SCM
Studies in Communication | Media

FULL PAPER

Vorsprung durch Technik? Die Analyse journalistischer Online-Angebote mit Hilfe automatisierter Verfahren

Advancement through technology? The analysis of journalistic online-content by using automated tools

Jörg Haßler, Marcus Maurer & Thomas Holbach

SCM, 3. Jg., 2/2014, S. 180-204

Korrespondierender Autor:

Jörg Haßler, Institut für Kommunikationswissenschaft, Friedrich-Schiller-Universität Jena, Ernst-Abbe-Platz 8, 07743 Jena, joerg.hassler(at)uni-jena.de

FULL PAPER

Vorsprung durch Technik? Die Analyse journalistischer Online-Angebote mit Hilfe automatisierter Verfahren

Advancement through technology? The analysis of journalistic online-content by using automated tools

Jörg Haßler, Marcus Maurer & Thomas Holbach

Zusammenfassung: Mit zunehmender Bedeutung des Internets als Kommunikationsmedium gewinnen auch Inhaltsanalysen von Online-Medien an Bedeutung. Dabei ergeben sich Chancen, z. B. durch die Digitalisierung und Maschinenlesbarkeit oder die Verfügbarkeit von Meta-Daten, aber auch Herausforderungen. Hierzu gehören die Flüchtigkeit und Dynamik, die Multimedialität, die Hypertextualität sowie die Reaktivität und Personalisierung von Websites. Der vorliegende Beitrag diskutiert zunächst die Vor- und Nachteile gängiger Vorschläge zur Lösung dieser Probleme und gibt dann einen Überblick über neue Verfahren zur automatisierten Speicherung von journalistischen Online-Angeboten.

Schlagwörter: Inhaltsanalyse, Online-Medien, Meta-Daten, Multimedialität, Datenbank

Abstract: Because of the growing importance of the internet as a communication channel, content analyses of websites are becoming more and more central to the field of communication research. Online content offers opportunities, such as digitalization and the availability of meta-data as well as challenges. These challenges are, e.g., the dynamics, multimediality, hypertextuality and personalization of websites. This article discusses frequently used strategies to cope with those challenges. Furthermore, the article presents five recently developed tools for automatic storage of journalistic online-content.

Keywords: Content analysis, online media, meta-data, multimediality, database

1. Einleitung[1]

Das Internet hat in den vergangenen Jahren für den Journalismus und für das Medienpublikum erheblich an Bedeutung gewonnen. Alle traditionellen Medienmarken sind seit geraumer Zeit auch mit einem eigenen Online-Angebot vertreten. Das Nachrichtenmagazin Der Spiegel startete schon 1994 eine eigene Website. Die Bild-Zeitung folgte zwei Jahre später. Auch auf Seiten der Rezipienten

1 Diese Publikation entstand im Rahmen der von der Deutschen Forschungsgemeinschaft (DFG) geförderten Forschergruppe „Politische Kommunikation in der Online-Welt“ (1381), Teilprojekt 4.

hat das Internet mittlerweile einen großen Stellenwert im Medienrepertoire. Rund drei Viertel der Deutschen sind online, 72 Prozent der Internetnutzer geben an, Informationen im Internet gezielt zu suchen, und 55 Prozent nutzen online aktuelle Nachrichten (van Eimeren & Frees, 2013, S. 363). Journalistische Nachrichtenangebote zählen hierbei nach wie vor zu den wichtigsten Informationsquellen (Hasebrink & Schmidt, 2013, S. 8).

Mit wachsender Bedeutung des Internets für Kommunikatoren und Rezipienten steigt auch die Notwendigkeit für die Kommunikationswissenschaft, Medieninhalte online zu erfassen und zu analysieren. Medieninhaltsanalysen berücksichtigen in der Regel besonders reichweitenstarke Medien, entweder weil sie diese als repräsentativ für das gesamte Mediensystem betrachten (diagnostischer Ansatz) oder weil die Inhaltsanalysen Grundlagen für Wirkungsanalysen sein sollen (prognostischer Ansatz) und deshalb die Medien ausgewählt werden müssen, die von vielen Befragten genutzt werden. Deshalb spricht schon heute vieles dafür, journalistische Online-Angebote in Inhaltsanalysen zu berücksichtigen. Allerdings sind Analysen von Online-Inhalten auch mit erheblichen Herausforderungen verbunden, die sich beispielsweise aus der Flüchtigkeit, der Multimedialität und der Hypertextualität der Inhalte ergeben. Diese Herausforderungen betreffen nicht nur den Codiervorgang, sondern insbesondere auch den Prozess der Speicherung und Bereitstellung der Inhalte für die Codierung. Wir wollen in diesem Beitrag zunächst die Chancen und Herausforderungen der Inhaltsanalysen von journalistischen Online-Angeboten sowie eine Reihe gängiger, aber ebenfalls problembehafteter Lösungsvorschläge diskutieren. Anschließend vergleichen wir fünf online frei verfügbare Tools, die in den letzten Jahren entwickelt wurden, um verschiedene Arbeitsschritte der Inhaltsanalyse von Online-Medien automatisiert durchzuführen. Dabei konzentrieren wir uns auf datenbankgestützte Tools, die Medienbeiträge automatisiert abspeichern und/oder als Codierplattform dienen. Obwohl einige dieser Tools zugleich auch automatisierte Codierungen vornehmen, steht diese Funktion hier nicht im Vordergrund (für einen Überblick über die Tools zur automatisierten Codierung vgl. Scharkow, 2012).

2. Inhaltsanalysen von Online-Medien: Chancen und Herausforderungen

Welker et al. (2010) unterscheiden sechs Merkmale von Online-Medien, die bei Inhaltsanalysen in Betracht gezogen werden müssen. In der Reihenfolge, in der sie bei der Konzeptionierung von Inhaltsanalysen relevant werden, sind dies 1) die Quantität, 2) die Flüchtigkeit, Dynamik und Transitorik, 3) die Digitalisierung und Maschinenlesbarkeit, 4) die Medialität, Multimedialität und Multimodalität, 5) die Nonlinearität und Hypertextualität und 6) die Reaktivität und Personalisierung. Wir wollen diese um eine siebte Eigenschaft ergänzen, die zuletzt erheblich an Bedeutung gewonnen hat, die Verfügbarkeit von Meta-Daten, also z. B. Informationen darüber, wie häufig ein Beitrag in sozialen Netzwerken weitergeleitet oder bewertet wurde:

Quantität: Die Anzahl der weltweit verfügbaren Websites liegt geschätzt zwischen 634 Millionen (royal.pingdom.com, 2013) und 3,94 Billionen (de Kunder, 2013). Zur Auswahl von zu analysierenden Angeboten bieten sich zwei Vorge-

hensweisen an. *Erstens*, eine systematische Identifikation journalistischer Angebote, wie sie z. B. von Neuberger, Nuernbergk und Rischke (2009) beschrieben wurde, und *zweitens* die Orientierung an besonders reichweitenstarken Angeboten. Die zweite Vorgehensweise entspricht im Wesentlichen dem Vorgehen bei herkömmlichen Inhaltsanalysen außerhalb des Internets. Dabei werden anstelle der Auflagenzahlen die Reichweitenzahlen der Arbeitsgemeinschaft Online-Forschung e. V. herangezogen. Grundannahme hierbei ist, dass spiegelbildlich zum Offline-Medienmarkt auch online Meinungsführermedien existieren, und man sich darüber hinaus am publizistischen Spektrum orientieren kann, um journalistische Inhalte umfänglich zu erfassen. Dies hat sich für Offline-Inhaltsanalysen als praktikabelste Lösung herausgestellt, da dort aus forschungsökonomischen Gründen auf Long-Tail-Analysen in Form spezialisierter Fachzeitschriften und Verbandszeitungen verzichtet wird.

Flüchtigkeit, Dynamik und Transitorik: Online-Inhalte verändern sich kontinuierlich. Dies trifft sowohl auf einzelne Beiträge zu, die nachträglich editiert werden können, als auch auf ganze Webangebote, bei denen sich die Platzierung der Beiträge kontinuierlich ändert (Karlsson & Strömbäck, 2010). Aus praktischen Gründen und Gründen der intersubjektiven Nachprüfbarkeit müssen die zu untersuchenden Websites deshalb vor der Analyse archiviert werden. Da die Archive der Anbieter von Websites lediglich einzelne Artikel, nicht aber deren redaktionelle Einbettung speichern und bestehende Web-Archive wie *archive.org* äußerst lückenhaft sind, ist es für die meisten Inhaltsanalysen unumgänglich, eigene Archivierungstools zu entwickeln oder bestehende Software zu nutzen (Neuberger, Nuernbergk, & Rischke, 2009). Entscheidender Anknüpfungspunkt bei der Archivierung von Websites ist, dass diese digital und maschinenlesbar vorliegen.

Digitalisierung und Maschinenlesbarkeit: Die Tatsache, dass Websites digitalisiert und maschinenlesbar vorliegen, eröffnet Online-Inhaltsanalysen eine Reihe von Möglichkeiten. Inhaltsanalysedaten können vollautomatisiert erhoben werden. Da Websites standardisierte Programmiersprachen und -codes verwenden, lassen sich so prinzipiell riesige Textmengen (Big Data) erfassen (Scharkow, 2012). Dies ermöglicht zudem schnelle und kostengünstige Analysen, weil keine menschlichen Codierer benötigt werden. Obwohl seit geraumer Zeit an Programmen zur automatischen Codierung von Online-Inhalten gearbeitet wird, sind diesen Verfahren bis heute allerdings enge Grenzen gesetzt. Zwar können mit Hilfe von Worterkennungsprogrammen z. B. Berichterstattungsthemen (z. B. King & Lowe, 2003) identifiziert oder mit Hilfe von Grammatikerkennungsprogrammen (Parsern) sogar wiederkehrende Satzstrukturen inhaltlich erfasst werden (z. B. de Nooy & Kleinnijenhuis, 2013). Komplexere Analysen, z. B. Bewertungstendenzen oder Argumentationsstrategien auf Beitragsebene, müssen jedoch nach wie vor manuell durchgeführt werden (Lewis, Zamith, & Hermida, 2013).

