate in „suspicious-document00001.txt“ des Test-Sets bezieht.

source-document	**suspicious-document**
suspicious-document00001.txt	source-document00001.txt
suspicious-document00001.xml	source-document00001.xml
suspicious-document00002.txt	source-document00002.txt
suspicious-document00002.xml	source-document00002.xml
...	...

Tabelle 3.20: Abbildung der Korpusstruktur auf das Dateisystem; Test-Set: die XML-Dateien enthalten keine Plagiatannotation, sondern dienen nur als Platzhalter.

source-document	**suspicious-document**
suspicious-document00001.xml	source-document00001.xml
suspicious-document00002.xml	source-document00002.xml
...	...

Tabelle 3.21: Abbildung der Korpusstruktur auf das Dateisystem; Evaluierungs-Set: hier werden nur die XML-Dateien mit den entsprechenden Plagiatannotation benötigt.

Das Test-Set besteht nach Abschluss der Korpuskonstruktion aus 27073 Dokumente (ohne die XML Dateien) mit 68558 enthaltenen Plagiatfällen.

Kapitel 4

Evaluierungsmaße

Zur Bestimmung der Genauigkeit von Algorithmen zur Plagiaterkennung wird im folgenden Kapitel ein auf *Precision* und *Recall* basierendes Evaluierungsmaß vorgestellt. Da es sich bei Plagiaterkennung nicht nur um eine Retrieval-, sondern auch um eine Extraktionsaufgabe handelt, soll hier ein Ansatz entwickelt werden, der diesen Aspekt ebenfalls berücksichtigt. Das vorgestellte Konzept basiert auf den bereits in Potthast et al. (2009); Potthast et al. (2010a); Potthast et al. (2010b) veröffentlichten Informationen.

4.1 Klassische Leistungsmaße im Information Retrieval

Will man die Leistung eines Systems in Bezug auf eine Retrieval-Aufgabe messen, so werden in der Regel die zwei Maße *Precision* und *Recall* verwendet. Diese werden unter der Annahme, dass ein IR-System auf eine Anfrage q aus einer Dokumentmenge X eine Menge von Dokumenten $X_{res} \subseteq X$ zurückliefert und dass X_q die Menge aller in Bezug auf q relevanten Dokumente aus X ist, wie folgt definiert:

- ***Precision*** ist der Anteil der zurückgelieferten Dokumente, die relevant sind.

$$prec(X_q, X_{res}) = \frac{|X_{res} \cap X_q|}{|X_{res}|} \tag{4.1}$$

- ***Recall*** ist der Anteil der relevanten Dokumente, die zurückgeliefert wurden.

$$rec(X_q, X_{res}) = \frac{|X_{res} \cap X_q|}{|X_q|} \tag{4.2}$$

Die Maße sind einzeln jedoch nur bedingt aussagekräftig. Liefert ein IR-System zum Beispiel neben allen relevanten auch alle irrelevanten Elemente, wäre der *Recall* dennoch 1.0. Ebenso wäre die *Precision* stets 1.0 wenn das System nur ein relevantes Element aus der Menge X_{rel} zurückliefern würde. Aus diesem Grund wird das F-Measure als harmonisches Mittel aus *Precision* und *Recall* als Qualitätsmaß verwendet:

$$F_\beta = \left(1 + \beta^2\right) \cdot \frac{prec \cdot rec}{(\beta^2 \cdot prec) + rec} \tag{4.3}$$

Dabei bestimmt β wie stark *Precision* und *Recall* jeweils gewichtet werden. Eine gleichmäßige Gewichtung ($\beta = 1$) ist dabei nicht immer sinnvoll. Für $\beta < 1$ würde die *Precision* stärker gewichtet, für $\beta > 1$ der *Recall*. Im Fall der Plagiaterkennung scheint eine Gleichgewichtung jedoch sinnvoll, da bisher keine Erkenntnis eine stärkere Gewichtung einer der beiden Maße begründet. Somit ergibt sich in vereinfachter Form:

$$F_{\beta=1} = \frac{2 \cdot prec \cdot rec}{prec + rec} \tag{4.4}$$

4.2 Maß zur Quantifizierung der Plagiaterkennungsleistung

Für den Fall der Plagiaterkennung betrachten wir zunächst einen Plagiatfall s in einem Dokument d_{plg}, welcher sich als Quadtuple $s = \langle s_{plg}, d_{plg}, s_{src}, d_{src} \rangle$ definieren lässt. Dabei stellt s_{plg} einen plagiierten Textabschnitt in d_{plg} und s_{src} den entsprechenden Ursprungsabschnitt im Quelldokument d_{src} dar (siehe Abbildung 4.1).

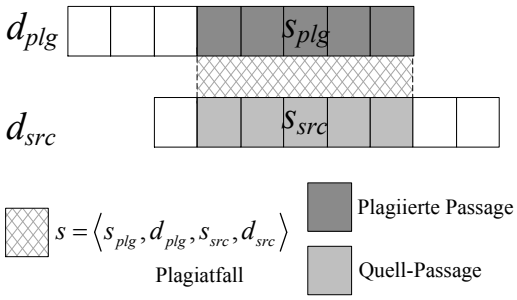


Abbildung 4.1: Schematische Darstellung eines Plagiatfalls

Ebenso lässt sich ein Plagiaterkennungsfall r als $r = \langle r_{plg}, d_{plg}, r_{src}, d'_{src} \rangle$ definieren, wobei r einen vorgeblich plagiierten Textabschnitt r_{plg} in d_{plg} mit

einer Textpassage r_{src} in einem möglichen Quelldokument d'_{src} in Verbindung bringt (siehe Abbildung 4.2).

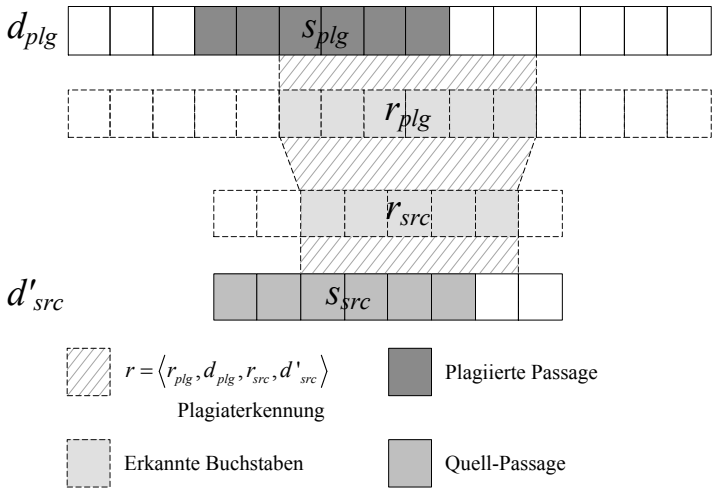


Abbildung 4.2: Illustrierung eines Plagiaterkennungsfalls. In diesem Beispiel werden s_{plg} und s_{src} durch r_{plg} und r_{src} zwar erkannt, allerdings stimmen die Grenzen nicht überein.

Dabei gilt „r *erkennt* s“ genau dann, wenn $r_{plg} \cap s_{plg} \neq \emptyset$, $r_{src} \cap s_{src} \neq \emptyset$ und $d'_{src} = d_{src}$. Des Weiteren beschreiben S und R die Menge der Plagiat- und Plagiaterkennungsfälle.

Betrachtet man nun ein Dokument d als eine Menge von Referenzen auf dessen Buchstaben, so erhält man folgende Formalisierung:

$$\mathbf{d} = \{(1, d), ..., (|d|, d)\}$$

wobei (i, d) den i-ten Buchstaben aus d referenziert. Dementsprechend kann ein Plagiatfall s als

$$\mathbf{s} = \mathbf{s}_{plg} \cup \mathbf{s}_{src}$$

beschrieben werden, wobei gilt, dass $\mathbf{s}_{plg} \subseteq \mathbf{d}_{plg}$ die Textpassage s_{plg} und $\mathbf{s}_{src} \subseteq \mathbf{d}_{src}$ die Textpassage s_{src} bildet.

Äquivalent kann ein Plagiaterkennungsfall r als

$$\mathbf{r} = \mathbf{r}_{plg} \cup \mathbf{r}_{src}$$

beschrieben werden. Dementsprechend gilt die Aussage „r *erkennt* s“ genau dann, wenn $\mathbf{r}_{plg} \cap \mathbf{s}_{plg} \neq \emptyset$ und $\mathbf{r}_{src} \cap \mathbf{s}_{src} \neq \emptyset$.

Entsprechend dieser Notationen lassen sich *Precision* und *Recall* wie folgt definieren:

$$prec_{micro}(S,R) = \frac{|\bigcup_{(s,r)\in(S\times R)} (\mathbf{s} \sqcap \mathbf{r})|}{|\bigcup_{r\in R} \mathbf{r}|} \tag{4.5}$$

$$rec_{micro}(S,R) = \frac{|\bigcup_{(s,r)\in(S\times R)} (\mathbf{s} \sqcap \mathbf{r})|}{|\bigcup_{s\in S} \mathbf{s}|} \tag{4.6}$$

$$\text{wobei } \mathbf{s} \sqcap \mathbf{r} = \begin{cases} s \cap r, & \text{„}r \text{ erkennt } s\text{„} \\ \emptyset, & \text{sonst} \end{cases}$$

Soll im Unterschied dazu die Länge der Plagiatfälle nicht berücksichtigt werden, definieren sich *Precision* und *Recall* wie folgt:

$$prec_{macro}(S,R) = \frac{1}{|R|} \sum_{r\in R} \frac{|\bigcup_{s\in S} (\mathbf{s} \sqcap \mathbf{r})|}{|\mathbf{r}|} \tag{4.7}$$

$$rec_{macro}(S,R) = \frac{1}{|S|} \sum_{s\in S} \frac{|\bigcup_{r\in R} (\mathbf{s} \sqcap \mathbf{r})|}{|\mathbf{s}|} \tag{4.8}$$

Da es sich bei Plagiaterkennung nicht nur um eine Retrieval- sondern auch um eine Extraktionsaufgabe handelt, wird noch ein weiterer wichtiger Aspekt in das Maß einfließen: in wie vielen Teilen wird s erkannt. Optimal wäre hierbei eine injektive Abbildung von $f\colon R \to S$, d.h. maximal ein Plagiaterkennungsfall r pro Plagiatfall s. Um die Leistung eines Algorithmus in Bezug auf diesen Aspekt zu messen, definieren wir die *Granularity* von R bezüglich S als:

$$gran(S,R) = \frac{1}{|S_R|} \sum_{s\in S_R} |R_s| \tag{4.9}$$

wobei $S_R \subseteq S$ Plagiatfälle sind, die von Plagiaterkennungsfällen in R erkannt wurden und $R_s \subseteq R$ Plagiaterkennungsfälle die ein bestimmtes s erkennen:

$$S_R = \{s | s \in S \wedge \exists r \in R : r \text{ erkennt } s\}$$
$$R_s = \{r | r \in R \wedge r \text{ erkennt } s\}$$

Der Wertebereich von $gran(S,R)$ liegt für ein Dokument der Länge n bei $[1, \frac{n\cdot(n-1)}{2}]$, wobei der Wert 1 für die optimale injektive Beziehung zwischen R und S steht und $\frac{n\cdot(n-1)}{2}$ für den schlechtesten Fall, in dem mindestens ein $s \in S$ wieder und wieder erkannt wurde.

