Alexander Mehler, Wahed Hemati, Rüdiger Gleim und
Daniel Baumartz

7 VienNA

Auf dem Weg zu einer Infrastruktur für die verteilte interaktive evolutionäre Verarbeitung natürlicher Sprache

Abstract: In diesem Beitrag untersuchen wir Entwicklungstendenzen von Infrastrukturen in den Digitalen Geisteswissenschaften. Wir argumentieren, dass infolge (1) der Verfügbarkeit von immer mehr Daten über sozial-semiotische Netzwerke, (2) der Methodeninflation in geisteswissenschaftlichen Disziplinen, (3) der zunehmend hybriden Arbeitsteilung zwischen Mensch und Maschine und (4) der explosionsartigen Vermehrung künstlicher Texte ein erheblicher Anpassungsdruck auf die Weiterentwicklung solcher Infrastrukturen entstanden ist. In diesem Zusammenhang beschreiben wir drei Informationssysteme, die sich unter anderem durch die Interaktionsmöglichkeiten unterscheiden, die sie ihren Nutzern bieten, um solchen Herausforderungen zu begegnen. Dabei skizzieren wir mit VienNA eine neuartige Architektur solcher Systeme, welche aufgrund ihrer Flexibilität die Möglichkeit bieten könnte, letztere Herausforderungen zu bewältigen.

Keywords: Architekturen, Forschungswerkzeuge, Interoperabilität, Texttechnologie

1 Einleitung

Infrastrukturen für die Digital Humanities (DH) stehen vor Herausforderungen, die in den kommenden Jahren einen erheblichen Anpassungsdruck auf deren

Alexander Mehler, Goethe-Universität Frankfurt, Robert-Mayer Straße 10, D-60325 Frankfurt a. M., E-Mail: mehler@em.uni-frankfurt.de
Wahed Hemati, Goethe-Universität Frankfurt, Robert-Mayer Straße 10, D-60325 Frankfurt a. M., E-Mail: hemati@em.uni-frankfurt.de
Rüdiger Gleim, Goethe-Universität Frankfurt, Robert-Mayer Straße 10, D-60325 Frankfurt a. M., E-Mail: gleim@em.uni-frankfurt.de
Daniel Baumartz, Goethe-Universität Frankfurt, Robert-Mayer Straße 10, D-60325 Frankfurt a. M., E-Mail: baumartz@stud.uni-frankfurt.de

https://doi.org/10.1515/9783110538663-008

Ausgestaltung ausüben dürften. Diese Erwartung steht in Zusammenhang mit vier Entwicklungen, die nahezu alle sprachbezogenen Wissenschaften betreffen:

1. *Online-Kommunikation:* Andernorts wurde bereits vielfach betont, dass Lesen und Schreiben zunehmend online geschehen, immer öfter auch so, dass die Taktrate ihrer Verzahnung sich der mündlichen Kommunikation annähert (Lobin 2014; Beißwenger & Storrer 2008). Von dieser Tendenz sind genuine Webgenres (Mehler, Sharoff & Santini 2010) ebenso betroffen wie webbasierte Auftritte von zuvor klassischen Printmedien (Lobin 2014). Diese bereits weit vorangeschrittene Entwicklung legt es nahe, in vernetzten, webbasierten, kollaborativ erstellten, diskutierten und oftmals fortwährend überarbeiteten sprachlichen Aggregaten erste Adressen für eine massive Erweiterung oder gar Verschiebung[1] des Objektbegriffs textbezogener Wissenschaften zu sehen.
2. *Sozial-semiotische Vernetzung:* Das Paradebeispiel der Wikipedia (Stegbauer 2009) eröffnet in diesem Zusammenhang den Blick auf eine Datenstruktur, die ein ganzes System nicht nur sprachlicher Ordnungsparameter beobachtbar macht. Dies betrifft neben den Entstehungsgeschichten von Schreibprozessen insbesondere jene Agenten bzw. deren *algorithmischen Identitäten* (Cheney-Lippold 2017), die an letzteren, ihrerseits zunehmend vernetzten Prozessen ursächlich beteiligt sind. Diese Beobachtung legt den Schluss nahe, dass die unter Punkt (1) diagnostizierte Gegenstandserweiterung nicht allein auf die Vernetzung intertextueller bzw. -medialer Aggregate bezogen ist, sondern die wechselseitige Konstitution von Sprechergemeinschaften, Kulturen und sprachlichen Subsystemen (vgl. Everett 2013) einbezieht. Diesen wohl nur im Methodenverbund von Soziologie, Psychologie und Informatik zu fassenden Gegenstand adressieren wir mit dem Begriff des *sozialsemiotischen Mehrebenennetzwerks* (Mehler et al. 2018) und wagen die Prognose, dass solche Netzwerke ins Zentrum informationswissenschaftlicher Arbeit rücken werden.
3. *Methodenzuwachs:* Eine dritte Entwicklung geht von der Informatik aus, betrifft aufgrund ihres Erfolges jedoch alle Bereiche, die um die Teilautomatisierung ihrer Forschungstätigkeit bemüht sind. Es geht um die Fortschritte im Bereich neuronaler Netzwerke, um das so genannte *Deep Learning* also, dem bahnbrechende Leistungen bei der automatischen Segmentierung und Klassifikation zu verdanken sind. Kosko (2017: 497) sieht in der gestiegenen Rechenleistung eine wesentliche Ursache für diesen Qualitätssprung, während Hearst (2017: 339) den Zuwachs an verfügbaren