Medialität, Multimedialität und Multimodalität von Inhalten: Websites setzen sich nicht nur aus Texten, sondern auch aus Bildern, Videos, Audiodateien und interaktiven Elementen zusammen. Dies erschwert die Inhaltsanalyse von Webinhalten aus drei Gründen (z. B. Sjøvaag, Moe, & Stavelin, 2012): Zunächst muss bei der Konzipierung von Codebüchern genau definiert werden, welche Elemente einer Website untersucht werden sollen. Hier stellt sich insbesondere die Frage, ob

die inhaltliche Codierung auf den zentralen Textartikel der Website beschränkt bleiben soll oder ob auch multimediale Elemente mit erfasst werden. Im zweiten Fall muss dann entschieden werden, ob diese als eigene Beiträge oder als Teil des Textartikels behandelt werden sollen. Dabei muss zudem zwischen multimedialen Elementen, die inhaltlich zu einem bestimmten Text gehören, und solchen, die bei allen Textartikeln eingebunden sind (z. B. Streams der aktuellen Ausgaben einer Fernsehnachrichtensendung auf deren Website), unterschieden werden. Sollen auch die multimedialen Elemente erfasst werden, stellt dies eine besondere Herausforderung bei der Archivierung von Websites dar, weil sichergestellt sein muss, dass sie mit abgespeichert werden. Idealerweise werden den Codierern die abgespeicherten Seiten mit allen multimedialen Elementen in der exakt gleichen Form wie in der Onlineversion zugänglich gemacht. Schließlich können komplexe Analysen von Websites, die Bilder und Videos enthalten, bisher nur mittels manueller Codierung adäquat durchgeführt werden. Mit Hilfe komplexer automatisierter Verfahren können Bild- und Toninformationen zwar in Texte umgewandelt werden (für einen Überblick vgl. Eble & Kirch, 2013). Der multimediale Charakter von Online-Inhalten wird hierdurch allerdings nicht abgebildet.

Nonlinearität und Hypertextualität: Texte lassen sich durch Hyperlinks mit anderen Texten, Bildern, Videos oder interaktiven Elementen verknüpfen. Hierdurch entsteht ein Informationsgeflecht, bei dem direkt auf die Quellen von Informationen verwiesen werden kann. Die Hyperlinkstruktur verursacht bei der Inhaltsanalyse von Websites vor allem zwei Probleme: das Speichern der Hyperlinkstruktur und die Archivierung der Websites. Beim ersten Problem handelt es sich vor allem um ein Kapazitätsproblem. Prinzipiell ist es technisch möglich, ausgehend von einer Website bis zu einer festgelegten Linktiefe alle Websites abzuspeichern, auf die verwiesen wird. Bei einer großen Linktiefe ist dies allerdings sehr zeitaufwändig und benötigt eine große Speicherkapazität, weil sich durch die Vernetzung die Zahl der zu speichernden Websites mit jeder Stufe der Linktiefe vervielfacht. Das zweite Problem besteht – ähnlich wie bei der Multimedialität – darin, dass die Hypertextstruktur bei der Archivierung der Websites erhalten bleiben muss.

Reaktivität und Personalisierung: Online-Inhalte können von Anbieterseite mit Hilfe von speziellen Algorithmen auf individuelle Nutzer zugeschnitten werden. Beispielsweise können Online-Medien per Algorithmen für einzelne Nutzer individuelle Startseiten generieren, auf denen Beiträge umso besser platziert sind, je eher sie dem vergangenen Leseverhalten der Nutzer entsprechen. Die Nutzer geraten im Extremfall in eine *Filter Bubble,* die ihnen nur Inhalte vorschlägt und präsentiert, die ihren Präferenzen entsprechen (Pariser, 2011). Die Individualisierung des Angebots ist ein erhebliches Problem für Online-Inhaltsanalysen, weil unterschiedlichen Codierern dabei möglicherweise unterschiedliche Inhalte angezeigt werden. Das Problem potenziert sich bei Inhaltsanalysen, in denen aus forschungsökonomischen Gründen nur die bestplatzierten Beiträge untersucht werden, weil es nicht mehr möglich ist, die bestplatzierten Beiträge rezipientenunabhängig zu bestimmen.

Verfügbarkeit von Meta-Daten: Mittlerweile stellen die meisten Websites auch so genannte Meta-Daten zur Verfügung. Diese Daten lassen Rückschlüsse auf die

Interaktion mit und zwischen Rezipienten zu. So verfügen viele Internetseiten heute über Kommentarspalten die häufig sogar für einzelne Beiträge nutzbar sind. Auch die Möglichkeit, Beiträge auf Facebook, Twitter oder GooglePlus zu empfehlen, ist mittlerweile weit verbreitet. Websites, die diese Daten zur Verfügung stellen, veröffentlichen dazu meist auch die Anzahl der Kommentare, Likes und Empfehlungen. Sofern es gelingt, diese Daten abzuspeichern oder automatisiert zu analysieren, eröffnen sich der Forschung vielfältige Analysemöglichkeiten, z. B. indem die Informationen mit Inhaltsanalysedaten verknüpft werden, um zu ermitteln, welche Beitragsmerkmale (z. B. Themen oder Bewertungstendenzen) zur Verbreitung eines Beitrags durch die Rezipienten führen.

3. Traditionelle Verfahren zur Speicherung von Websites

Grundvoraussetzung dafür, dass die oben diskutierten Probleme überhaupt gelöst werden können, ist im Regelfall eine Abspeicherung der Websites. Zur Speicherung und Archivierung von Websites wird in den meisten Studien bislang auf eines der folgenden Verfahren zurückgegriffen (Karlsson & Strömbäck, 2010): das Anfertigen von Screenshots, das Speichern als PDF-Datei, die Benutzung von Downloadprogrammen (sog. Crawlern, Offline-Browsern oder Webspidern) und der Zugriff über RSS-Feeds. Wir wollen die Vor- und Nachteile dieser Verfahren im Folgenden kurz diskutieren.

Screenshots und PDF-Dateien: Das Anfertigen von Screenshots ist ein vergleichsweise einfaches Verfahren. Die zu untersuchenden Websites werden hierbei manuell aufgerufen und als Bild-Dateien, z. B. im JPG-Format gespeichert. Dies kann entweder im Vorfeld der Codierung oder im gleichen Arbeitsschritt mit der Codierung geschehen. Das Verfahren ist technisch nicht besonders aufwändig, hat aber auch mehrere Nachteile: Zum einen ist der Zeitaufwand zur manuellen Abspeicherung jeder einzelnen Website enorm. Zum anderen können Screenshots weder multimediale Elemente noch Hyperlinks adäquat abbilden. Hyperlinks lassen sich allenfalls erahnen, z. B. wenn sie als unterstrichene oder farblich hervorgehobene Textstellen erkennbar werden. Welche Inhalte sich hinter dem Link verbergen, kann nicht ermittelt werden, weil der Link im Screenshot nicht angeklickt werden kann. Auch multimediale Inhalte werden nicht angezeigt, da JPG-Dateien statisch sind. Auch die Personalisierung des Webangebots wird beim Anlegen von Screenshots nicht umgangen. All dies gilt in ähnlicher Form auch für die Speicherung von Websites als PDF-Dateien. Allerdings können in neueren PDF-Versionen Hyperlinks angezeigt werden. Fährt man mit der Maus über einen Link, bekommt man die URL angezeigt, auf die dieser Link verweist (Mouse-Over-Effekt). Auf diese Art und Weise ist es in der Regel zumindest möglich zu erfassen, auf welche Seite ein Beitrag verlinkt, auch wenn die Seite nicht direkt angesehen werden kann.

Download-Programme: So genannte Webcrawler, Offline-Browser oder Webspider rufen Websites automatisch auf und speichern sie in verschiedenen Formaten ab. Zwei kostenlos nutzbare Programme, die in der Kommunikationswissenschaft häufiger eingesetzt werden, sind HTTrack und Wget. HTTrack ermöglicht es dem Nutzer, die URL einer Website einzugeben und eine Linktiefe zu definieren. Der Crawler speichert dann automatisch alle öffentlich zugänglichen

Bereiche der gewünschten Website sowie die Beiträge der Linkziele bis zur eingestellten Linktiefe. Die Websites lassen sich offline mit identischem Layout, Inhalt und gleicher Funktionsweise hinsichtlich der Hypertextualität öffnen. Videos und Tondokumente müssen allerdings manuell gespeichert werden. Ähnliches gilt auch für Wget. Einige professionelle, kostenpflichtige Programme wie der Offline Explorer oder Teleport bieten auch eine Archivierung multimedialer Inhalte an (Rüdiger & Welker, 2010). Hinsichtlich der Personalisierung der Seiten ist bei Webcrawlern nicht auszuschließen, dass Websites in einer personalisierten Form gespeichert werden. Für den Anbieter der Website sind IP-Adresse und z. B. das Betriebssystem des speichernden Rechners einsehbar. Bei HTTrack lassen sich einige der Informationen, die an den Betreiber der Website gesendet werden, zwar einstellen und manipulieren, dennoch kann nicht sicher davon ausgegangen werden, dass die Inhalt personenunabhängig gespeichert werden.