Um die Gesamtleistung von Plagiaterkennungsalgorithmen vergleichen zu können, müssen *Precision*, *Recall* und *Granularity* zu einem Gesamtmaß zusammengefügt werden:

$$plagdet(S, R) = \frac{F_\beta}{\log_2 (1 + gran(S, R))} \tag{4.10}$$

Der Logarithmus dient hierbei zur Verringerung der Gewichtung der *Granularity*.

Kapitel 5

PAN Competition

In diesem Kapitel sollen die Ergebnisse der im Rahmen des PAN-Wettbewerbs 2010 zum Einsatz gekommenen Algorithmen zur Plagiaterkennung vorgestellt und diskutiert werden. Die Aufgabe bestand darin, in dem in Kapitel 3 vorgestellten Korpus, alle Plagiate zu finden und ihre exakte Position anzugeben. Die entsprechenden Annotationen der Plagiate wurden für diese Aufgabe aus dem Korpus entfernt.

Insgesamt traten in dem Wettbewerb 18 Systeme gegeneinander an. Der externen Plagiatanalyse wurde dabei deutlich mehr Aufmerksamkeit gewidmet, als der Intrinsischen. Das mag unter anderem daran liegen, dass die Forschung auf dem Gebiet der intrinsischen Plagiatanalyse im Vergleich zur externen noch in den Kinderschuhen steckt. Lediglich Muhr et al. (2010) und Grozea/Popescu (2010) kombinierten beide Plagiaterkennungsstrategien (intrinsisch/extern) in ihren Systemen, wohingegen Suárez/González/Villena-Román (2010) als einziger Teilnehmer ausschließlich auf intrinsische Plagiaterkennung setzte. Festzustellen war auch, dass die Mehrheit der Teilnehmer dem in Kapitel Abschnitt 2.2 vorgestellt allgemeinen Ablauf bei der Plagiaterkennung folgten. Im folgenden sollen nun die Ergebnisse jedes Teilnehmers vorgestellt und anschließend in Abhängigkeit zu den in Abschnitt 3.3 vorgestellten Variablen des Korpus untersucht werden.

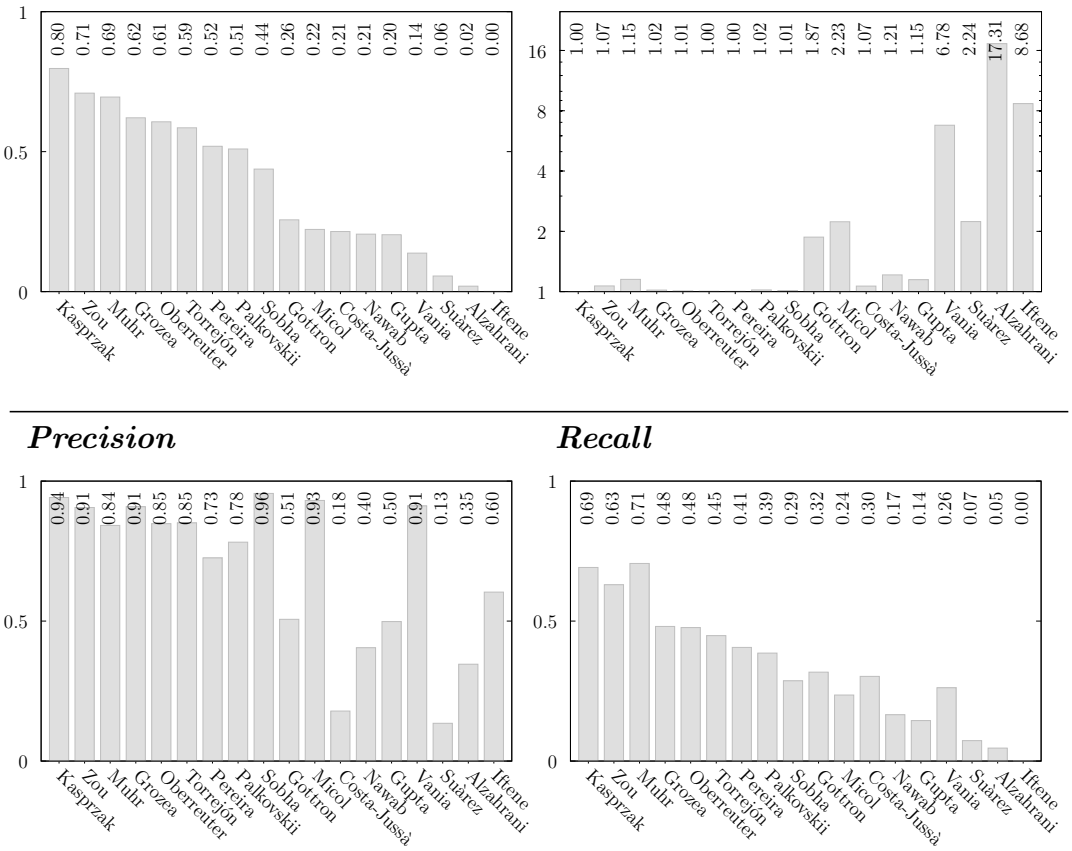


Tabelle 5.1: Erkennungsleistung für den gesamten Korpus

Tabelle 5.1 zeigt zunächst die Gesamtergebnisse des PAN-Wettbewerbs 2010. Das Ranking ergibt sich dabei aus dem *plagdet* Wert, welcher sich wiederum aus den Werten von *Precision*, *Recall* und *Granularity* ergibt. Das beste Ergebnis erzielte dabei Kasprzak/Brandejs (2010), der mit seinem Ergebnis über 10 Prozentpunkte höher lag als der Zweit- und Drittplazierte (Zou/Long/Ling, 2010; Muhr et al., 2010). Die weiteren Ergebnisse schwanken sehr stark zwischen gut und sehr schlecht. Bezüglich der *Precision* können grob zwei Gruppen unterschieden werden: jene mit einer hohen *Precision* ($> 0,7$) und jene mit einer niedrigen. Dabei erzielt fast jeder Algorithmus mit einer hohen *Precision* auch einen hohen Platz im Ranking. Während der *Recall* recht genau der Tendenz des Rankings folgt, verhält es sich mit den Werten der *Granularity* genau umgekehrt. Dabei ist zu beachten, dass der beste zu errei-

externen Plagiaterkennung

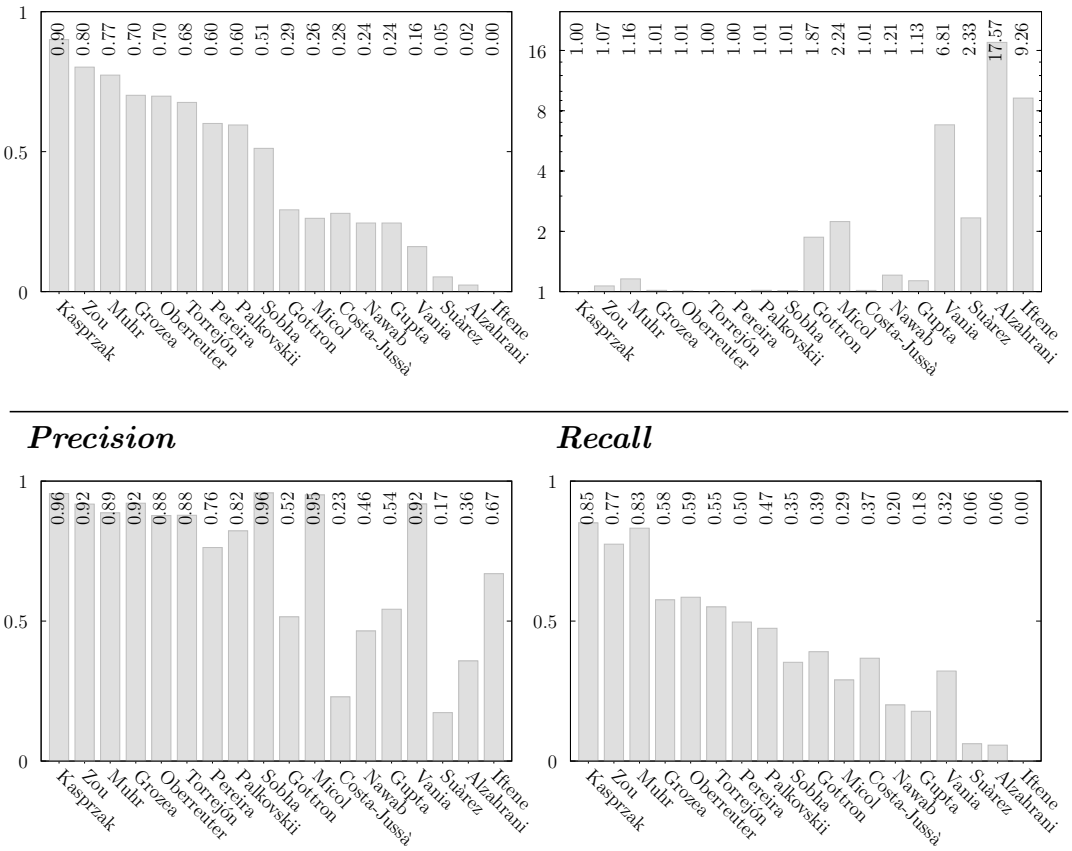


Tabelle 5.2: Erkennungsleistung der externen Plagiaterkennung

chende Wert bei der *Granularity* 1 ist. Werte die darüber liegen markieren ein schlechteres Ergebnis, was sich bei den Letztplatzierten deutlich zeigt. Trotz guter Werte bei einem Leistungsmaß zeigt sich bei einigen Teilnehmern, dass für ein gutes Gesamtergebnisse auch die anderen Maße von Bedeutung sind. Immerhin erzielt der Gewinner bei jedem von ihnen sehr gute Werte.