1 In dem Sinne, dass die Akzeleration der internetbasierten Kommunikation zu einer stetigen Anpassung der informationswissenschaftlichen Aufmerksamkeit führen wird.

Daten und Speicherkapazitäten als weitere Ursachen nennt. Dieser Lesart zufolge erleben wir negativ formuliert keinen algorithmischen Qualitätssprung, sondern werden bloß Nutznießer eines Ressourcenzuwachses. Positiv formuliert jedoch lernen Algorithmen aufgrund dieser Entwicklung immer schneller immer komplexere Muster (Kosko 2017). Dessen ungeachtet dürften ihre Auswirkungen enorm sein, und zwar bezogen auf den Zuwachs an verfügbaren Werkzeugen – ob nun in Form von Webservices oder Komponenten von Entwicklungsumgebungen (z. B. Weka) bzw. Programmiersprachen (z. B. R) –, die immer mehr Aufgaben der automatischen Analyse und Annotation sprachlicher Daten übernehmen. Es entwickelt sich sozusagen ein „Überangebot" leicht handhabbarer Werkzeuge, die die Digitalisierung der jeweiligen Disziplin zu forcieren versprechen, vermeintlich ohne *informatics literacy* aufseiten ihrer Anwender einzufordern. Methodologisch gesprochen entstehen auf diese Weise Blackboxes, und zwar vergleichbar jenen, die bereits für das *Data Mining* im Allgemeinen diagnostiziert wurden (Cheney-Lippold 2017). Während also die unter Punkt (1) und (2) adressierten Entwicklungen einen Forschungsgegenstands-bezogenen Anpassungsdruck erzeugen, geht es hier um die Auswirkungen eines geradezu inflationären Zuwachses an digitalen Methoden auch solcher Disziplinen, welche traditionell der Informatik fernstehen. Gewährleistung von Reproduzierbarkeit (Peng 2011) und algorithmischer Transparenz werden daher zu Schlüsselfunktionen zukünftiger Infrastrukturen.

4. *Hybridisierung:* Für sich genommen mögen letztere Entwicklungen bereits herausfordernd sein. Angesichts einer parallelen Entwicklung, die auf unsere Arbeits- und Kommunikationskultur als Ganzes gerichtet ist, gewinnen sie jedoch erheblich an Bedeutung. Dies betrifft zunächst die fortschreitende Hybridisierung der Arbeitsteilung zwischen menschlichen und künstlichen Agenten im Bereich der allgemeinen Zeichenverarbeitung. Unter textanalytischer Perspektive geht es dabei um die Methodenseite textorientierter Disziplinen. Begleitet wird diese Entwicklung von einer geradezu inflationären[2] Erzeugung künstlicher Texte, deren Autoren in erster Linie vollautomatischen, unabhängig von der zugrundeliegenden Software-Plattform agierenden Software-Agenten[3] bzw. Software Robots (Ferrara et al. 2016: 96) oder schlicht Bots (Geiger 2014: 347) entsprechen und also nur mittelbar jenen Personen zuzurechnen sind, welche die zugrundeliegenden Algorithmen entworfen bzw. implementiert haben. Die