Speichern von RSS-Feed-Meldungen: Das einzige Verfahren, welches das Problem der Reaktivität und Personalisierung der Websites umgeht, ist das Speichern von RSS-Feed-Meldungen. Viele Betreiber von Websites stellen RSS-Feeds zur Verfügung, die meist automatisiert vom Content-Management-System angelegt werden. Ihre eigentliche Funktion besteht darin, die Nutzer über Änderungen auf den Websites zu informieren. Dies kann durch Kurznachrichten geschehen, aber auch dadurch, dass die gesamte neue Website per RSS-Feed zur Verfügung gestellt wird. Die Beiträge erscheinen in RSS-Feeds in umgekehrt chronologischer Reihenfolge – der neueste Beitrag also zuerst. RSS-Feeds abstrahieren folglich von der (individuellen) Ansicht der Website. Da für alle neuen Beiträge, die auf einer Website verfügbar sind, RSS-Feeds angelegt werden, ist die Speicherung der RSS-Feeds unabhängig davon, ob und mit welcher Platzierung die Beiträge einzelnen Nutzern angezeigt werden. Das löst in Wirkungsanalysen allerdings nicht das Problem, dass individuelle Nutzer unterschiedliche Beiträge angezeigt bekommen und nutzen. Dieses Problem besteht aber auch z. B. bei der Inhaltsanalyse von Tageszeitungen, bei der auch alle Beiträge codiert werden, obwohl offensichtlich ist, dass die Rezipienten nur einen (individuell verschiedenen) Teil davon lesen. Die RSS-Meldungen beinhalten zudem meist nur den zentralen Textartikel und erscheinen nicht im eigentlichen Format der Website. Wie stark beim Speichern dieser Meldungen Hypertextualität und Multimedialität erhalten bleiben, hängt vom Anbieter der Website ab. Immer enthalten ist zumindest ein Link, der zur Websitesversion des Beitrags führt. Multimediale Elemente sind dagegen in der Regel nicht eingebunden.

4. Die Analyse von Online-Angeboten mit Hilfe automatisierter Verfahren

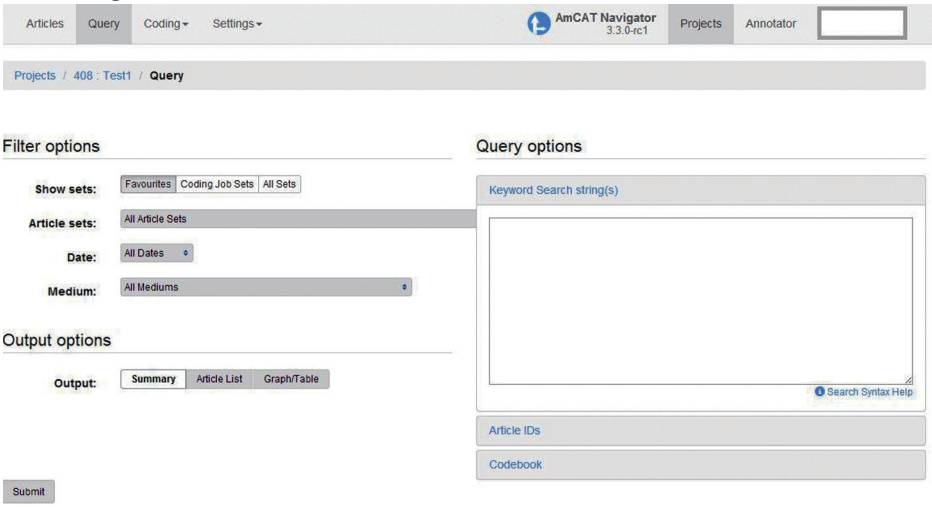
Vergleicht man die Vor- und Nachteile der verschiedenen traditionellen Verfahren zur Abspeicherung von Websites miteinander, wird deutlich, dass offensichtlich keines davon allen Anforderungen gerecht wird. Das Problem der Personalisierung der Seiten wird nur durch die Speicherung von RSS-Feeds gelöst, die sich formal zugleich aber so weit von der ursprünglichen Online-Version des Beitrags entfernen, dass z. B. multimediale Elemente nicht mit erfasst werden können. Wir wollen deshalb im Folgenden fünf vergleichsweise neue datenbankgestützte und

online mehr oder weniger frei verfügbare Tools vergleichen, die eine automatisierte Speicherung von Online-Inhalten ermöglichen und die Codierorganisation erleichtern: das Amsterdam Content Analysis Toolkit (AmCAT), den News Classifier zur Speicherung und Analyse von Textbeiträgen, die Codieroberfläche ANGRIST/IN-TOUCH, den Facepager zur Analyse von Social Media-Daten und ARTICLe, ein neues Tool zur Speicherung von Texten und Multimediadateien. Obwohl einige der Verfahren zugleich auch automatisierte Codierungen durchführen, steht diese Funktion hier nicht im Vordergrund. Wir werden dabei zunächst die Anwendungsmöglichkeiten und anschließend die jeweiligen Vor- und Nachteile der einzelnen Verfahren diskutieren. Dabei beschränken wir uns jeweils auf einen groben Überblick und verweisen für Details auf die ausführlichen Dokumentationen zu den Verfahren, die in der Regel online verfügbar sind.

AmCAT: Ein Tool, das die Organisation und den Codiervorgang von Online-Inhalten verbindet, ist das Amsterdam Content Analysis Toolkit (AmCAT)[2] (van Atteveldt, 2008). AmCAT ermöglicht es, eine große Anzahl von Beiträgen in einer SQL-Datenbank darzustellen und sie anschließend mithilfe verschiedener Programme automatisiert oder manuell zu codieren. Im ersten Arbeitsschritt werden hierzu Beiträge in den AmCAT Navigator geladen (Abb. 1). Dort können verschiedene Dateiformate abgelegt werden, die in einer Datenbank eingetragen werden. In dieser Datenbank werden automatisiert Informationen aus den Beiträgen, wie deren Quelle oder das Veröffentlichungsdatum, verzeichnet. AmCAT ist so konzipiert, dass die Archivierung der Beiträge manuell oder durch den Einsatz eines zusätzlichen Tools durchgeführt wird. Da AmCAT gezielt für semantische und netzwerkbasierte Textanalysen entwickelt wurde, liegt der Schwerpunkt auf der Speicherung und Organisation von Beiträgen in Textformaten wie XML, RTF oder CSV. Durch den Einsatz von AmCAT alleine, kann deshalb nicht gewährleistet werden, dass die Hypertextualität und Multimedialität von Online-Inhalten bei Inhaltsanalysen abgebildet werden. Es hängt vielmehr vom vorgeschalteten Speichervorgang ab, ob ersichtlich bleibt, wie viele Bilder, Videos oder Hyperlinks in den zu untersuchenden Beiträgen eingebunden sind. Die Online-Inhalte werden für die Verarbeitung in AmCAT mit einem zusätzlichen Tool gespeichert und dann in einem weiteren Arbeitsschritt manuell in den Navigator geladen. Die gespeicherten Daten können dabei komplett z. B. als ZIP-Archiv hochgeladen werden. Auch die Umgehung der Personalisierung von Websites hängt vom zuvor gewählten Archivierungsverfahren ab. AmCAT kann Algorithmen auf Websites nicht per se umgehen. Werden aber im Speicherverfahren z. B. die RSS-Feeds genutzt, um von dort Texte in die Datenbank zu laden, können Beiträge unabhängig vom Nutzer analysiert werden (van Atteveldt, 2008, S. 182).

2 Eine Dokumentation des Tools ist unter https://github.com/amcat/amcat abrufbar. Darüber hinaus kann man sich unter http://amcat.vu.nl/ als Benutzer registrieren und Projekte anlegen, ohne einen eigenen Server zu betreiben.

Abbildung 1: Online-Benutzeroberfläche von AmCAT



Quelle: http://amcat.vu.nl/ (Zugriff am 24.03.2014)

Der zweite Schritt im Arbeitsprozess mit AmCAT ist die Codierung der gespeicherten Beiträge. Die Stärke von AmCAT liegt in den Möglichkeiten der automatisierten Textanalyse. Durch die Verknüpfung mit Tools z. B. zum Natural Language Processing (NLP) oder zum Part-of-Speech Tagging (POS), werden vielfältige Analysen direkt durch den Computer durchgeführt. Gleichzeitig ermöglicht es AmCAT, durch menschliche Codierungen mittels einer Eingabemaske, die Validität der Codierungen zu prüfen oder Trainingsdaten für maschinelles Lernen durch die manuellen Codierungen zu generieren. Für den Codierprozess wurde das Tool iNet in AmCAT integriert. Es dient als Benutzeroberfläche, von der aus verschieden Analysen, z. B. Codierungen auf Beitragsebene und Codierungen auf Aussagenebene, durchgeführt werden. iNet ermöglicht die umfassende Organisation des kompletten Codiervorgangs, von der Zuteilung der Beiträge, über die Codierung bis hin zur Datengenerierung und -übertragung in ein Statistikprogramm (van Atteveldt, 2008, S. 185).