Hinsichtlich der möglichen Plagiaterkennungsstrategien war der Korpus in zwei Teile geteilt. Ca. 30% der Plagiatfälle waren nur mithilfe von intrinsischen Verfahren zu erkennen, da die entsprechenden Quelldokumente d_{src} nicht im Korpus enthalten waren. Die restlichen 70% der Plagiatfälle im Korpus war mit Hilfe von Verfahren der externen Plagiatanalyse auffindbar. Bei der Betrachtung der Ergebnisse in Tabelle 5.2 und Tabelle 5.3 wird deutlich, dass sich die meisten Teilnehmer nur auf externe Analyseverfahren konzentrierten.

intrinsischen Plagiaterkennung

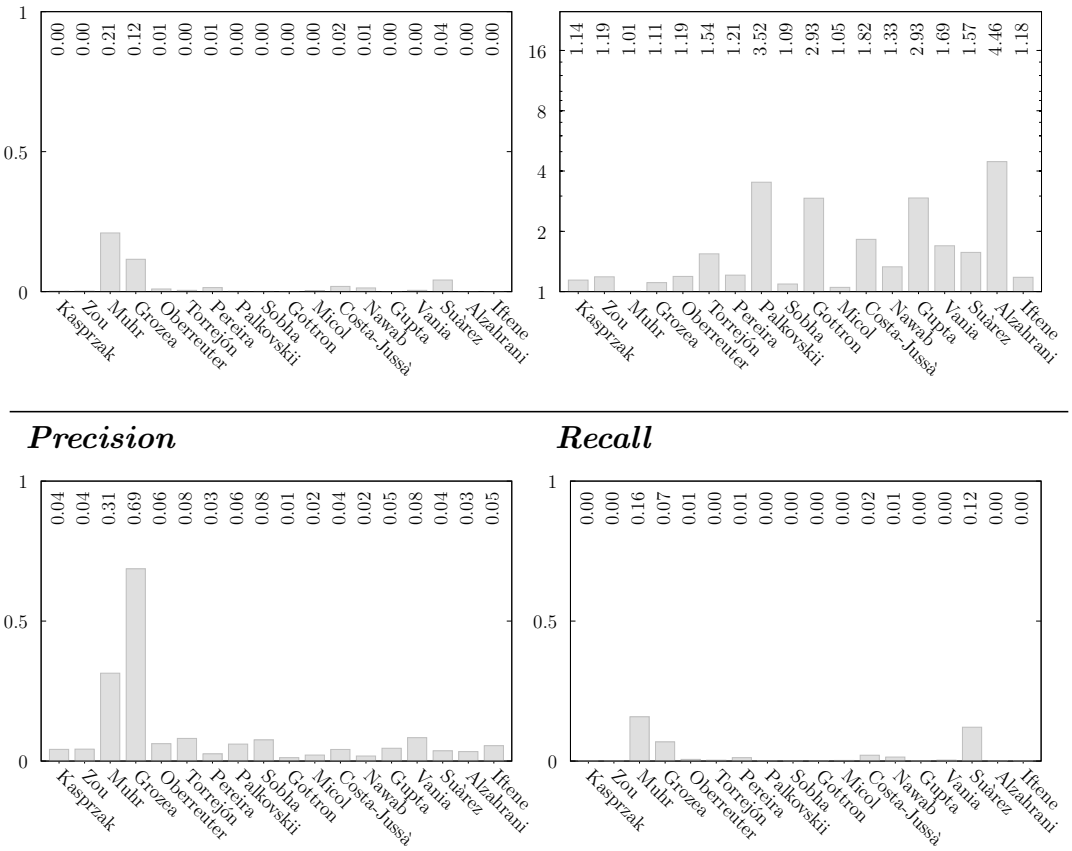


Tabelle 5.3: Erkennungsleistung der intrinsischen Plagiaterkennung

Lediglich die drei Teilnehmer Muhr et al. (2010), Grozea/Popescu (2010) und Suárez/González/Villena-Román (2010) erzielten nennenswerte Ergebnisse bei der intrinsischen Erkennung. Nichtsdestotrotz bleiben die Ergebnisse, die auf diese Weise erzielt werden konnten unbefriedigend. Bei den Ergebnissen der externen Plagiatanalyse kann hingegen ein sogar 20-30% höherer *Recall* als bei den Gesamtergebnissen beobachtet werden. Dies ist dem Umstand geschuldet, dass nur die 70% der im Korpus enthaltenen Plagiatfälle betrachtet wurde, die sich mit externen Verfahren erkennen lassen.

Hinsichtlich der unterschiedlichen Modifikationsgrade von s_{plg} in Bezug auf s_{src}, waren zumindest die höher platzierten Systeme in der Lage, nahezu alle Wort-für-Wort-Plagiate richtig zu erkennen (siehe Tabelle 5.4). Entgegen den Erwartungen fiel die erzielte Leistung jedoch auch bei einem leichten und

höheren Grad der Modifikation (Tabelle 5.5) kaum ab. Lediglich der *Recall* fiel mit steigendem Modifikationsgrad bei allen Teilnehmern leicht ab, während die *Granularity* bei einigen leicht zunahm. Stärkere Probleme hatten die Algorithmen mit der Erkennung von den simulierten Plagiaten. Interessanterweise waren die besten Methoden nicht die im Gesamtranking Höchstplazierten.

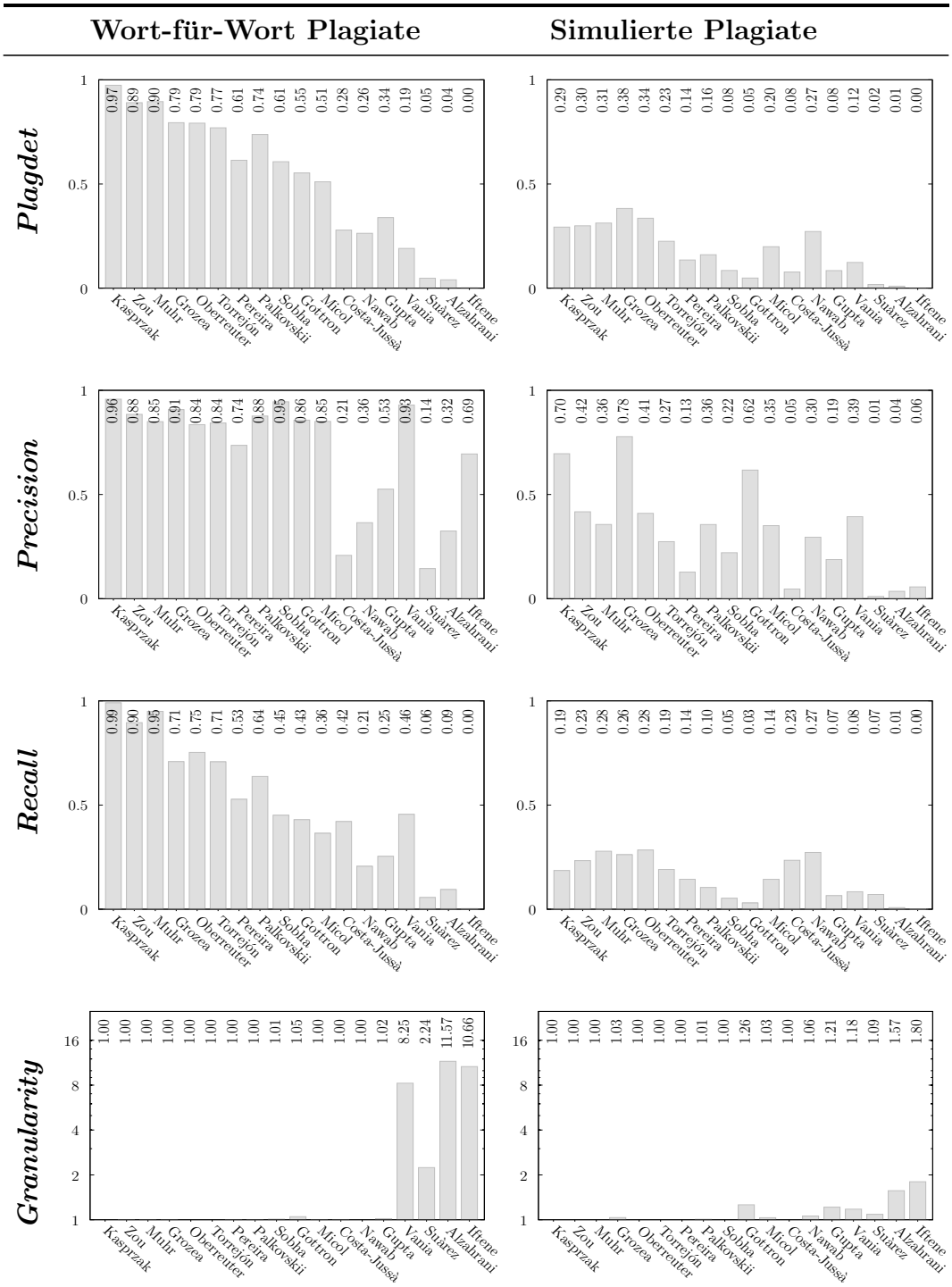


Tabelle 5.4: Erkennungsleistung für künstliche Wort-für-Wort- und simulierte Plagiate