2 Zu dieser Einschätzung siehe die Beispiele unten.

3 Geiger (2014) spricht von so genanntem „bespoke code".

hiermit verbundene generative Perspektive betrifft nun die Objektseite textorientierter Disziplinen. Erstere Entwicklung reicht vom Paradigma des *Interactive Machine Learning* (Patel et al. 2010) bzw. des *Human-in-the-loop* (Hoque & Carenini 2015), bei dem Maschinen den Lernprozess implementieren, während Menschen als Innovatoren agieren (siehe Tab. 7.1), bis hin zum *Human Computation* (von Ahn 2008), bei dem Maschinen die Aufgabe der rudimentären Arbeitsorganisation übernehmen (Kosorukoff 2001) und das maschinelle Lernen (wenn überhaupt) nachgeordnet ist. Letztere Entwicklung reicht von *Social Bots* (mit dem Spezialfall der *Conversational Companions* (Wilks et al. 2010) oder *Chatbots*)[4] über *Wiki(pedia) Bots*[5] (Geiger 2014), *News Bots*[6] und *Influence Bots* (Subrahmanian et al. 2016) bis hin zu *Spambots* (Cresci et al. 2017).[7] Am Beispiel von Wikipedia erläutern Geiger & Ribes (2010), inwiefern Bots gar als Agenten eines soziotechnischen Netzwerks aufzufassen sind, das eine Art der verteilten Kognition (Hollan, Hutchins & Kirsh 2000) implementiert, die unter anderem auf die Herausbildung von Qualitätsstandards zielt.

Die eigentliche Bedeutung letzterer Entwicklung betrifft die Erwartung, zukünftig von künstlichen Texten geradezu überschwemmt zu werden, was mehrere Fragen aufwirft:

- Wie nahe ist der Zeitpunkt, ab dem die Wahrscheinlichkeit, einen Text von einem artifiziellen Autor zu lesen, höher sein wird, als einen Text von einem Menschen?
- In welchem Kommunikationsbereich (etwa der Nachrichtenkommunikation (Ferrara et al. 2016; Lokot & Diakopoulos 2016), der Freizeitkommunikation (Ferrara et al. 2016), der Wissenskommunikation (Iosub et al. 2014) oder der Erinnerungskultur (Ferron & Massa 2014)) wird dieser „breakeven point“ zuerst erreicht?

---

4 Bot-Software also, die mit menschlichen Agenten in *Online Sozialen Netzwerken* (OSN) „interagieren“, indem sie deren Verhalten simulieren – ob nun zu deren Schaden oder Nutzen (Boshmaf et al. 2012; Abokhodair, Yoo & McDonald 2015; Freitas et al. 2016; Ferrara et al. 2016). Zu dem zugrundeliegenden Interaktionskonzept siehe kritisch Mehler (2010).

5 Die überwiegend Editionsarbeiten und Vandalismusbekämpfung adressieren (Halfaker & Riedl 2012), teils aber auch ganze Artikel (etwa mittels Übersetzung) erzeugen – siehe unten.

6 Die beispielsweise in Twitter *trending topics* ebenso wie Nischenthemen adressieren können, unter anderem mittels Reproduktion, Aggregation oder datenbezogener Ergänzung (Lokot & Diakopoulos 2016; Steiner 2014). Dabei zeigt sich eine Art Skalenfreiheit, wonach sehr wenige Bots einen sehr großen Anteil an gesendeten Tweets bzw. Links produzieren (Larsson & Hallvard 2015).

7 Bots, die rein repetitive Aufgaben der Organisation von Informationssystemen übernehmen, zählen wir ebenso wenig hierzu wie *Retweet Bots* (Gilani et al. 2016).

Tab. 7.1: Kombinationsmöglichkeiten von menschlichen, maschinellen oder gemischten selektierenden oder innovierenden Agenten. Die innere Mensch-Maschine-Matrix (Szenario 1–4) stammt von Kosorukoff (Kosorukoff 2001).