Eine Analyse der Meta-Daten von sozialen Netzwerkseiten, wie Likes und Shares eines Beitrages auf Facebook ist mit AmCAT nur begrenzt möglich und hängt von der Art der Speicherung der Beiträge ab. Wird ein Speicherverfahren eingesetzt, mit dem diese Meta-Daten in Textform vorliegen, so können sie auch mit den in AmCAT integrierten Tools analysiert werden. Insgesamt bietet AmCAT den Vorteil, dass eine Vielzahl von Arbeitsprozessen bei der Inhaltsanalyse digitaler Texte vereinfacht und miteinander verbunden werden. Das Tool verbindet somit die Vorteile automatisierter und manueller Textanalysen und ermöglicht einen kontinuierlichen Validitätstest der automatisierten Codierung. Darüber hinaus ermöglicht es, Trainingsdaten für das maschinelle Lernen zu generieren. Dem steht allerdings die hohe Komplexität des Tools gegenüber. Durch die vielfältige Kombinierbarkeit der einzelnen Elemente sind sehr weitreichende Kenntnisse von Programmiersprachen und der Logik automatisierter Textanalyseverfahren nötig.

Darüber hinaus ergeben sich durch die Konzipierung von AmCAT für automatisierte Analysen neben den großen Vorteilen des Tools auch Nachteile: Online-Inhalte werden in reiner Textform in die Datenbank integriert. Eine ganzheitliche Darstellung z. B. als HTML-Datei ist nicht möglich. Hypertextualität, Interaktivität und Multimedialität können folglich nicht erfasst werden, da Elemente wie Hyperlinks, Kommentarfelder, Bilder oder Videos nicht in der Datenbank abgelegt werden. Diese Nachteile können aber prinzipiell umgangen werden, wenn man AmCAT mit einem Tool kombiniert, das Websites vollständig abspeichert und gleichzeitig die Texte extrahiert.

NewsClassifier: Auch das Tool NewsClassifier[3] (Scharkow, 2012) ist dafür entwickelt worden, den gesamten Inhaltsanalyseprozess von der Archivierung bis hin zur Codierung automatisiert durchzuführen.

> „NewsClassifier ist als integriertes Framework konzipiert, das von der automatischen Datenerhebung und -bereinigung über die Stichprobenziehung, die Organisation der Feldarbeit und die Durchführung von Reliabilitätstests bis hin zur eigentlichen manuellen und/oder automatischen Codierung reicht." (Scharkow, 2012, S. 250).

Durch die automatisierte Speicherung von Online-Inhalten ermöglicht das Tool im ersten Arbeitsschritt die kontinuierliche Erhebung von journalistischen Online-Angeboten. Die zeitaufwendige Recherche und Speicherung der Untersuchungseinheiten wird hierbei erheblich reduziert, weil nur eine einmalige Eingabe der zu untersuchenden Websites nötig ist. Die Speicherung der Websites erfolgt über den Zugriff auf ihre RSS-Feeds. Dies bietet den Vorteil, dass Algorithmen umgangen werden, die eine personalisierte Darstellung der Websites bedingen. Stattdessen werden alle Beiträge gespeichert, die innerhalb eines bestimmten Zeitraums veröffentlicht wurden. Beim Speichern der Websites wird nach einem vierstufigen Verfahren vorgegangen: Als erstes werden die URL-Adressen der RSS-Feeds der zu untersuchenden Websites eingetragen. Im zweiten Schritt wird geprüft, ob bereits in den RSS-Feeds Volltexte vorliegen. Ist dies der Fall, werden diese in eine relationale Datenbank eingetragen. Sind keine Volltexte vorhanden, wird im dritten Schritt geprüft, ob die HTML-Seiten des jeweiligen Beitrags vorhanden sind. Sind diese erhältlich, werden sie in die relationale Datenbank importiert und bereinigt, d.h. der reine inhaltliche Text der zu untersuchenden Website wird in der Datenbank gespeichert. Sind die HTML-Seiten nicht verfügbar, wird geprüft, ob Druckversionen der Beitragsseiten von den Anbietern der Websites zur Verfügung gestellt werden. Ist dies der Fall, werden diese in Textform in der Datenbank gespeichert (Scharkow, 2012, S. 260). Der NewsClassifier zielt folglich vor allem auf die automatisierte Textanalyse ab. Das redaktionelle Umfeld der Beiträge wird dagegen weitgehend ausgeklammert.

Neben der Speicherung ermöglicht der NewsClassifier die Stichprobenziehung und die Zuteilung der Beiträge zu den Codierern. Auch die Codierung selbst wird durch die Codierer mittels einer Eingabemaske direkt am Rechner vorgenommen. Hierfür kann im Programm ein Codebuch angelegt werden. Für jede Kategorie

3 Das Tool ist unter https://github.com/mscharkow/newsclassifier dokumentiert und kann mit Programmierkenntnissen eingesetzt und weiterentwickelt werden.

kann dabei festgelegt werden, auf welchen Bereich des zu analysierenden Textes sie sich bezieht, z. B. auf die Überschrift oder den Fließtext. Alle Kategorien lassen sich neben der manuellen Codierung auch durch den Einsatz automatisierter Verfahren erheben. Die manuellen Codierungen liefern hierbei die Trainingsdaten für die automatische Klassifikation. Je mehr korrekte manuelle Codierungen für eine Kategorie vorliegen, desto zuverlässiger gelingt die automatisierte Codierung. Allerdings beziehen sich die Codierungen immer auf den Beitrag als Ganzes. Eine Codierung einzelner Aussagen oder Abschnitte ist bisher nicht möglich (Scharkow, 2012, S. 277). Schließlich ermöglicht es der NewsClassifier, Reliabilitätstests durchzuführen und die erhobenen Daten unkompliziert in ein Statistikprogramm zu übertragen. Der gesamte Arbeitsablauf einer Inhaltsanalyse von Online-Inhalten lässt sich also prinzipiell mit dem Einsatz dieses Tools organisieren (Scharkow, 2012, S. 268). Der größte Nachteil des Tools besteht darin, dass es weitgehend auf die Analyse von Textartikeln beschränkt bleibt. Zwar ist die Archivierung von HTML-Dateien möglich und auch Video- und Audio-Dateien können prinzipiell analysiert werden. Allerdings ist die Speicherung multimedialer Elemente und vor allem die Verknüpfung dieser Elemente mit den Beiträgen, in die sie eingebunden sind, nicht möglich. Hyperlinks und interaktive Elemente wie z. B. Nutzerkommentare müssen zudem durch menschliche Codierer analysiert werden.

ANGRIST/IN-TOUCH: Einen etwas anderen Schwerpunkt legen die Tools ANGRIST (Adjustable Non-commercial Gadget for Relational data Input in Sequential Tasks) und IN-TOUCH[4] (Integrated Technique for Observing and Understanding Coder Habits) (Wettstein, 2012; Wettstein, Reichel, Kühne, & Wirth, 2012). Die beiden Tools setzen erst nach der Speicherung von Online-Inhalten an und ermöglichen eine computergestützte manuelle und halbautomatisierte Codierung, bei der die Codierer direkt am Rechner arbeiten. Hierdurch sind sie von vorneherein nicht dafür konzipiert, die Quantität und die Dynamik von Online-Inhalten zu bewältigen, sondern setzen dezidiert bei der Digitalisierung und Maschinenlesbarkeit an. ANGRIST stellt dabei ein Skript in der Programmiersprache Python dar, das die schrittweise Abfrage einzelner Codebuch-Kategorien ermöglicht. Hierzu werden im ersten Arbeitsschritt das Codebuch und die zu codierenden Artikel für den Zugriff durch das ANGRIST-Script formatiert. Die Beiträge müssen hierzu in Unicode- oder ASCII-Formatierung vorliegen. Das bedeutet, innerhalb des Programms lassen sich Buchstaben, Zahlen und eine begrenzte Anzahl von Sonderzeichen darstellen. Multimediale, hyptertextuelle und interaktive Inhalte werden nicht dargestellt. Anhand des hinterlegten Codebuches werden die Codierer mit ANGRIST von Kategorie zu Kategorie geführt. Die Codierer arbeiten dabei direkt mit einer Eingabemaske. Dies bietet den Vorteil, dass die Codierer sich keine Zahlencodes merken müssen, sondern wie bei einer Befragung ausformulierte Ausprägungen präsentiert bekommen, die sie durch verschiedene Auswahlwerkzeuge wie Dropdown-Menüs oder Checkboxen vergeben können. Die Codierungen liegen dann bereits in digitaler Form vor und können leicht in einem Statistikprogramm weiterverarbeitet werden. Insgesamt erleichtert das Tool die menschliche Codierung von Online-Inhalten

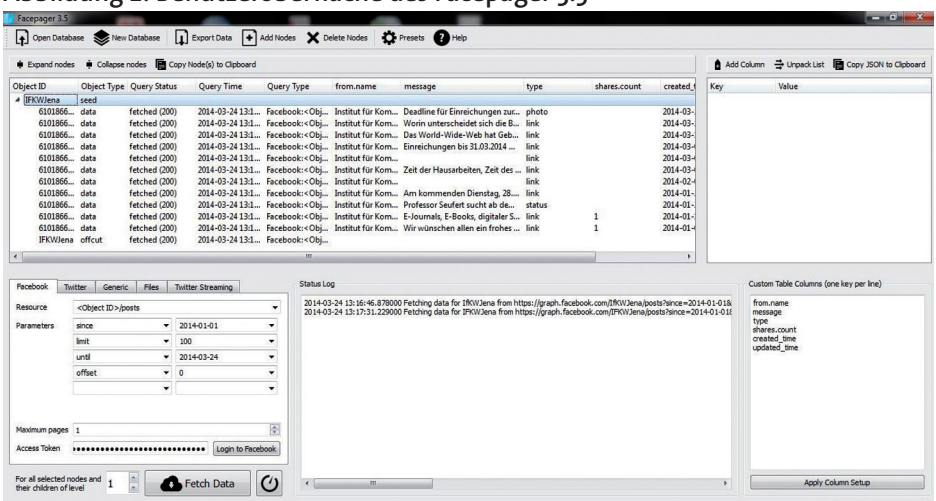
4 Eine Dokumentation zur selbstständigen Implementierung des Tools ANGRIST ist verfügbar unter: http://www.ipmz.uzh.ch/Abteilungen/Medienpsychologie/Recource/Angrist.html.

erheblich. Aufgrund seiner Konzipierung bildet es die Hypertextualität und Multimedialität von Websites allerdings nicht ab. Da es sich bei ANGRIST nicht um ein Tool für die Archivierung handelt, kann mit ihm auch der Personalisierung von Online-Angeboten nicht begegnet werden. Die Erhebung von Meta-Daten ist nur durch menschliche Codierer möglich.