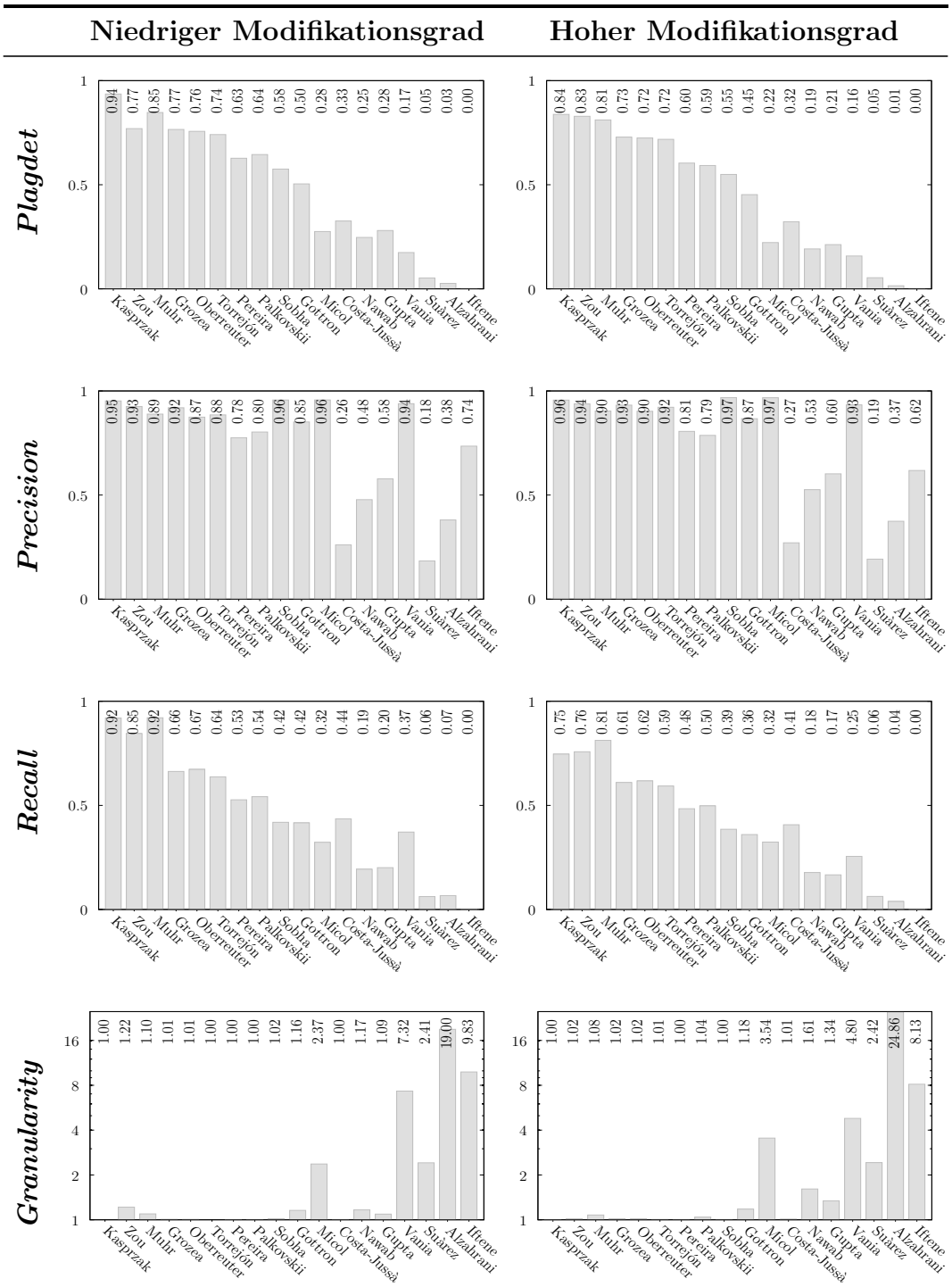


Tabelle 5.5: Erkennungsleistung für künstliche Paraphrasenplagiate mit niedrigem und hohem Modifikationsgrad

Übersetzungsplagiate

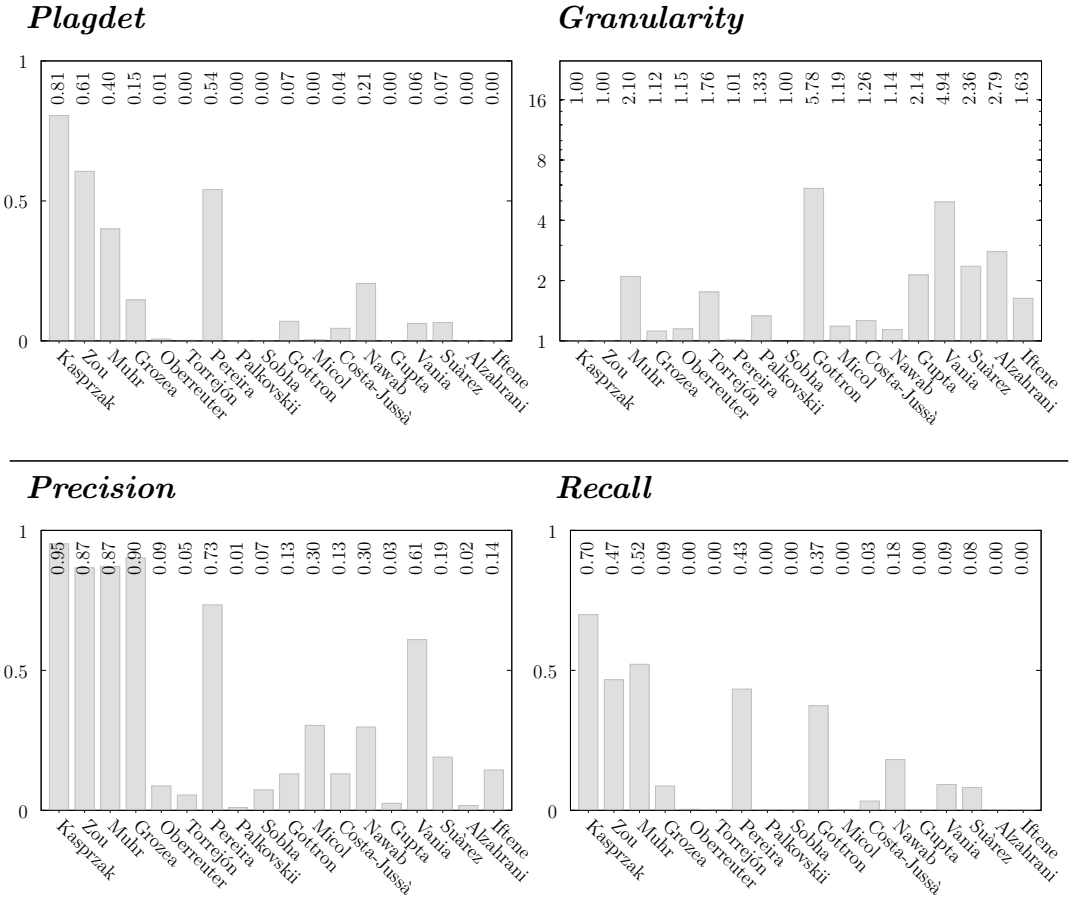


Tabelle 5.6: Erkennungsleistung für Übersetzungsplagiaten

Übersetzungsplagiate wurden von den Systemen gut erkannt, die alle fremdsprachigen Dokumente im Korpus zunächst maschinell übersetzten. Auffällig ist jedoch, dass einige Systeme hierbei noch starke Probleme mit der *Granularity* hatten.

Hinsichtlich der Länge der Plagiatfälle (Tabelle 5.7 und Tabelle 5.8) und der Länge der Dokumente (Tabelle 5.9 und Tabelle 5.10) kann beobachtet werden, dass je länger die Plagiatfälle oder die Dokumente sind, die Systeme die Plagiate besser erkennen. Dieser Umstand dürfte damit zusammenhängen, dass längere Plagiate schwächer modifiziert wurden als kürzere. Außerdem enthalten längere Dokumente tendenziell mehr lange Plagiate als kurze Dokumente, wodurch der durchschnittliche Modifikationsgrad der Texte in längeren Dokumenten ebenfalls abfällt.

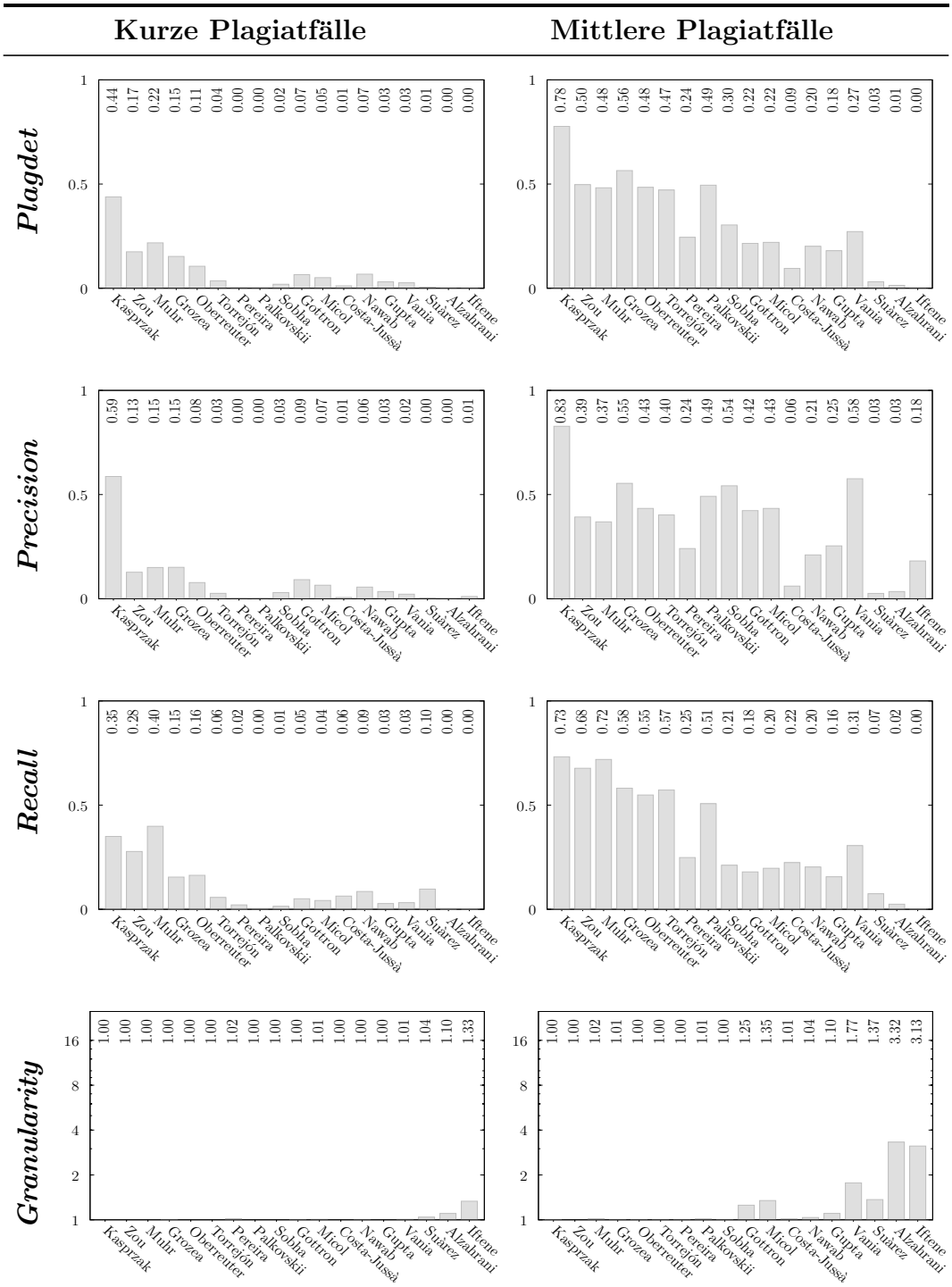
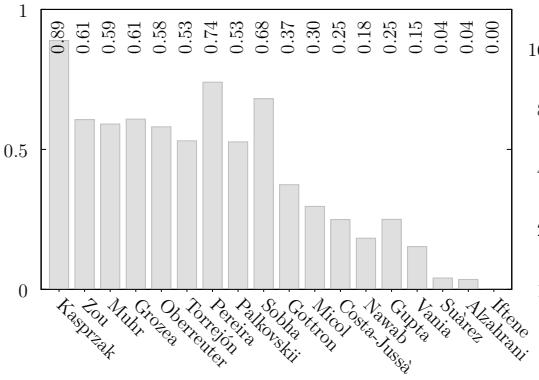