		Selektierender Agent		
		Maschine	**Mensch**	**Gemischt**
Innovierender Agent	Maschine	Szenario 1 *Distributed Genetic Algorithm* (DGA)	Szenario 2 *Interactive Genetic Algorithm* (IGA)	Szenario 3
	Mensch	Szenario 4 *Computer-Aided Design* (CAD)	Szenario 5 *Human-based Genetic Algorithm* (HbGA)	Szenario 6
	Gemischt	Szenario 7	Szenario 8	Szenario 9 *VienNA*

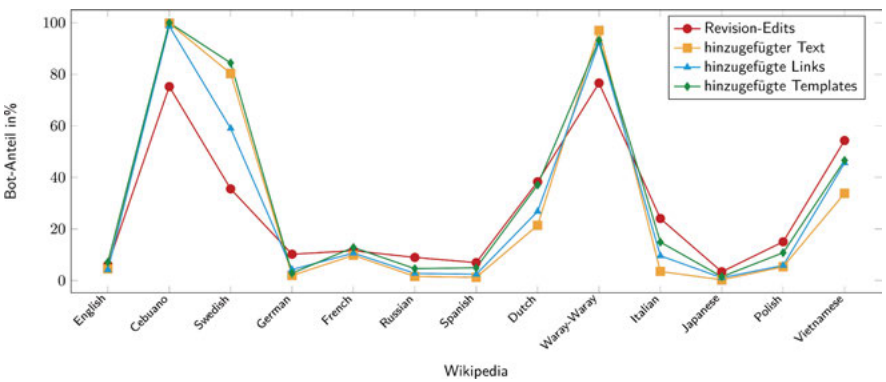

Abb. 7.1: Von Bots autorisierte Anteile an den 13 größten (der Größe nach von links nach rechts geordneten) Wikipedias mit mindestens 1.000.000 Artikeln. Berücksichtigt werden nur Beiträge offiziell registrierter Bots.

– Und was bedeutet diese Entwicklung für die Arbeitsteilung unter den textorientierten Disziplinen? Wird die Informatik hier gar zur dominanten Disziplin, da sie es ist, die Algorithmen zur Erzeugung und Erkennung textgenerierender Bots an erster Stelle erforscht?

Die Wissenskommunikation ist längst in höherem Maße ein Beleg für die mit diesen Fragen umkreiste Entwicklung als es angesichts der für sie einschlägigen Beispiele in Form von Wikipedia und Wiktionary unmittelbar glaubwürdig zu sein scheint. Seien hierzu in Abbildung 7.1 alle Wikipedias betrachtet,

die über mindestens eine Millionen Artikel verfügen. Offenbar stechen zwei Extrembeispiele hervor, und zwar die Wikipedia für *Cebuano* (nunmehr die zweitgrößte Wikipedia) wie auch jene für *Waray-Waray*. In beiden Fällen liegt der Anteil der von Bots autorisierten Texte bzw. Links über 90 % – im Falle von Cebuano werden nahezu 100 % erreicht, durch eine Sprache also, welche eher zu den *low-resource languages* zählt.[8] Mit Vietnamesisch und Schwedisch sind offenbar weitere „Bot-lastige" Wikipedias gegeben. Der Anteilswert von knapp 60 % an Links, den Bots an der schwedischen Wikipedia halten, zeigt z. B., dass die Wahrscheinlichkeit, eine solcherart manifestierte, durch einen Bot erzeugte Textrelation zu rezipieren, den oben erwähnten Break-even-Point bereits überschritten hat. Das bedeutet, dass über intertextuelle Relationen vermittelte, kotextuelle Einbettungen, die maßgeblich Einfluss darauf nehmen, im Kontext welcher Texte ein gegebener enzyklopädischer Artikel oder Textabschnitt rezipiert wird, in solchen Beispielen überwiegend, teils sogar ausschließlich maschinell erzeugt sind. Anders formuliert: Was in welchem textuellen Kontext rezipiert wird, darüber entscheiden immer häufiger Algorithmen und nur noch mittelbar deren Entwickler.[9] Abbildung 7.2 zeigt ergänzend das Beispiel des russischen Wiktionarys, an dem auffällt, dass die mit Abstand aktivsten (an der Kreisunterseite lokalisierten) Agenten abermals Bots sind. In diesem Wiktionary wird lexikalisches Wissen offenbar in hohem Maße durch artifizielle Agenten geformt und vermittelt.