Das Tool IN-TOUCH lässt sich als Ergänzung zu ANGRIST einsetzen und dient der Überwachung und Auswertung des Codierverhaltens von menschlichen Codierern. Mit dem Tool soll vor allem der Codierprozess kontrollierbar gemacht werden. Neben dem Verhalten der Codierer wie z. B. der Codierergeschwindigkeit und der Tageszeit der Codierung lassen sich auch Reliabilitätstests durchführen. Für die computergestützte Inhaltsanalyse durch menschliche Codierer bietet das Tool enorme Vorteile, da sehr arbeitsintensive Schritte wie Reliabilitätstest, Codierauswahl und -schulung deutlich vereinfacht werden können. Die beiden Tools sind insgesamt für die computerunterstützte manuelle Inhaltsanalyse zu empfehlen, ermöglichen allerdings keine Speicherung von multimedialen, hypertextuellen und interaktiven Online-Inhalten. Sie entwickeln ihre Stärke folglich vor allem in der Kombination mit anderen Tools zur automatisierten Speicherung von Online-Inhalten.

Facepager: Ein Tool zur automatisierten Speicherung von Social-Media-Angeboten ist der von Keyling und Jünger (2013) entwickelte Facepager[5] (Abb. 2). Der Facepager setzt an den jeweiligen Programmierschnittstellen (engl. API, Application Programming Interface) von Facebook und Twitter an und lässt sich prinzipiell auf Online-Angebote erweitern, die mit dem Java Script basierten Datenformat JSON arbeiten. Mit dem Tool lassen sich große Mengen von Meta-Daten aber auch formale und inhaltliche Informationen von Facebook-Seiten und Twitter-Kanälen automatisiert speichern.

Abbildung 2: Benutzeroberfläche des Facepager 3.5



Quelle: https://github.com/strohne/Facepager (Zugriff am 24.03.2014)

5 Der Facepager kann zur freien Verwendung und Weiterentwicklung unter https://github.com/strohne/Facepager heruntergeladen werden.

Der Arbeitsablauf beginnt mit dem manuellen Hinzufügen der zu speichernden Facebookseiten, Twitterkanäle oder sonstigen Quellen mit JSON-basierter Schnittstelle. Anschließend wird ausgewählt, welche Informationen von den betroffenen Seiten erfasst werden sollen. Die Möglichkeiten reichen hier von der Erfassung der bloßen Anzahl von „Fans" und „Followern" bis zum automatisierten Herunterladen von Statusmeldungen und Kommentaren. In der neuesten Fassung ermöglicht das Tool auch das automatisierte Speichern von auf Facebook oder Twitter geteilten Dateien wie z. B. Bilddateien. Die Möglichkeiten zur Speicherung sind beim Facepager davon abhängig, welche Informationen von den Seitenbetreibern also z. B. von Facebook oder Twitter zur Verfügung gestellt werden. Prinzipiell lassen sich hier aber durch eigene Scripts vielfältige Informationen auch von anderen Plattformen extrahieren. Sind die gewünschten Optionen gewählt, beginnt der Speichervorgang. Um nicht-öffentliche Bereiche von sozialen Netzwerkseiten zu analysieren benötigt man einen eigenen Account im betreffenden Netzwerk mit Benutzernamen und Passwort. Hiermit loggt man sich über den Facepager ein und beginnt die automatisierte Erhebung der gewünschten Informationen. Diese werden im nächsten Schritt zunächst innerhalb des Programms angezeigt, lassen sich aber problemlos für die Weiterverarbeitung in Excel oder ein Statistikprogramm übertragen. Während multimediale Dateien wie Bilder in ihrer ursprünglichen Form auf dem eingesetzten Rechner abgelegt werden – z. B. im Bildformat JPG – werden die textlichen und numerischen Inhalte wie Fan-Zahlen, Follower-Zahlen oder Statusmeldungen von den jeweiligen Facebookseiten bzw. Twitter-Kanälen extrahiert und liegen in Textform vor. Die Inhalte der Social-Media-Seiten werden auf diese Weise in maschinenlesbarem Format abgelegt. Dies stellt einen erheblichen Vorteil für die automatisierte Textanalysen dar. Sie kann mit einem zusätzlichen Tool – z. B. ANGRIST, AmCAT oder NewsClassifier – durchgeführt werden. Gleichzeitig erschwert das Vorliegen der Informationen in bloßer Textform aber manuelle Codierungen. Denn alle gespeicherten Informationen werden den Codierern in Tabellenform und nicht in ihrem ursprünglichen Layout angezeigt. Das bedeutet, dass zwar im Text ersichtlich ist, wie viele und welche Hyperlinks in den Beiträgen vorhanden sind sowie ob und wie viele multimediale Elemente enthalten sind, die entsprechenden multimedialen Elemente werden aber losgelöst von ihrem redaktionellen Kontext abgespeichert. Darüber hinaus ist der Facepager bisher nicht für die Speicherung herkömmlicher Websites einsetzbar, sondern dezidiert für Social-Media-Seiten konzipiert. Hinsichtlich der Personalisierung von Online-Inhalten bestehen die Vorteile des Facepagers in dessen Zugriff auf die Informationen über die jeweilige Programmierschnittstelle des Anbieters. Es ist davon auszugehen, dass die hier verfügbaren Informationen unabhängig vom individuellen Nutzungsverhalten des jeweiligen Codierers sind. Insgesamt lässt sich folgern, dass der Facepager für die Speicherung von Social-Media-Seiten bestens geeignet ist und die Voraussetzungen für automatisierte Inhaltsanalysen schafft, bislang aber nicht für die Codierung journalistischer Online-Angebote konzipiert ist.

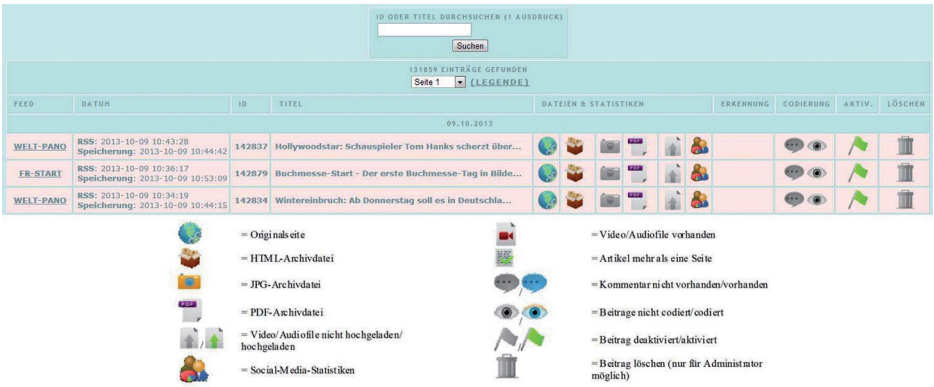
ARTICLe: Das Tool ARTICLe[6] (Automatic Rss-crawling Tool for Internet-based Content anaLysis) legt den Schwerpunkt auf die automatisierte Speicherung von Websites sowie die Aufbereitung und Organisation von Beiträgen für Codierungen. Es wurde an der Universität Jena von Thomas Holbach programmiert und im Rahmen des DFG-Projekts „Digitale Wissensklüfte" von Christoph Uschkrat und Jörg Haßler weiterentwickelt. Im Unterschied zu allen anderen Programmen speichert es die Websites vollständig automatisiert inklusive aller multimedialen und hypertextuellen Elemente sowie Meta-Daten zur Verbreitung von Beiträgen in sozialen Netzwerken. Zudem dient die Datenbank als Codierplattform, die verschiedene Interaktionsmöglichkeiten bietet. Ausgangspunkt des Zugriffs auf die zu untersuchenden Websites sind deren RSS-Feeds.

Nach dem manuellen Eintragen der RSS-Feeds aller zu untersuchenden Websites, besteht der erste automatisierte Arbeitsschritt in der *Speicherung* aller Artikel. Hierbei werden alle Artikel über ihre Verlinkung innerhalb der RSS-Feeds gespeichert. Bei ARTICLe liegt der Fokus nicht auf der Extrahierung von Rohtexten. Ziel ist es vielmehr, die Websites für die Codierer so darzustellen, wie sie auch tatsächlich online abrufbar sind bzw. waren. Daher werden Screenshots in den Formaten HTML, PDF und JPG gespeichert und in einer relationalen Datenbank angelegt. Die Speicherung kann in beliebigen Zeitabständen, z. B. alle zwei Stunden, durchgeführt werden.