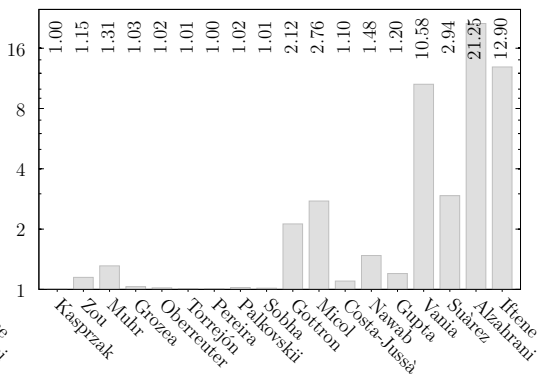
Tabelle 5.7: Erkennungsleistung für Plagiatfälle kurzer und mittlerer Länge

Lange Plagiate

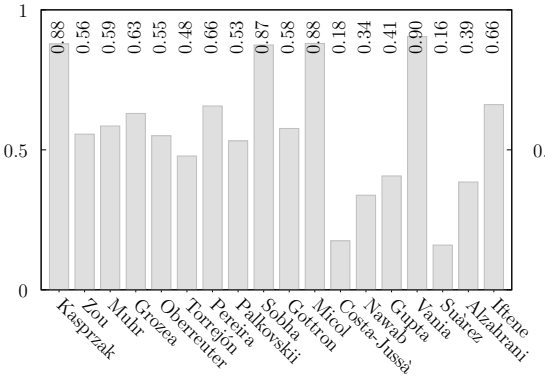
Plagdet



Granularity



Precision



Recall

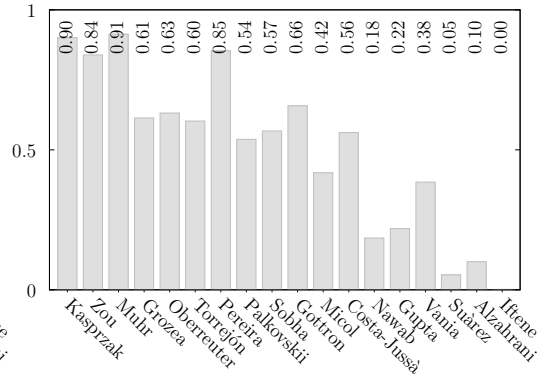


Tabelle 5.8: Erkennungsleistung für Plagiatfälle großer Länge

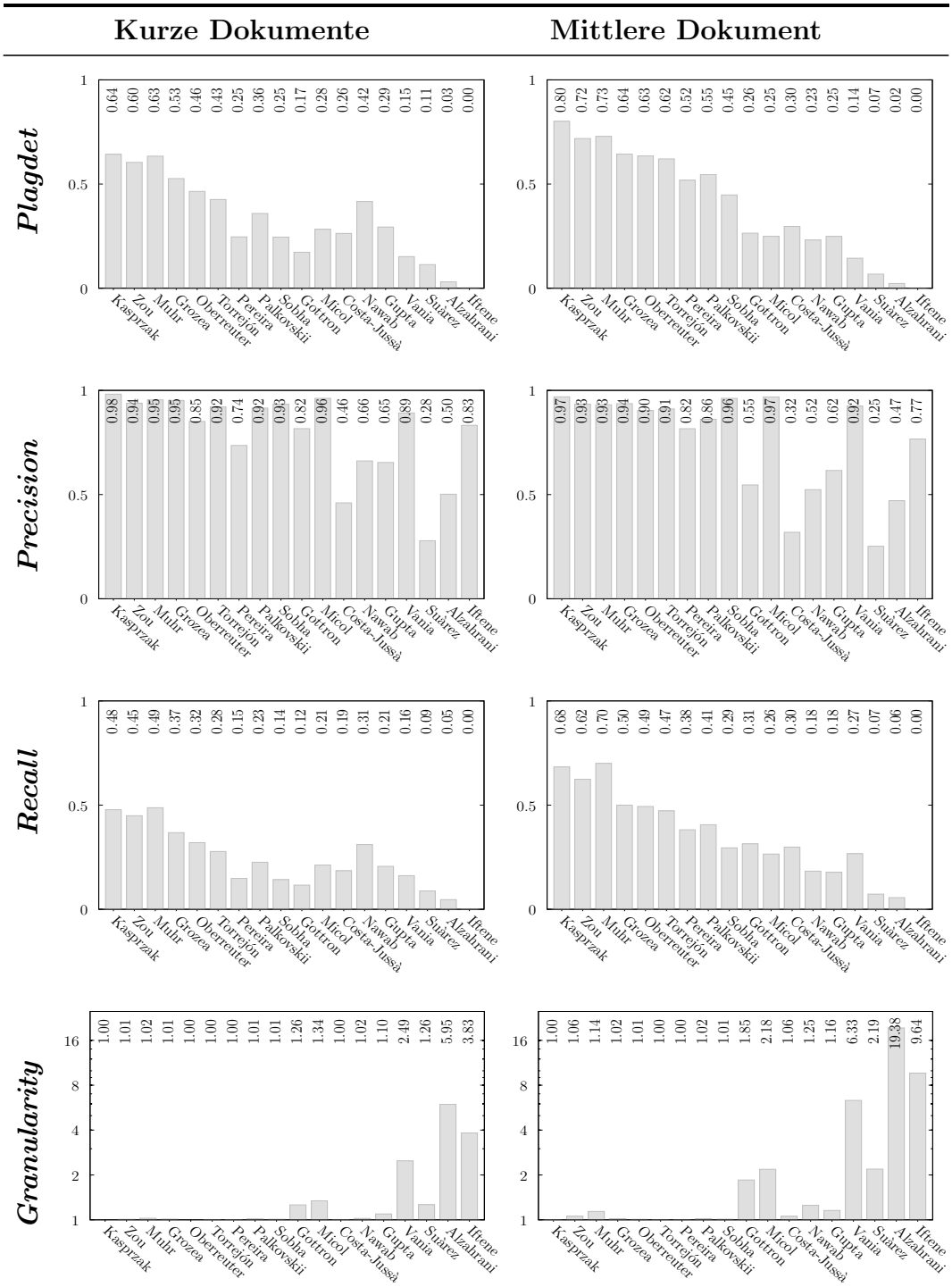
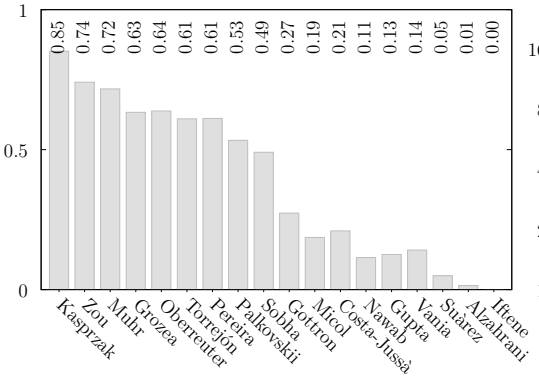


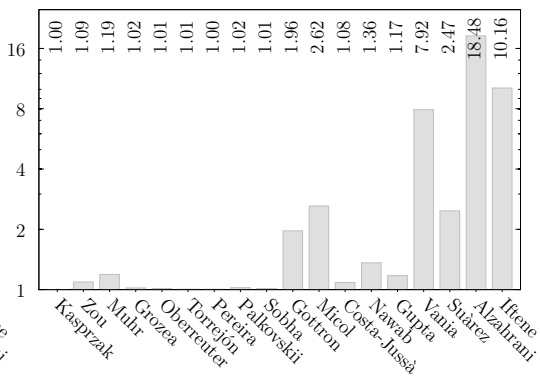
Tabelle 5.9: Erkennungsleistung für Dokumente kurzer und mittlerer Länge

Lange Dokumente

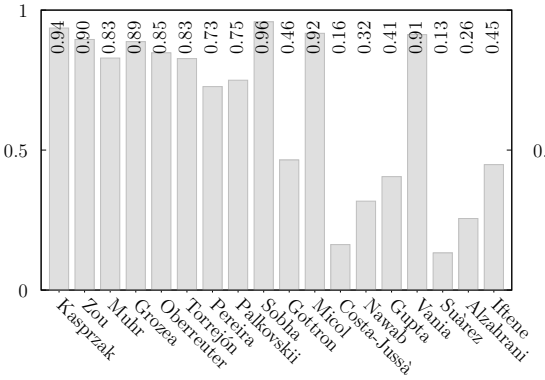
Plagdet



Granularity



Precision



Recall

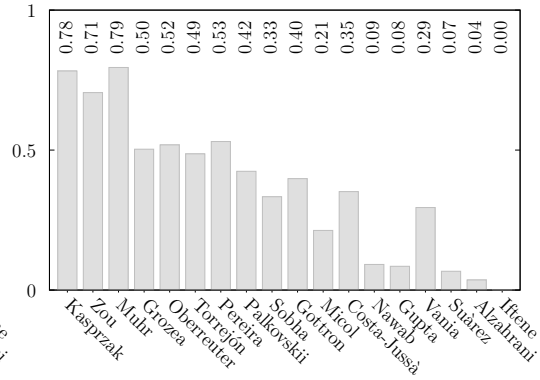


Tabelle 5.10: Erkennungsleistung für Dokumente großer Länge

Dass, wie man in Tabelle 5.11 und Tabelle 5.12 beobachten kann, die Erkennungsleistung mit steigendem Plagiatanteil in den Dokumenten deutlich zunimmt kann damit erklärt werden, dass aus Sicht eines Plagiaterkennungssystems die Wahrscheinlichkeit eines korrekt erkannten Plagiatfalls steigt, je mehr Plagiate im selben Dokument erkannt wurden.

Ob die Quelle eines Plagiats und das Dokument, in welches es eingefügt wird thematisch relevant sind, spielt offensichtlich für die meisten Systeme keine Rolle (siehe Tabelle 5.13). Lediglich das System von Gottron (2010) verzeichnete einen starken Einbruch der *Precision* bei nicht themenverwandten Dokumenten. Dieser Effekt lässt sich jedoch nicht auf die thematische Relevanz zurückführen.

Zusammenfassend kann gesagt werden, dass die größten Schwierigkeiten vor

allem bei der Erkennung von Plagiaten ohne verfügbare Quelle sowie kurzen und von Menschen umgeschriebenen Plagiaten beobachtet werden konnten. Letzterer Fall mag jedoch auch mit dem Umstand zusammenhängen, dass die Teilnehmer aufgefordert wurden, den Text so stark wie möglich zu modifizieren.