(1) Die Verfügbarkeit von immer mehr Daten über sozial-semiotische Netzwerke, (2) eine Methodeninflation in zuvor rein geisteswissenschaftlichen Disziplinen, die sich methodisch der Informatik annähern, wie auch (3) die zunehmend hybride Arbeitsteilung zwischen Mensch und Maschine und schließlich (4) der Anstieg an artifiziellen Zeichenproduktionen bilden Triebfedern für die Weiterentwicklung von Infrastrukturen, die eine Reihe von informationswissenschaftlichen Anpassungen bzw. Dynamisierungen nach sich ziehen. Um die hiermit verbundenen Diversifikationsmöglichkeiten zu überschauen, knüpfen wir an die Systematik von Kosorukoff (2001) an (siehe Tab. 7.1), die wir um Varianten erweitern, in denen gemischte (künstliche und menschliche) Agenten als Innovatoren (Rekombinatoren) bzw. Selektoren im evolutionstheoretischen Sinne agieren.

8 Letzteres Beispiel ist u. a. dadurch erklärbar, dass ein einzelner, nichtmuttersprachlicher Autor (https://ceb.wikipedia.org/wiki/Gumagamit:Lsj) einen Bot namens *Lsjbot* (https://ceb.wikipedia.org/wiki/Gumagamit:Lsjbot (letzter Zugriff 23. 10. 2017) verantwortet, mittels dessen er Texte in das Wiki einfügt.

9 Man könnte in solchen Beispielen einen Anlass für das Wiederbeleben traditioneller Enzyklopädien erblicken, insofern diese garantieren können, gänzlich von Menschen verfasst zu sein.

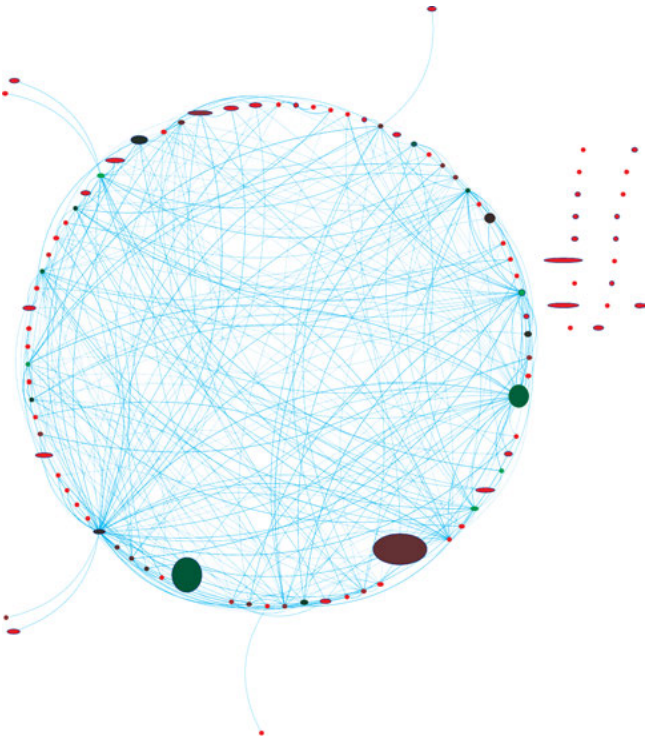

Abb. 7.2: Kollaborationsgraph der 100 aktivsten Agenten des russischen Wiktionarys: Knoten denotieren Agenten, Kanten deren Koautorenschaftsbeziehungen. Die Knotengröße reflektiert den Aktivitätsgrad eines Autors, wobei Knoten von Bots (blau) umrandet sind. Zu den Details der Netzwerk-Berechnung und -Visualisierung, welche auf Brandes et al. (2009) basiert, siehe Mehler et al. (2018).