Ebenfalls automatisiert verläuft die Speicherung in Artikeln enthaltener Videos und Tondokumente sowie aller Seiten von Artikeln, die länger als eine Seite sind. Die Erkennung von multimedialen Elementen und Folgeseiten basiert auf einem zuvor manuell angelegten Schlagwortkatalog. Kommt eines der Schlagworte in einem Artikel vor, weist die Datenbank auf das Vorhandensein von Videos, Tondokumenten oder Artikeln mit mehr als einer Seite hin. Hierzu muss zunächst recherchiert werden, welche Dateiformate in den Quellcodes der zu untersuchenden Websites verwendet werden. Bindet eine Website z. B. regelmäßig Youtube-Videos ein, kann es hinreichend sein, dass die Datenbank das Wort „youtube" innerhalb der Quellcodes sucht. Für Artikel mit mehr als einer Seite kommen etwa Wörter wie „Seite 2" oder „nächste Seite" in Frage. Anhand eines so genannten regulären Ausdrucks (engl. regular expression, RegExp) werden die Videos und Folgeseiten extrahiert und mittels eines PHP-Scripts heruntergeladen. Alle gespeicherten Artikel werden gemeinsam mit allen enthaltenen multimedialen Elementen in umgekehrt chronologischer Reihenfolge in einer Datenbanktabelle abgelegt (Abb. 3). Diese Datenbankansicht dient zugleich als Oberfläche für die Codierung der Beiträge (s.u.). Zusätzlich speichert die Datenbank Screenshots von der Startseite der eingetragenen Website, von der die RSS-Feeds stammen. Hierdurch werden beispielsweise Analysen zur Dynamik von Startseiten ermöglicht, die dann allerdings unabhängig von den RSS-Feeds und folglich nicht entpersonalisiert sind.

6 Die Benutzeroberfläche von ARTICLe ist unter https://141.35.119.155/feeds/view/ abrufbar. Die Veröffentlichung des Codes auf github.com ist derzeit in Vorbereitung.

Abbildung 3: Datenbankansicht von ARTICLe



Quelle: https://141.35.119.155/feeds/view/

Im zweiten Arbeitsschritt erhalten die Codierer über die Internetadresse der Datenbank oder intern im Netz der Universität Zugang zu den archivierten Artikeln auf einem passwortgeschützten Server. Die Datenbank basiert auf MySQL und ist für den Zugriff der Codierer im Format PHP als Website aufbereitet. In diesen Formaten werden Einbindung und Platzierung von dynamischen multimedialen Elementen, wie Videos und Tondokumenten, zwar angezeigt, sie lassen sich aber nicht abspielen. Deshalb werden die getrennt gespeicherten multimedialen Elemente innerhalb der Datenbank zusätzlich noch einmal in der gleichen Zeile angezeigt wie der dazugehörige Artikel. Nach der Codierung der Textartikel folgt die Codierung von eventuell vorhandenen Videos und Tondokumenten. Ist die manuelle Codierung abgeschlossen, markiert der Codierer den Artikel als codiert. Auf diese Weise wird der Fortschritt der Codierarbeiten in Echtzeit ablesbar. Treten Probleme oder Fragen seitens der Codierer auf, haben sie die Möglichkeit, einen Kommentar zu hinterlassen. Im dritten Schritt werden die *Meta-Daten* der Beiträge, z. B. Likes und Shares auf Facebook, automatisiert weiterverarbeitet. Sie können dann direkt in ein Statistik-Programm eingepflegt werden. Die Datenbank wertet Daten von Facebook, Twitter und GooglePlus aus. Diese werden für jeden Beitrag in Tabellenform dargestellt und sind in der jeweiligen Spalte des betreffenden Beitrags in der Datenbank abrufbar.

Insgesamt liegen die Vorteile von ARTICLe folglich in der umfassenden automatisierten Speicherung von Websites inklusive aller Hyperlinks, mulitmedialer Elemente und den Meta-Daten von sozialen Netzwerken. Damit bietet das Tool die Voraussetzungen für die umfassende manuelle Analyse von Online-Inhalten. Darüber hinaus ermöglicht es die Benutzeroberfläche, mit den Codierern zu kommunizieren und den Codierprozess zu überwachen. Allerdings ist die Datenbank bislang nicht für die automatisierte Codierung von Texten oder gar multimedialen Elementen konzipiert. Für die Integration automatisierter Analyseverfahren über die Analyse von Meta-Daten hinaus können aber zusätzlich Parser oder Tools zur semantischen oder netzwerkbasierten Textanalyse eingesetzt werden.

5. Zusammenfassung und Diskussion

Die zunehmende Bedeutung des Internets als politisches Kommunikationsmedium führt dazu, dass auch Inhaltsanalysen von Online-Medien an Bedeutung gewinnen. Allerdings unterscheiden sich Online-Medien in vielfacher Hinsicht von Offline-Medien: Vor allem die Digitalisierung und Maschinenlesbarkeit sowie die Möglichkeit, Meta-Daten zu erheben, stellen erhebliche Chancen der Online-Inhaltsanalyse dar. Dagegen stellen die Flüchtigkeit und Dynamik ihrer Inhalte, die Multimedialität, die Hypertextualität sowie die Personalisierung der Beiträge diejenigen, die Online-Inhaltsanalysen durchführen wollen, vor erhebliche Probleme. Den genannten Problemen lässt sich zunächst vor allem durch eine geeignete Speicherung der zu untersuchenden Websites begegnen. Die hierfür bislang gängigen Verfahren (Screenshots, Webcrawler, RSS-Feeds) weisen jedoch durchweg spezifische Nachteile auf. Deshalb wurden in den vergangenen Jahren einige Tools entwickelt, die mit unterschiedlichen Schwerpunkten die Speicherung, aber auch zahlreiche andere Arbeitsschritte von Online-Inhaltsanalysen automatisieren.

Da viele dieser Tools aber in erster Linie mit dem Ziel konzipiert sind, automatisierte Inhaltsanalysen von Texten zu ermöglichen, sind Analysen von komplexen Websites unter Berücksichtigung von Hyperlinks, interaktiver Elemente und multimedialer Inhalte gar nicht oder nur mit erheblichen Einschränkungen möglich. Andere Tools sind für die Speicherung von Websites optimiert und speichern diese mit Hyperlinks und multimedialen Elementen ab, können aber keine automatisierten Codierungen durchführen. Einen Überblick über die Funktionen der Tools gibt Tabelle 1.

Tabelle 1: Die Funktionen der Tools im Vergleich

	Am-CAT	News-Classifier	ANGRIST / IN-TOUCH	Face-pager	ARTICLe
Automatisierte Speicherung von Hyperlinks	–	X	–	X	X
Automatisierte Speicherung von multimedialen Elementen	–	–	–	(X)	X
Umgehung der Personalisierung	–	X	–	X	X
Funktion als Codier-Plattform/ Codierverwaltung	X	X	X	–	X
Download und Analyse von Meta-Daten von sozialen Netzwerkseiten	–	–	–	X	X
Automatisierte Codierung von Texten	X	X	–	–	–
Automatisierte Codierung von multimedialen Elementen	–	–	–	–	–

Insgesamt bietet sich für die Analyse von journalistischen Online-Inhalten deshalb meist eine Verknüpfung der verschiedenen Verfahren an. Je nach Fragestellung kann der Fokus dabei stärker auf dem Einsatz menschlicher Codierer oder automatisierter Codierungen liegen. So lassen sich etwa mit dem Facepager alle Sta-

tusmeldungen innerhalb eines gewählten Facebookprofils speichern. Diese werden von dort aus als CSV-Datei extrahiert und lassen sich in AmCAT hochladen. Dort können sie mit dem programmierten Codebuch analysiert werden. Der vermutlich sinnvollste Ansatz zur umfassenden Analyse von Online-Inhalten ist die Speicherung von RSS-Feeds, die von den Seitenanbietern zur Verfügung gestellt werden. Sie dienen dabei aber nur als Ausgangspunkt für die Speicherung der Beiträge, die ausgehend von den Links innerhalb der Feeds aufgerufen und gespeichert werden können. Werden nun automatisch Screenshots in Formaten wie HTML oder PDF angelegt, können Hypertextualität und Multimedialität von Online-Inhalten bei der Analyse mit berücksichtigt werden. Für die automatisierte Textanalyse werden in einem zusätzlichen Arbeitsschritt die Rohtexte extrahiert. Den menschlichen Codierern würden somit Dateien im Layout der tatsächlichen Websites für ihre Analysen präsentiert, während die reinen Texte automatisiert verarbeitet würden. Mit diesem Vorgehen können z. B. komplexe Kategorien manuell codiert werden, während Meta-Daten zum „Erfolg" des Beitrags beim Publikum automatisch erfasst und ausgewertet werden. Ein solches Vorgehen kann z. B. durch die Verbindung von ARTICLe mit dem NewsClassifier, AmCAT oder ANGRIST realisiert werden. Auf diese Weise lassen sich die Vorteile manueller und automatisierter Analysen von Online-Inhalten umfassend umsetzen.