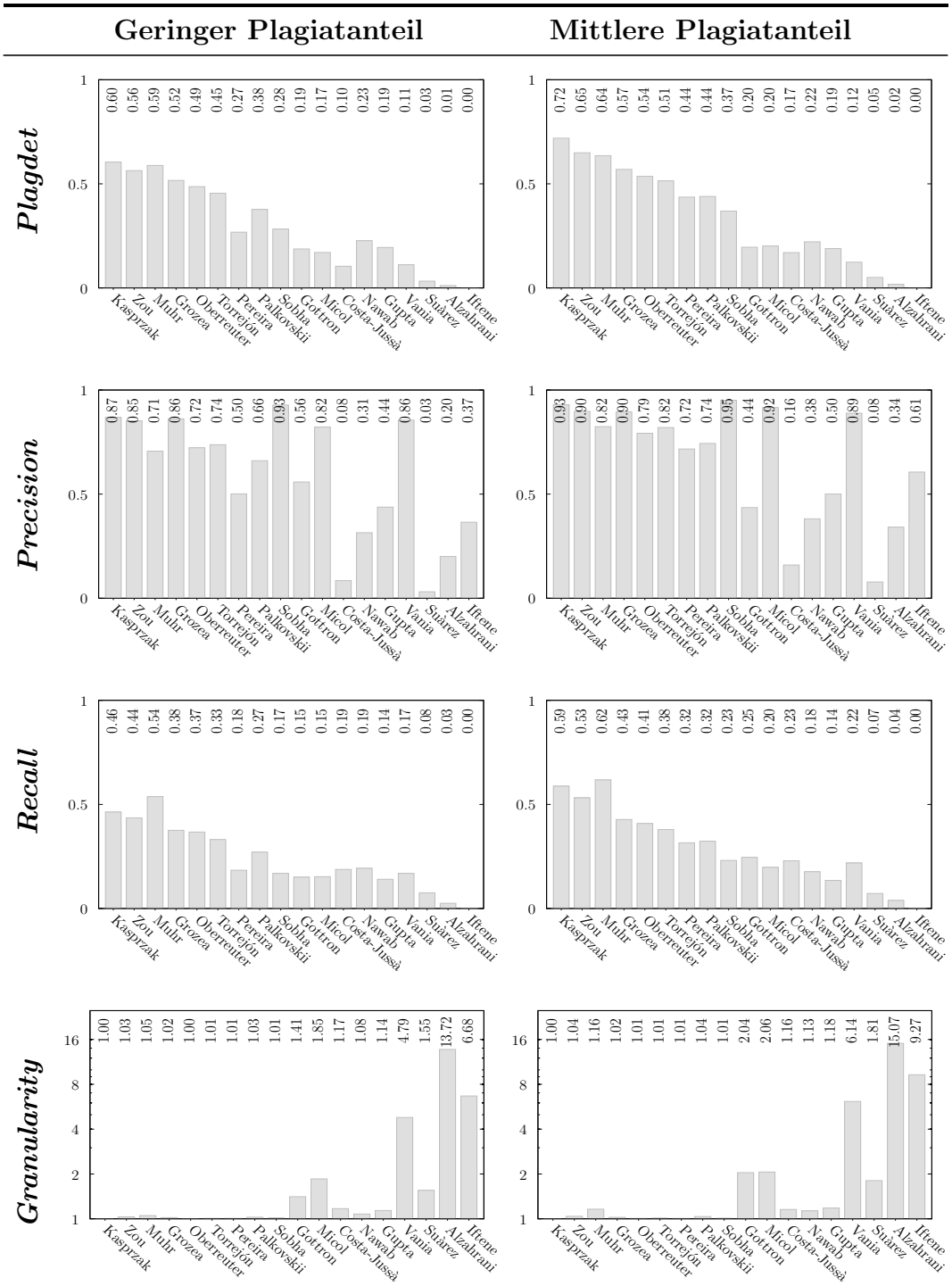


Tabelle 5.11: bei Dokumenten mit geringem (5%−20%) und mittlerem (20%−50%) Plagiatanteil

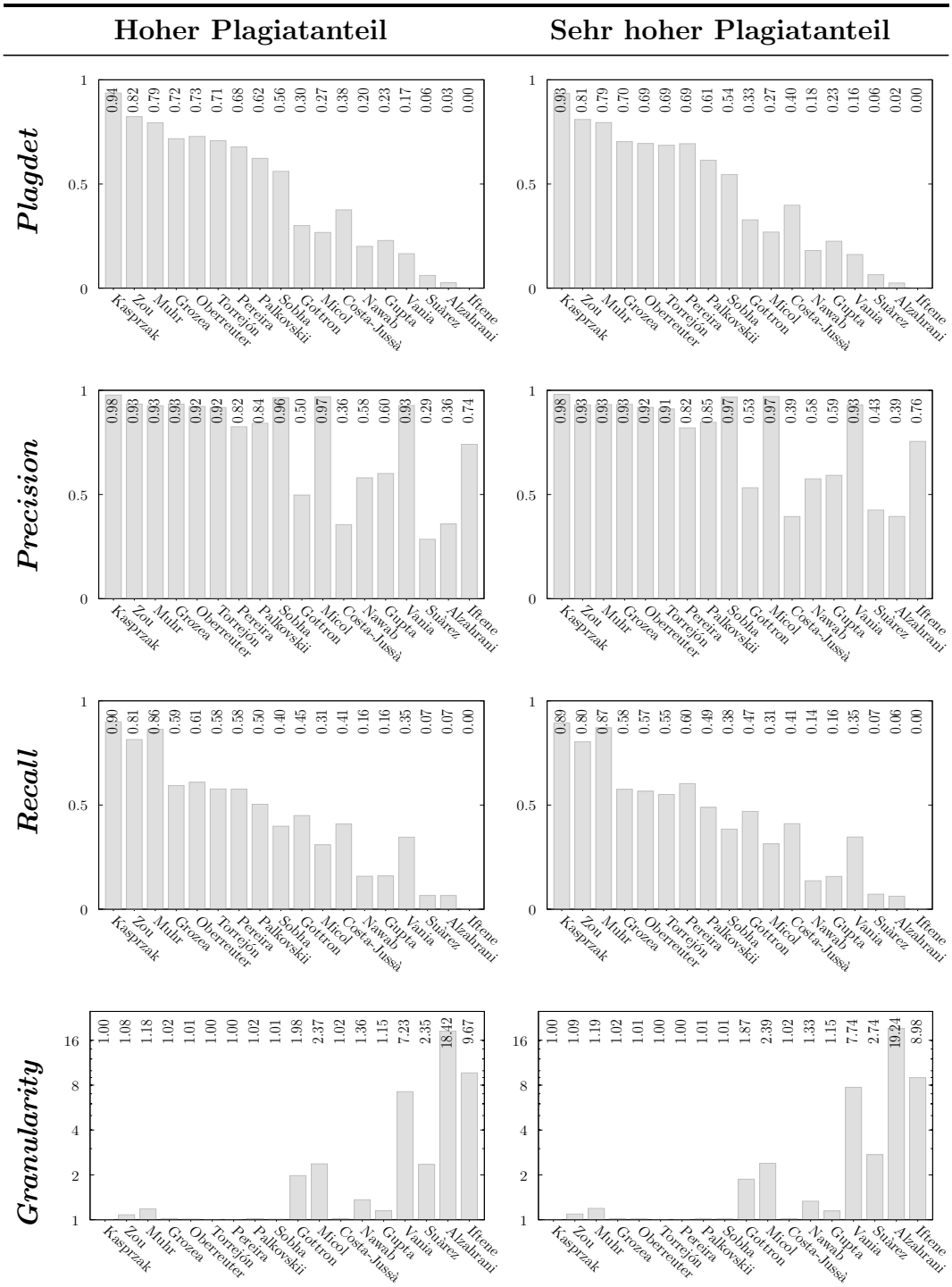


Tabelle 5.12: bei Dokumenten mit hohem (50% – 80%) und sehr hohem (> 80%) Plagiatanteil

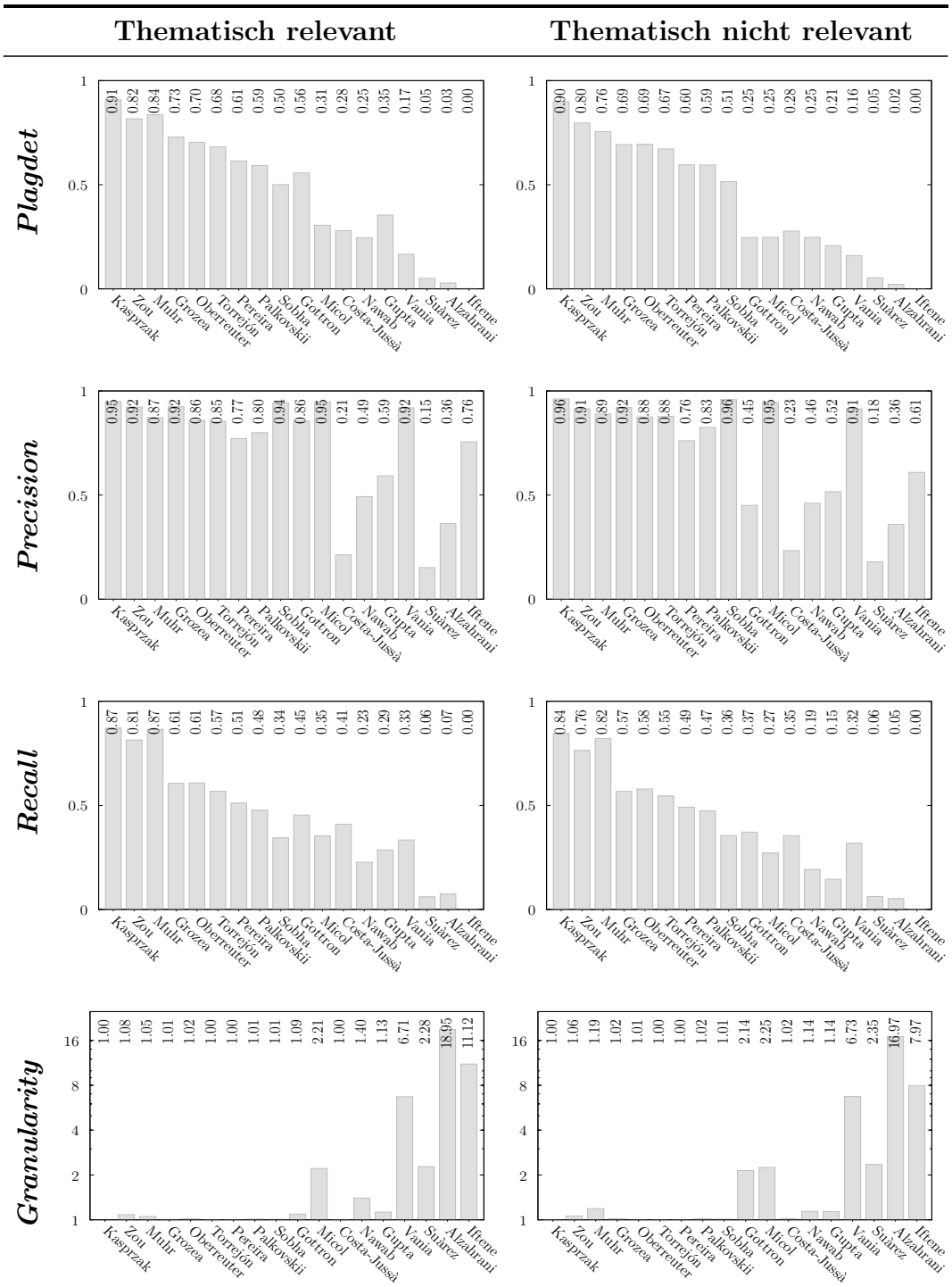


Tabelle 5.13: Erkennungsleistung für Plagiatfälle bei denen d_{src} und d_{plg} thematisch verwandt oder nicht verwandt sind

Kapitel 6

Zusammenfassung

In dieser Arbeit wird eine kontrollierte Umgebung zur Evaluierung von Plagiaterkennungsalgorithmen entwickelt. Teil dieser Umgebung sind sowohl eine Testkollektion von Plagiatfällen in Form eines Korpus, als auch Maße zur Evaluierung der durch Plagiaterkennungsalgorithmen auf der Testkollektion erzielten Ergebnisse. Sowohl bei der Planung und Erstellung des Korpus, als auch beim Entwurf der Evaluierungsmaße geht diese Arbeit über bisherige Ansätze hinaus. So ist der Korpus mit seiner Größe, der Vielfalt an enthaltenen Plagiatformen in jeweils unterschiedlichen Ausprägungen und der lizenzfreien Verfügbarkeit bisher einmalig. Auch die Evaluierungsmaße gehen über die klassischen Leistungsmaße im Information Retrieval hinaus und berücksichtigen zusätzlich plagiaterkennungsspezifische Aspekte.