Gemäß Tabelle 7.1 ist der Standardfall einer Infrastruktur, bei der eine Maschine *M* das jeweilige Inputkorpus analysiert und Menschen die resultierenden Ergebnisse im Zuge ihrer Interpretation evaluieren, eine degenerierte Form eines *Interactive Genetic Algorithm* (IGA), bei dem der Wissenschaftsprozess, in dessen Rahmen es zur Weiterentwicklung bzw. Ersetzung von *M* durch geeignetere Modelle kommt, den entsprechenden Optimierungsprozess implementiert. In Sektion 2 fokussieren wir auf diesen Standardfall und somit auf das gegenwärtig dominante Paradigma von Infrastrukturen.

Indem die Geschwindigkeit von Evaluation und entsprechender Anpassung im Rahmen des zuvor genannten Wissenschaftsprozesses stetig zunimmt, nähern wir uns einem IGA an, bei dem *M* idealerweise selbst zum Gegenstand von Innovationen wird, um immer bessere Analyseresultate aus der Sicht ihrer

menschlichen Interpreten zu erzielen.[10] An dieser Stelle sind Szenarien denkbar, in denen Maschinen den Selektionsprozess unterstützen (Szenario 3) oder umgekehrt Menschen den Innovationsprozess ergänzen (Szenario 8). Beide Szenarien betten Instanzen dessen ein, was Kosorukoff *Human-based Genetic Algorithm* (HbGA) nennt und wofür Wiki-basierte Systeme derzeit die prominentesten Beispiele darstellen (Szenario 5). Dieser Lesart zufolge geht es im Zusammenhang der automatischen Textanalyse bei Szenario 3 und 8 um Ansätze, bei denen Menschen Wikis dazu verwenden, ihre Evaluationen, Korrekturen, Ergänzungen und Interpretationen automatisch erzeugter Analyseresultate zu dokumentieren, so dass diese dazu herangezogen werden können, die zugrundeliegenden Algorithmen zu optimieren. Sektion 3 widmet sich einer Vorstufe dieser Variante von Infrastruktur, und zwar am Beispiel von Wikidition (Mehler et al. 2015).

Die Extremfälle entlang der Hauptdiagonale von Tabelle 7.1, denen zufolge entweder Maschinen (DGA) oder Menschen (HbGA) die Rollen von Innovatoren und Selektoren einnehmen, ruft Szenario 9 auf den Plan, in welchem diese Rollen gemischt besetzt sind. Auf dieses Szenario, dem unserer Auffassung nach die Zukunft wissenschaftlicher Infrastrukturen gehört, fokussiert Beispielgebend Sektion 4. Es geht dabei um eine Architektur von Infrastrukturen für die verteilte interaktive evolutionäre Verarbeitung natürlicher Sprachen, welche auf die optimale Verflechtung von *artificial* und *human computation* zielt. In einem solchen Informationssystem interagieren NLP-Bots als textanalytische Pendants der oben genannten textgenerierenden Bots, und zwar so, dass ihre evolutionäre Weiterentwicklung durch wechselseitige Interaktion (DGA) wie auch durch Interaktion mit ihren menschlichen Nutzern (IGA), die untereinander kommunizieren, um ihre Ziele und Anforderungen weiterzuentwickeln (HbGA), stetig vorangetrieben wird. Szenario 9 zielt folglich darauf, die oben angesprochene Hybridisierung auf die Spitze zu treiben – diesem Szenario dürften zukünftige Infrastrukturen gewidmet sein.

Zusammenfassend gesprochen thematisiert der Beitrag drei aufeinander aufbauende Infrastrukturbeispiele von zunehmender Interaktionskomplexität: Zunächst erläutert Sektion 2 eine Reihe von Bezugsgrößen der Dynamisierung von Infrastrukturen, die von adaptierbaren NLP-Pipelines bis hin zu verteilten Serverarchitekturen reichen. Damit werden zugleich Möglichkeiten und Grenzen des oben genannten Standardfalls benannt. Hierauf aufsetzend thematisiert Sektion 3 die Verzahnung von *Natural Language Processing* (NLP) und Wiki-Prinzip. Zu diesem Zweck wird mit *Wikidition* ein Hybrid-Wiki beschrieben, das