Allerdings bleibt auch bei der Kombination verschiedener Tools zur Speicherung und Weiterverarbeitung von Online-Inhalten festzuhalten, dass derartige Verfahren eine regelmäßige manuelle Überprüfung erfordern. Unerwartete Veränderungen wie etwa die Veränderung der RSS-Feeds oder die Einführung so genannter Paywalls erschweren die Speicherung enorm. Ohne manuelle Eingriffe wie z. B. die Erstellung eines Accounts zum Abruf von Artikeln lassen sich Bezahlinhalte nicht automatisiert speichern. Diese Probleme treten allerdings auch bei Offline-Inhaltsanalysen auf, wenn z. B. eine Bibliothek einen Zeitungstitel nicht abonniert. Durch sorgfältige Vorbereitung, Planung und eine Abwägung von Kosten und Nutzen der Beschaffung der Titel sowie einer kritischen Beurteilung des Aufwandes und des Ertrages einer Analyse betreffender Titel und Beiträge, lassen sich diese Probleme minimieren. Andere online-spezifische Probleme, wie unerwartete Änderungen von Internetadressen, nicht abrufbare Seiten oder Darstellungsfehler bei Javascript- oder Flash-Inhalten, lassen sich häufig nur durch manuelle Überprüfungen des Speichervorgangs verhindern. Hier müssen hinsichtlich des Speicherverfahrens ggf. schrittweise Abstriche gemacht werden: Lässt sich eine Website nicht automatisiert speichern, so ist zunächst manuell nach den Ursachen zu suchen. Anschließend kann es genügen, die Inhalte alternativ mit einem anderen Tool, wie z. B. einem Webcrawler zu erfassen, ist dies nicht möglich bleibt häufig nur der Rückgriff auf das manuelle Anlegen von Screenshots, z. B. als PDF-Datei. Idealerweise werden diese manuell gespeicherten Beiträge den Codierern jedoch in der relationalen Datenbank auf die gleiche Weise wie die automatisch gespeicherten Artikel bereitgestellt. Das Auftreten solcher Detailprobleme ist durch eine ausführliche Sichtung des Materials vor dem Beginn der Speicherung zu minimieren. Ist man mit den Online-Angeboten, die gespeichert werden, vertraut, so werden häufig Muster erkennbar, an welchen Stellen Unregelmäßigkeiten auftreten können und wie diese zu bewältigen sind. So wird z. B. erkennbar, in

welchen Ressorts besonders häufig multimediale Elemente in den Programmiersprachen Flash oder Javascript eingesetzt werden.

Die sorgfältige Planung des Forschungsvorhabens sowie die sinnvolle Kombination der hier präsentierten Verfahren begegnen somit prinzipiell allen hier diskutierten technischen Herausforderungen, die bei der Analyse von journalistischen Online-Angeboten entstehen. Lediglich die automatisierte Codierung von multimedialen Elementen ist bislang nahezu unmöglich. Zwar existieren hier bereits automatisierte Tools zur Erfassung spezieller Bildmerkmale wie z. B. der Gestik und Mimik der dargestellten Personen (z. B. Cohn & Ekman, 2005) oder der allgemeinen Bildstruktur (z. B. Stommel & Müller, 2011). Eine detaillierte automatisierte Erfassung aller Bildinhalte scheint dagegen kaum realisierbar.

Literatur

Cohn, J. F., & Ekman, P. (2005). Measuring facial action. In J. A. Harrigan, R. Rosenthal, & K. S. Scherer (Hrsg.), *The new handbook of methods in nonverbal behavior research* (S. 9-64). Oxford: Oxford University Press.

de Kunder, M. (2013). *The size of the World Wide Web (The Internet)*. Verfügbar unter http://www.worldwidewebsize.com (Zugriff am 24.03.2014).

de Nooy, W., & Kleinnijenhuis, J. (2013). Polarization in the media during an election campaign: A dynamic network model predicting support and attack among political actors. *Political Communication*, *30*(1), 117–138. doi:10.1080/10584609.2012.737417

Eble, M., & Kirch, S. (2013). Wissenstransfer und Medienerschließung: Werkzeuge für die Integration von Multimedia-Inhalten in das Wissensmanagement. *Open Journal of Knowledge Management*, *7*(1), S. 42–46. Verfügbar unter http://www.community-of-knowledge.de/beitrag/wissenstransfer-und-medienerschliessung-werkzeuge-fuer-die-integration-von-multimedia-inhalten-in-d/ (Zugriff am 24.03.2014).

Hasebrink, U., & Schmidt, J.-H. (2013). Medienübergreifende Informationsrepertoires. *Media Perspektiven*, (1), 2–12.

Karlsson, M., & Strömbäck, J. (2010). Freezing the flow of online news: Exploring approaches to the study of the liquidity of online news. *Journalism Studies*, *11*(1), 2–19. doi:10.1080/14616700903119784

Keyling, T., & Jünger, J. (2013). Facepager (Version, f.e. 3.3). An application for generic data retrieval through APIs. Verfügbar unter: https://github.com/strohne/Facepager (Zugriff am 24.03.2014).

King, G., & Lowe, W. (2003). An automated information extraction tool for international conflict data with performance as good as human coders: A rare events evaluation design. *International Organization*, *57*(3). doi:10.1017/S0020818303573064

Lewis, S. C., Zamith, R., & Hermida, A. (2013). Content analysis in an era of big data: A hybrid approach to computational and manual methods. *Journal of Broadcasting & Electronic Media*, *57*(1), 34–52. doi:10.1080/08838151.2012.761702

Neuberger, C., Nuernbergk, C., & Rischke, M. (2009). Journalismus – neu vermessen: Die Grundgesamtheit journalistischer Internetangebote – Methode und Ergebnisse. In C. Neuberger, C. Nuernbergk, & M. Rischke (Hrsg.), *Journalismus im Internet. Profession, Partizipation, Technisierung* (S. 197–230). Wiesbaden: Verlag für Sozialwissenschaften / GWV Fachverlage.

Pariser, E. (2011). *The filter bubble: What the Internet is hiding from you*. New York: Penguin Press.

royal.pingdom.com. (2013). *Internet 2012 in numbers*. Verfügbar unter http://royal.pingdom.com/2013/01/16/internet-2012-in-numbers/ (Zugriff am 24.03.2014).

Rüdiger, K., & Welker, M. (2010). Redaktionsblogs deutscher Zeitungen. Über die Schwierigkeiten diese inhaltsanalytisch zu untersuchen – ein Werkstattbericht. In M. Welker & C. Wünsch (Hrsg.), *Neue Schriften zur Online-Forschung: Vol. 8. Die Online-Inhaltsanalyse. Forschungsobjekt Internet* (S. 448–468). Köln: Herbert von Halem.

Scharkow, M. (2012). *Automatische Inhaltsanalyse und maschinelles Lernen*. Berlin: epubli.

Sjøvaag, H., Moe, H., & Stavelin, E. (2012). Public service news on the web: A large-scale content analysis of the Norwegian Broadcasting Corporation's online news. *Journalism Studies*, *13*(1), 90–106. doi:10.1080/1461670X.2011.578940

Stommel, M., & Müller, J. (2011). Automatische, computerunterstützte Bilderkennung. In T. Petersen & C. Schwender (Hrsg.) *Die Entschlüsselung der Bilder. Methoden zur Erforschung visueller Kommunikation. Ein Handbuch* (S. 246-263). Köln: Herbert von Halem.

van Atteveldt, W. (2008). *Semantic network analysis: Techniques for extracting, representing and querying media content*. Charleston, SC: BookSurge.

van Eimeren, B., & Frees, B. (2013). Rasanter Anstieg des Internetkonsums – Onliner fast drei Stunden täglich im Netz: Ergebnisse der ARD/ZDF-Onlinestudie 2013. *Media Perspektiven*, (7-8), 358–372.

Welker, M., Wünsch, C., Böcking, S., Bock, A., Friedemann, A., Herbers, M., Isermann, H., Knieper, T., Meier, S., Pentzold, C., Schweitzer, E. J. (2010). Die Online-Inhaltsanalyse: Methodische Herausforderung, aber ohne Alternative. In M. Welker & C. Wünsch (Hrsg.), *Neue Schriften zur Online-Forschung: Vol. 8. Die Online-Inhaltsanalyse. Forschungsobjekt Internet* (S. 9–30). Köln: Herbert von Halem.

Wettstein, M. (2012). angrist.py. Dokumentation und Anleitung für die Programmierung des Codierer-Interface. Verfügbar unter: http://uzh.academia.edu/MartinWettstein/Papers/965234/angrist.py (Zugriff am 24.03.2014).

Wettstein, M., Reichel, K., Kühne, R., & Wirth, W. (2012). IN-TOUCH – ein neues Werkzeug zur Prüfung und Bewertung von Codiereraktivitäten bei der Inhaltsanalyse. Vortrag auf der 13. Jahrestagung der SGKM, Neuchâtel.

Extended Abstract

Advancement through technology? The analysis of journalistic online-content by using automated tools[1]

Jörg Haßler, Marcus Maurer & Thomas Holbach

1. Introduction

Without any doubt, the Internet is continually gaining in significance for political communication research. At present, about 75 percent of the German population state that they use the Internet at least occasionally (van Eimeren & Frees, 2013, p. 363). All traditional mass media operate websites that provide real-time information. For citizens, these journalistic websites are the most important information sources online (Hasebrink & Schmidt, 2013, p. 8).

The growing importance of online media also gives rise to consequences for content analyses of journalistic online media coverage. Because such analyses consider wide-reaching media that are representative for the entire media system or media that are meant to be the basis for effect analyses, journalistic online media have to be included in many content analyses nowadays. On the one hand, such analyses appear very promising because online media use standardized programming languages and codes are available in digitized form. On the other hand, the quantity, dynamics, multimediality, hypertextuality or the personalization set boundaries for storage and analyses of websites. This article discusses frequently used strategies to address those challenges and presents five recently developed tools for automated storage, organization or coding of online content.