Der Einsatz der Evaluierungsumgebung im Rahmen zweier internationaler Wettbewerbe zur Evaluierung von Plagiaterkennungsverfahren erlaubte es außerdem, Erfahrungen hinsichtlich der weiteren Entwicklung zu sammeln. So liegt der derzeitige Fokus des Korpus hinsichtlich externer Plagiaterkennung ausschließlich auf dem Durchsuchen von Dokumenten in einer geschlossenen lokalen Umgebung. Ein wichtiger, jedoch vernachlässigter Aspekt ist somit das Retrieval von möglichen Quelldokumenten eines Plagiats aus dem WWW. Auch die künstlich erstellten Paraphrasen- und Übersetzungsplagiate sind, wie die Auswertung der bei den Wettbewerben erzielten Ergebnisse gezeigt hat, keine Hürde für die Plagiaterkennungsalgorithmen. Hier sollte in Zukunft über die Verwendung von von Menschen übersetzten Texten, sowie Paraphrasierungen auf Basis von fortgeschritteneren Algorithmen der Computerlinguistik nachgedacht werden. Auch die Möglichkeit der Erkennung von korrekten und fehlerhaften Zitaten sollte im Korpus durch entsprechende Fälle berücksichtigt werden. Bezüglich der Evaluierungsmaße wurde außerdem angeregt, eine höhere Gewichtung des *Recalls* gegenüber der *Precision* vorzunehmen. Für die Plagiaterkennung besitzt dieser ungleich mehr Bedeutung, da davon auszu-

gehen ist, dass sich der Benutzer die gefundenen Ursprungsdokumente eines Plagiats ohnehin selbst genauer anschauen wird.

Die Entwicklung einer Evaluierungsumgebung, die Schlüsse über die Leistungsfähigkeit von Plagiaterkennungsalgorithmen in realen Szenarien zulässt, steht also immer noch relativ am Anfang. Das Ergebnis dieser Arbeit ist jedoch ein erster Schritt in diese Richtung.

Anhang A

Danksagung

Mein besonderer Dank gilt zunächst meinem Betreuer Martin Potthast, der mich während der gesamten Entstehungsphase des Projekts und dieser Arbeit, betreute und stets mit vielen hilfreichen fachlichen und didaktischen Hinweisen zur Seite stand. Darüber hinaus gilt ein besonderer Dank Alberto, Laura, Silvia & Christian, die mir vor allem während meiner Zeit in Spanien sowohl fachlich als auch persönlich stets zur Seite standen. Auch Paolo Rosso und Prof. Benno Stein sei an dieser Stelle für ihre Unterstützung und für das in mich gesetzte Vertrauen sowie die Möglichkeit an diesem Projekt mitarbeiten zu können gedankt. Nicht zuletzt möchte ich mich auch bei meiner Familie und meinen Freunden (Verena, Peter, Cristina, Teresa, Johanna, Elba und Sebastian) und allen anderen bedanken, die mich auf dem Weg zum Abschluss dieser Arbeit begleitet und unterstützt haben.

Abbildungsverzeichnis

2.1 Taxonomie von Plagiatvergehen in Verbindung mit entsprechenden Erkennungsmethoden (nach Meyer Zu Eissen/Stein/Kulig (2007)) 5
2.2 Schematische Darstellung des Prozesses der externen Plagiatanalyse (für die Bedeutung der Variablen, siehe Text) (nach Potthast et al. (2009)) 6

3.1 Schematische Darstellung des Vorgangs der Plagiaterstellung. Abschnitt s_{rpl} in d_{plg} wird durch s_{plg} ersetzt. s_{plg} wird mit Hilfe von Textoperationen von s_{src} abgeleitet, d.h. s_{plg} ist eine Übersetzung, Umformulierung oder Wort-für-Wort-Kopie von s_{src}. . . 12
3.2 Korpuspipeline 29
3.3 Schematische Darstellung der einzelnen Etappen der Vorverarbeitung als Teil der Konstruktionspipeline 29
3.4 Schematische Darstellung der Vorverarbeitung als Teil der Konstruktionspipeline 31

4.1 Schematische Darstellung eines Plagiatfalls 34
4.2 Illustrierung eines Plagiaterkennungsfalls. In diesem Beispiel werden s_{plg} und s_{src} durch r_{plg} und r_{src} zwar erkannt, allerdings stimmen die Grenzen nicht überein. 35

Tabellenverzeichnis

3.1 Probleme und entsprechende Lösungsansätze bei der Verwendung der unterschiedlichen Plagiatformen (siehe dazu auch Abschnitt 3.2). 10

3.2 Liste der verschiedenen Plagiatformen, welche für den Korpus künstlich (k) und/oder simuliert (s) erstellt wurden. Eine Markierung durch einen Stern bedeutet, dass diese Plagiatform nicht explizit erstellt wurde, sondern in ihrem Vorkommen eine Unterklasse einer anderen Plagiatform bildet. 11

3.3 Beispiel eines Übersetzungsplagiats. Der Originaltext stammt aus dem Buch „Mein Leben Und Streben“ von Karl May und wurde mit Hilfe von *Google Translate* ins Englische übersetzt. . 13

3.4 Ein Textausschnitt mit den zugehörigen Wortarten (part-of-speech); Die PoS-Tags basieren auf dem *Penn Treebank Tag Set* (Marcus/Marcinkiewicz/Santorini, 1993). *NN* = Substantiv; *VBG* = Verb; *BEZ* = „is“; *DT* = Determinativ; *JJ* = Adjektiv; *IN* = Präposition. 15

3.5 Die Wahrscheinlichkeit des Vorkommens aller möglichen PoS-Tag-Trigramme, die durch Einfügen von *JJ* („illegal“) entstehen können. 16

3.6 Die Wahrscheinlichkeit des Vorkommens aller möglichen PoS-Tag-Trigramme, die durch Löschen eines Wortes entstehen können. Der Punkt muss am Beginn des Textes stehen, um auch die Wahrscheinlichkeit eines alternativen Satzbeginns berechnen zu können. 16

3.7 Beispiel für das Ergebnis des automatisierten Löschens/Einfügens/Ersetztens von Wörtern. Kursiv gedruckte Wörter in s_{src} wurden gelöscht. Kursiv gedruckte Wörter in s_{plg} wurden entweder ersetzt oder eingefügt. 17

3.8 Beispiel für das Ergebnis des automatisierten Ersetzens von Wörtern durch semantisch verwandte Wörter unter Verwendung des *WordNets*. Kursiv gedruckte Wörter wurden im Vergleich zu s_{src} verändert. 17
3.9 Beispiel für das Ergebnis des automatisierten Vertauschens von Wörtern unter Beibehaltung der syntaktischen Struktur. Kursiv gedruckte Wörter wurden im Vergleich zu s_{src} vertauscht. 18
3.10 Beispiel für das Ergebnis des automatisierten Vertauschens von Phrasen. Kursiv gedruckte Abschnitte wurden im Vergleich zu s_{src} vertauscht. 18
3.11 Statistik der 4000 auf *Amazons Mechanical Turk* bearbeiteten Aufgaben 20
3.12 Überblick zu wesentlichen Merkmalen der 907 beteiligten Personen 20
3.13 Beispiel eines auf der Plattform von *Amazons Mechanical Turk* erstellten simulierten Plagiatfalls s. Der Ursprungstext ist ein Auszug aus „Abraham Lincoln; A History“ von Hay/Nicolay (2009). 22
3.14 Experimentell bestimmte Spanne der n-Gramm-Distanz für die Kategorien des Grades der Textmodifikation (niedrig/hoch) in Abhängigkeit von der Länge des Plagiatfalls. 23
3.15 Dokument- und Plagiatfallstatistik: gibt den prozentualen Anteil der entsprechenden Dokument-/Plagiatfallkategorie an der Gesamtmenge der Dokumente/Plagiatfälle an. 24
3.16 Häufigkeitsverteilung der verschiedenen Plagiatformen in D_{plg}. . 25
3.17 Liste der Attribute bei der Annotation von Plagiatfällen 27
3.18 Liste der Möglichen Attribute bei der Annotation von Metadaten des Textdokuments 28
3.19 Statistik über die Verteilung der Dokumente über die verwendeten Sprachen 29
3.20 Abbildung der Korpusstruktur auf das Dateisystem; Test-Set: die XML-Dateien enthalten keine Plagiatannotation, sondern dienen nur als Platzhalter. 32
3.21 Abbildung der Korpusstruktur auf das Dateisystem; Evaluierungs-Set: hier werden nur die XML-Dateien mit den entsprechenden Plagiatannotation benötigt. 32

5.1 Erkennungsleistung für den gesamten Korpus 39
5.2 Erkennungsleistung der externen Plagiaterkennung 40
5.3 Erkennungsleistung der intrinsischen Plagiaterkennung 41
5.4 Erkennungsleistung für künstliche Wort-für-Wort- und simulierte Plagiate 43

5.5 Erkennungsleistung für künstliche Paraphrasenplagiate mit niedrigem und hohem Modifikationsgrad 44
5.6 Erkennungsleistung für Übersetzungsplagiaten 45
5.7 Erkennungsleistung für Plagiatfälle kurzer und mittlerer Länge . 46
5.8 Erkennungsleistung für Plagiatfälle großer Länge 47
5.9 Erkennungsleistung für Dokumente kurzer und mittlerer Länge . 48
5.10 Erkennungsleistung für Dokumente großer Länge 49
5.11 bei Dokumenten mit geringem (5%−20%) und mittlerem (20%−50%) Plagiatanteil 51
5.12 bei Dokumenten mit hohem (50% − 80%) und sehr hohem (> 80%) Plagiatanteil 52
5.13 Erkennungsleistung für Plagiatfälle bei denen d_{src} und d_{plg} thematisch verwandt oder nicht verwandt sind 53