10 Dieses Szenario entspräche – allerdings positiv gewendet – der evolutionären Optimierung von Bots (Cresci et al. 2017).

auf Interaktionsmöglichkeiten im Sinne eines HbGA zielt. Schließlich fokussiert Sektion 4 auf die Aufgabe, sämtliche der zuvor thematisierten Entwicklungstendenzen in einen Anforderungskatalog für Infrastrukturen zu überführen, die auf die optimale Interaktion von Algorithmen und ihren Nutzern zielen. Zu diesem Zweck wird mit VienNA ein Architekturmodell entworfen, das sich dieser Vision – sozusagen mit Blick auf 2020 – annimmt. Sektion 5 fasst die Ergebnisse des Beitrags zusammen und gibt einen Ausblick auf zukünftige Arbeiten.

2 TextImager

Die Verfügbarmachung von immer neuen Methoden für immer mehr Annotationsaufgaben, wie sie in der Einleitung beschrieben wurde, bildet eine Herausforderung für alle bestehenden Infrastrukturen. Folglich bedarf es der Entwicklung flexibilisierter Architekturen, die dieses Wachstum bewältigen können. Dabei ist zu fragen, entlang welcher Dimensionen diese Flexibilisierung zu gewährleisten ist. Zwecks Beantwortung dieser Frage rekurrieren wir auf TextImager (Hemati, Uslu & Mehler 2016; Uslu et al. 2017; Hemati, Uslu & Mehler 2017) als ein Beispiel für Systeme, welche sich an der „klassischen“ und also weitgehend linearen, nicht-interaktiven Arbeitsteilung von zuerst algorithmischer (automatischer) Analyse und nachfolgender (menschlicher) Interpretation der Analyseergebnisse orientieren (Hearst 1999).

TextImager[11] stellt NLP-Pipelines für die automatische Analyse textueller Einheiten bereit. Vergleichbar mit Voyant Tools (Rockwell & Sinclair 2016) geht es dabei um die interaktive Visualisierung textueller Strukturen, und zwar so, dass Interaktionen mit Texten zu Anpassungen der mit ihnen verschränkten Visualisierungen führen. TextImager erlaubt daher die Interaktion mit Visualisierungen, um das Browsen in den Inputtexten zu steuern, wie auch umgekehrt das Browsen in Texten, um kontextsensitive Visualisierungen zu erzeugen. Vergleichbar mit DKPro (Eckart de Castilho & Gurevych 2014) nutzt TextImager Apache UIMA, um NLP-Tools zu orchestrieren. Eine der Aufgaben von TextImager besteht darin, die hiermit verbundenen Kombinationsmöglichkeiten zu erweitern. Vergleichbar mit GATE (Cunningham 2002), RapidMiner (Ertek, Tapucu & Arin 2013) und WebLicht (Hinrichs, Hinrichs & Zastrow 2010) wird das Ziel verfolgt, Text-Mining-Werkzeuge auf einer möglichst breiten Basis verfügbar zu machen. Wie bei ConText (Diesner 2014)

11 http://textimager.hucompute.org/ (letzter Zugriff: 30. 4. 2018).

betrifft dies auch netzwerkanalytische Software (Uslu et al. 2017). Im Prinzip soll jedes frei verfügbare Tool für jede natürliche Sprache, das UIMA-konform standardisiert ist, in TextImager integrierbar sein. Zurzeit werden Werkzeuge für Englisch (44), Deutsch (26), Spanisch (22), Französisch (13), Latein (10), Niederländisch (7) , Portugiesisch (8), Chinesisch (7), Italienisch (5), Dänisch (5), Arabisch (4), Türkisch (3) und Bulgarisch (2) bereitgestellt.