2. Challenges of the Content Analysis of Websites

The internet constantly changes. This holds true both for the available websites in their entirety and for individual articles within web services. For practical reasons and for reasons of intersubjective comprehensibility, websites have to be stored before analysis. As websites are standardized, the dynamics can be addressed by automatically storing their content. The overall *quantity* of online content and the *dynamics* of websites can thus be challenged by careful selection and storage of websites. To address the *mulitmediality* it is necessary to store and code not only the text of websites but also embedded pictures, videos and audiofiles. The same holds true for *hyperlinks*. A further challenge for online content analyses is *personalization*. Online content can be tailored individually using algorithms to

1 This publication was created in the context of the Research Unit "Political Communication in the Online World" (1381), subproject 4 which is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation).

address single users. This individualization can cause enormous problems for automated tools. For example, it is nearly impossible to analyze the placement of single articles as articles can be presented in different order to different users. But online content does not only challenge content analysis. It also provides opportunities like the wide availability of *meta-data* such as comments, likes and shares of articles.

3. Traditional procedures to store online content

Many studies in communication research use procedures like taking screenshots, webcrawling or storing RSS-Feeds to archive online content for content analyses. These procedures vary regarding the degree of automatization. Taking screenshots of websites is the easiest but also the most time-consuming procedure to store online content. While websites mostly appear in the same layout and style and there are formats that make hyperlinks usable, it is impossible to save videos and audiofiles with just a screenshot. Webcrawlers like screenshots save online content in the same layout and style as it appears online. They make hyperlinks available but they also demand manual storage of videos and audiofiles. Furthermore, both procedures do not bypass personalization algorithms. A procedure that can be used to store websites regardless of personalization is the access via RSS-Feeds. These feeds are often automatically created by the content-management-system and list published articles. As all articles appear in the same layout in reverse chronological order, they are not personalized. Unfortunately, the RSS-Feeds do not per se show the articles in the layout and style like they appear online. To store the articles in the outlook of the online versions it is necessary to use the RSS-Feeds as a register and store the articles online versions from there. This short overview shows, that conventional procedures do not address all challenges of websites for reliable content analyses. Therefore it is necessary, to combine these procedures and use tools that best fit the needs regarding the particular research questions.

4. The analysis of journalistic online-content by using automated tools

To address the challenges of online content analyses we want to compare five recently developed tools to store, organize and code online content. These tools are AmCAT, the NewsClassifier, the coding platforms ANGRIST/IN-TOUCH, the Facepager and ARTICLe.

AmCAT[2] combines the organization and coding of online content (van Atteveldt, 2008). It allows to list great amounts of data in a SQL-Database. This data can be analyzed automatically or manually. AmCAT focusses on text formats like XML, RTF or CSV. Thus, AmCAT alone does not address the internet's multimediality and hypertextuality. Furthermore it depends on the procedure of data storage whether AmCAT can be used for content analyses of videos, audio-

2 A documentation of the AmCAT (Amsterdam Content Analysis Toolkit) is available at https://github.com/amcat/amcat. Furthermore, registration to use the tool is possible at http://amcat.vu.nl.

files or hyperlinks. This holds true for bypassing algorithms. The data has to be stored in a way that neutralizes personalization. Besides the organization of data, AmCAT allows various procedures of automated text analysis, like Natural Language Processing (NLP) or Part-of-Speech Tagging (POS). It combines the organization of the coding process, the allocation of the material and the generation of data as well as their export to a statistics software (van Atteveldt, 2008, p. 185). The opportunities to organize, code and generate data lead to a high complexity of the tool. Coding of videos and pictures has to be done manually, as the tool is specialized for text analyses. These disadvantages can be addressed by combining AmCAT with tools for data storage that are appropriate for research questions that focus on multimediality or hypertextuality and by combinations of automatic and manual coding.

The tool NewsClassifier[3] was created to automate the whole process of content analysis from data storage to coding (Scharkow, 2012, p. 250). The tool allows to automatically store journalistic websites. It is possible to access the data via the RSS-Feeds of the websites. Thus, algorithms that personalize content can be bypassed. The data can be stored as HTML or text files. Like AmCAT NewsClassifier focusses on automated text analysis. To organize the coding procedure the tool is able to select a sample of data for automated or manual coding. Manual coding data can be used as training data for the automated coding. Furthermore, NewsClassifier calculates reliability tests and allows exporting data to a statistics software. The disadvantages of the NewsClassifier are similar to those of AmCAT. It is not possible to automatically code content information from pictures, videos or audiofiles. But the automatic storage of the NewsClassifier makes manual coding of such data possible, thus addressing the main challenges of online content analyses.

The tools ANGRIST and IN-TOUCH[4] focus on the coding process itself (Wettstein, 2012; Wettstein, Reichel, Kühne, & Wirth, 2012). They allow computer-assisted half automatic coding. ANGRIST provides a step by step coding along categories within a programmed codebook. The texts for coding are displayed in the tool. Therefore, Unicode or ASCII formats are required. The user interface makes it unnecessary to code single numbers as it provides dropdown menus or checkboxes for coding. The tool IN-TOUCH complements ANGRIST as it is a tool for supervising the coding process. It provides reliability tests and controls the progress of the project. As both are tools for manual coding of text data, they do not per se account for multimediality and hypertextuality. Thus both tools can only complement storage tools if the research questions focus on videos, audiofiles or links.

A tool only for data storage is the Facepager[5] (Keyling & Jünger, 2013). It was developed to collect information from social network sites. It accesses the appli-

3 The tool is available at https://github.com/mscharkow/newsclassifier.

4 A documentation of ANGRIST (Adjustable Non-commercial Gadget for Relational data Input in Sequential Tasks) is available at http://www.ipmz.uzh.ch/Abteilungen/Medienpsychologie/Recource/Angrist.html.

5 The Facepager can be downloaded at https://github.com/strohne/Facepager.

cation programming interface (API) of *Facebook* and *Twitter*. But it can also be used to save information from other JSON-based platforms, like *YouTube*. After adding the Facebook-Feeds or Twitter-Channels that should be collected, the Facepager saves information such as status updates, the number of page likes or the number of comments. The Facepager collects all data that is available from each platforms API. It might thus be insensitive with regards to personalization. To collect data it is necessary to have a user account at the social network sites of interest. The collected data is shown in the user interface and can as well be exported to a statistics software. Multimedia data that is shared in status updates can also be saved and is copied to the local hard disk. The text information is machine readable. The tool can thus be combined with one of the previously described tools for automated or half-automated coding. As the Facepager is developed for social network sites it cannot be used to store or analyze complete websites or articles from websites.

ARTICLe[6] was developed for the automated storage of articles from journalistic websites by Thomas Holbach, Christoph Uschkrat and Jörg Haßler for the DFG funded project "Digital Knowledge Gaps". In contrast to the previous tools it stores articles from websites fully automatic including all multimedia elements like pictures, videos and audiofiles. Furthermore, it stores meta-data such as the likes and shares of an article on social network sites. A third advantage of the tool is that it serves as a coding platform for manual coding. As ARTICLe saves articles via the RSS-Feeds of the websites it is able to bypass algorithms for personalization. Articles are stored as they appear online, as the focus of the database is to provide a platform for manual coding. Therefore screenshots in the formats HTML, PDF and JPG are saved in a relational database. As well as the texts and pictures, videos and audiofiles are collected automatically. The source codes of the articles are searched for keywords. If a keyword appears a regular expression (RegExp) extracts videos and audiofiles and a php-script allows to download these files. All stored articles are saved in a table together with all embedded multimedia files and the meta-data of the articles. This table serves as a coding platform were human coders can select, edit and comment all stored articles. Meta-data like Facebook-likes and –shares can be exported to statistics software. The main advantages of ARTICLe are the presentation of the articles like they appear online. It accounts for the multimediality and hypertextuality of websites and it allows to bypass the personalization of websites. In combination with tools for automated coding ARTICLe might provide a fully automated content analysis of news websites.

5. Conclusion

The growing importance of the internet as a political communication channel as lead to a growing importance of online content analyses. To address the challenges of online content for scientific analyses, like the quantitiy, dynamics, multi-

6 The user interface of ARTICLe (The Automatic RSS-crawling Tool for Internet-based Content analysis) can be accessed at https://article.publizistik.uni-mainz.de/feeds/view.

mediality, hypertextuality and personalization of websites, it is necessary to use tools for data storage. Depending on the research questions there are a few recent tools that address these challenges and allow an automatization of several steps within the process of content analysis. Although there are technical obstacles like flash or javascript applications that are hardly storable, a careful planning of the content analysis and a mindful use of the presented tools allows to automate many working steps of the content analysis.

References

Hasebrink, U., & Schmidt, J.-H. (2013). Medienübergreifende Informationsrepertoires. *Media Perspektiven*, (1), 2–12.

Keyling, T., & Jünger, J. (2013). Facepager (Version, f.e. 3.3). An application for generic data retrieval through APIs. Retrieved from: https://github.com/strohne/Facepager (24.03.2014).

Scharkow, M. (2012). *Automatische Inhaltsanalyse und maschinelles Lernen*. Berlin: epubli.

van Atteveldt, W. (2008). *Semantic network analysis: Techniques for extracting, representing and querying media content*. Charleston, SC: BookSurge.

van Eimeren, B., & Frees, B. (2013). Rasanter Anstieg des Internetkonsums – Onliner fast drei Stunden täglich im Netz: Ergebnisse der ARD/ZDF-Onlinestudie 2013. *Media Perspektiven*, (7-8), 358–372.

Wettstein, M. (2012). angrist.py. Dokumentation und Anleitung für die Programmierung des Codierer-Interface. Retrieved from: http://uzh.academia.edu/MartinWettstein/Papers/965234/angrist.py (24.03.2014).

Wettstein, M., Reichel, K., Kühne, R., & Wirth, W. (2012). IN-TOUCH – ein neues Werkzeug zur Prüfung und Bewertung von Codiereraktivitäten bei der Inhaltsanalyse. Vortrag auf der 13. Jahrestagung der SGKM, Neuchâtel.