Literaturverzeichnis

Alonso, O./Mizzaro, S. (2009): Can we get rid of TREC assessors? Using Mechanical Turk for relevance assessment. In Proceedings of the SIGIR 2009 Workshop on the Future of IR Evaluation. Citeseer, 15–16

Amancio, Diego R./Antiqueira, Lucas/Pardo, Thiago A. S./Costa, Luciano Da F./Oliveira, Osvaldo N./Nunes, Maria G. V. (2008): Complex Networks Analysis of Manual and Machine Translations. International Journal of Modern Physics C, 19 Nr. 04, 583 ⟨URL: `http://www.worldscinet.com/ijmpc/19/1904/S0129183108012285.html`⟩, ISSN 0129–1831

Ambati, Vamshi/Vogel, Stephan/Carbonell, Jaime (2010): Active learning and crowd-sourcing for machine translation. In **Calzolari, Nicoletta/Choukri, Khalid/Maegaard, Bente/Mariani, Joseph/Odijk, Jan/Piperidis, Stelios/Rosner, Mike/Tapias, Daniel (Hrsg.):** Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10), Valletta, Malta, may. European Language Resources Association (ELRA). Citeseer

Androutsopoulos, Ion/Malakasiotis, Prodromos (2010): A Survey of Paraphrasing and Textual Entailment Methods. Artificial Intelligence, 38, 135–187

Armstrong, Timothy G./Moffat, Alistair/Webber, William/Zobel, Justin (2009): Improvements that don't add up. Proceeding of the 18th ACM conference on Information and knowledge management - CIKM '09,, 601–610 ⟨URL: `http://portal.acm.org/citation.cfm?doid=1645953.1646031`⟩, ISBN 9781605585123

Basili, Victor/Shull, Forrest/Lanubile, F. (2000): Using experiments to build a body of knowledge. In Perspectives of System Informatics. Springer ⟨URL: `http://www.springerlink.com/index/A87PXECV6FDNWRKK.pdf`⟩, 265–282

Bowers, William J. (1964): Student Dishonesty and its Control in College. New York: Columbia University, New York Bureau of Applied Social Research , 291 Seiten

Clough, Paul; Bently, Lionel/Davis, Jennifer/Ginsburg, Jane C. (Hrsg.) (2010): Kap. 12. InMeasuring Text Re-Use in the News Industry. Cambridge, UK: Cambridge University Press, 480, ISBN 0521193435

Clough, Paul/Gaizauskas, Robert/Piao, Scott (2002): Building and annotating a corpus for the study of journalistic text reuse. LREC,, 1678–1691

Duden (2003): Duden. Das große Fremdwörterbuch. Herkunft und Bedeutung der Fremdwörter. 85.000 Stichwörter. Auflage: 3., überarb. A. Auflage. Bibliographisches Institut, Mannheim, 1542, ISBN 3–411–04163–3

Google (2010): Google Translate API. ⟨URL: `http://code.google.com/intl/de-DE/apis/language/translate/overview.html`⟩

Gottron, Thomas (2010): External Plagiarism Detection Based on Standard IR Technology and Fast Recognition of Common Subsequences. In **Braschler, Martin/Harman, Donna/Pianta, Emanuele (Hrsg.):** Lab Report for PAN at CLEF 2010. Padua

Goutte, Cyril/Cancedda, Nicola/Dymetman, Marc (2009): Learning machine translation. MIT Press, 316, ISBN 0262072971

Grozea, Cristian/Popescu, Marius (2010): Encoplot – Performance in the Second International Plagiarism Detection Challenge. In **Braschler, Martin/Harman, Donna/Pianta, Emanuele (Hrsg.):** Lab Report for PAN at CLEF 2010. Padua, 3–6

Gulli, A./Signorini, A. (2005): The indexable web is more than 11.5 billion pages. In WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web. New York, NY, USA: ACM, ISBN 1–59593–051–5, 902–903

Hay, John/Nicolay, John George (2009): Abraham Lincoln; A History. BiblioBazaar, 526, ISBN 1115266942

Kasprzak, Jan/Brandejs, Michal (2010): Improving the Reliability of the Plagiarism Detection System. In **Braschler, Martin/Harman, Donna/Pianta, Emanuele (Hrsg.):** Lab Report for PAN at CLEF 2010. Padua

Kittur, A./Chi, E.H./Suh, B. (2008): Crowdsourcing user studies with Mechanical Turk. In Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems. ACM, 453–456

Kondrak, Grzegorz (2005): N-gram similarity and distance. In String Processing and Information Retrieval. Springer, 115–126

Kunder, Maurice De (2007): Geschatte grootte van het geïndexeerde World Wide Web., 1–68 ⟨URL: `http://www.worldwidewebsize.com`⟩

Leech, G.; Wynne, Martin (Hrsg.) (2005): Adding Linguistic Annotation. Oxford ⟨URL: `http://ahds.ac.uk/linguistic-corpora/`⟩, 17–29

Madnani, Nitin/Dorr, Bonnie J (2010): Generating Phrasal and Sentential Paraphrases : A Survey of Data-Driven Methods. Computational Linguistics Nr. November 2009

Marcus, M.P./Marcinkiewicz, M.A./Santorini, B. (1993): Building a large annotated corpus of English: The Penn Treebank. Computational linguistics, 19 Nr. 2, 330, ISSN 0891–2017

Mason, O. (1997): QTAG: a portable probabilistic tagger. Corpus Research, The University of Birmingham, UK

Mason, Winter/Watts, Duncan J. (2009): Financial incentives and the performance of crowds. In Proceedings of the ACM SIGKDD Workshop on Human Computation. New York, NY, USA: ACM, HCOMP '09, ISBN 978–1–60558–672–4, 77–85

May, Karl (2006): Mein Leben Und Streben. Echo Library, 184, ISBN 1406823082

McCabe, Donald L. (2005): Cheating among college and university students : A North American perspective. International Journal for Educational Integrity, 1 Nr. 1 (2005), 1–11

Meyer Zu Eissen, Sven/Stein, Benno/Kulig, Marion (2007): Plagiarism Detection Without Reference Collections. In **Decker, Reinhold/ Lenz, Hans J. (Hrsg.):** GfKl: 30th Annual Conference of the German

Classification Society. Berlin, Heidelberg: Springer, Studies in Classification, Data Analysis, and Knowledge Organization ⟨URL: `http://www.springerlink.com/index/10.1007/978-3-540-70981-7`⟩, ISBN 978–3–540–70980–0, 359–366

Miller, G.A. (1995): WordNet: a lexical database for English. Communications of the ACM, 38 Nr. 11, 39–41, ISSN 0001–0782

Muhr, Markus/Kern, Roman/Zechner, Mario/Granitzer, Michael (2010): External and Intrinsic Plagiarism Detection using a Cross-Lingual Retrieval and Segmentation System. In **Braschler, Martin/Harman, Donna/Pianta, Emanuele (Hrsg.):** Lab Report for PAN at CLEF 2010. Padua

Nitterhouse, Denise (2003): Plagiarism – Not Just an "Academic" Problem. Teaching Business Ethics, 7, 215–227, ISSN 1382–6891

Papineni, K./Roukos, S./Ward, T./Zhu, W.J. (2002): BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, 311–318

Potthast, Martin (2010): Crowdsourcing a Wikipedia Vandalism Corpus. In **Chen, Hsin-Hsi/Efthimiadis, Efthimis N./Savoy, Jaques/Crestani, Fabio/Marchand-Maillet, Stéphane (Hrsg.):** 33rd Annual International ACM SIGIR Conference. ACM, ISBN 978–1–4503–0153–4, 789–790

Potthast, Martin/Barrón-Cedeño, Alberto/Eiselt, Andreas/Stein, Benno/Rosso, Paolo (2010a): Overview of the 2nd International Competition on Plagiarism Detection. In **Braschler, Martin/Harman, Donna (Hrsg.):** Notebook Papers of CLEF 2010 LABs and Workshops, 22-23 September, Padua, Italy. , ISBN 978–88–904810–0–0, 107

Potthast, Martin/Stein, Benno/Barrón-Cedeño, Alberto/Rosso, Paolo (2010b): An Evaluation Framework for Plagiarism Detection. In Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010). Beijing, China: Association for Computational Linguistics

Potthast, Martin/Stein, Benno/Eiselt, Andreas/Barrón-Cedeño, Alberto/Rosso, Paolo (2009): Overview of the 1st International Competition on Plagiarism Detection. In **Koppel, Moshe/Rosso, Paolo/Stamatatos, Efstathios/Agirre, Eneko/Stein, Benno (Hrsg.):**

PAN-09 3rd Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse and 1st International Competition on Plagiarism Detection. San Sebastian (Donostia): CEUR-WS.org ⟨URL: `http://ceur-ws.org/Vol-502`⟩, ISSN 1613–0073, 1–9

Rieble, Volker (2010): Das Wissenschaftsplagiat. Vom Versagen eines Systems. Frankfurt am Main: Klostermann, 120, ISBN 978–3–465–04101–6

Schilling, Peter (2006): Plagiarismus. Jeder dritte Drehbuchautor ist betroffen. Potsdam: PriorMart AG , 15 Seiten

Stein, Benno/Lipka, Nedim/Prettenhofer, Peter (2010): Intrinsic Plagiarism Analysis. Language Resources and Evaluation (LRE) (to appear)

Stein, Benno/Meyer Zu Eissen, Sven/Potthast, Martin; Vries, Arjen de/Clarke, Norbert Fuhr Charles (Hrsg.) (2007): Strategies for retrieving plagiarized documents. New York, New York, USA: ACM Press, 825–826, ISBN 978–1–59593–597–7

Suárez, Pablo/González, José Carlos/Villena-Román, Julio (2010): A plagiarism detector for intrinsic plagiarism. In **Braschler, Martin/Harman, Donna/Pianta, Emanuele (Hrsg.):** Lab Report for PAN at CLEF 2010. Padua, 4–7

Taycher, Leonid (2010): Books of the world, stand up and be counted! All 129,864,880 of you. ⟨URL: `http://booksearch.blogspot.com/2010/08/books-of-world-stand-up-and-be-counted.html`⟩

Unesco (1998): UNESCO Statistical Yearbook. Unesco, 800, ISBN 978–0890591567

Zou, Du/Long, Wei-jiang/Ling, Zhang (2010): A Cluster-Based Plagiarism Detection Method. In **Braschler, Martin/Harman, Donna/Pianta, Emanuele (Hrsg.):** Lab Report for PAN at CLEF 2010. Padua