Im Folgenden rekurrieren wir auf TextImager, um die oben eingeforderte Dynamisierung entlang von sechs Kriterien zu skizzieren:

1. *Multi-User-Systeme:* Reproduzierbares datenorientiertes Arbeiten erfordert die Möglichkeit einer arbeitsteiligen Organisation unter Kooperationspartnern, die unterschiedliche Rechte an der Analyse und Annotation der zugrundeliegenden Daten halten können. Ausgehend von den Möglichkeiten, die Betriebssysteme zur Bildung entsprechender Arbeitsgruppen bieten, verwendet TextImager hierzu das Rechteverwaltungssystem des *eHumanities Desktops* (Gleim, Mehler & Ernst 2012). Auf diese Weise ist es prinzipiell möglich, Vererbungshierarchien über Entitäten zu definieren, die Subgraphen tripartiter Graphen über Usern (*wer darf*), Funktionen bzw. Werkzeugen (*was* bzw. *womit*) und Daten(-repositorien) (*worauf*) aufspannen, um etwa Rechte an Unter-Arbeitsgruppen einschränkend oder erweiternd zu vererben. Darüber hinaus ist TextImager auch anonym verwendbar und erlaubt somit Nutzungsmodalitäten, wie sie für Wikis typisch sind.
2. *Multi-Service-Systeme:* Automatisierungsfortschritte werden es obsolet machen, dass der Mensch auf allen Ebenen des NLP arbeitsorganisierend agiert, so dass zukünftig eher selbstorganisierende, selbstlernende Maschinen diese Arbeitsprozesse in Gang halten werden (siehe Sektion 4). Dies dürfte insbesondere den Bereich der so genannten Vorverarbeitung (zu der beispielsweise die Lemmatisierung oder das *PoS-Tagging* zählt) betreffen. Es sind aber auch Zusammenschlüsse bestehender Systeme und Entwicklungsumgebungen denkbar, die Arbeitspakete untereinander austauschen, um Spezialisierungsvorteile zu nutzen. Eingedenk dieser Entwicklung bietet TextImager sämtliche seiner Werkzeuge als Webservices an, die folglich Plattform- und Programmiersprachen-unabhängig nutzbar sind. Das mit dieser Vorgehensweise verbundene Dynamisierungsprinzip bezieht sich folglich darauf, Infrastrukturen unabhängig von HCIs nutzbar zu machen.
3. *Multi-Pipeline-Systeme:* Infolge der oben diagnostizierten Methodenvielfalt stehen für dieselbe Aufgabe (z. B. *Tagging*) immer mehr Algorithmen (z. B. *Conditional Random Fields*) und Werkzeuge (z. B. *MarMoT*) bereit, die ihrerseits Parameterräume aufspannen und somit Entscheidungsalternativen auf zumindest drei Ebenen erzeugen. Zudem hat sich im Zuge der Entwicklung der Computerlinguistik als wissenschaftliche Disziplin ein rudimen-

täres Prozessmodell etabliert, das eingeübte Abfolgeregularitäten solcher Aufgaben festlegt. So setzt etwa das *Semantic Role Labeling* vielfach auf Parsing auf, das wiederum Ergebnisse von Lemmatisierung und PoS-Tagging voraussetzt. Ein solches Prozessmodell erlaubt Abstraktionen auf zumindest vier Ebenen: im Hinblick auf (a) die Klassifikation von Teilaufgaben zum Zwecke der Methodenübertragung (PoS-Tagging und NER als Beispiele für *Sequence Labeling*), (b) die variierende Instanziierung des Prozessmodells zwecks Fokussierung auf Teilaufgaben, (c) die Orchestrierung von Teilmengen von Werkzeugen für dieselbe Aufgabe zur Erzeugung von Skaleneffekten (etwa mittels *majority voting*) und schließlich (d) die Neuordnung von Prozessschritten zwecks Erzielung von Seiteneffekten (wenn etwa Parsing-Ergebnisse dazu verwendet werden sollen, das PoS-Tagging nachzubessern). In diesem Zusammenhang bietet TextImager aufgrund seiner UIMA-basierten Architektur die Möglichkeit, seine Dienste mittels alternativer Pipelines zu ordnen, die zur Laufzeit erstellt werden. Auf diese Weise werden zumindest die Dynamisierungsschritte (b) und (c) adressiert.

4. *Multi-Server-Systeme:* Zuwachs an Ressourcen bedeutet auch, dass immer mehr Rechnerkapazität benötigt wird, um die Arbeit von NLP-Tools auf immer größeren Datenmengen zu orchestrieren. TextImager begegnet dieser Herausforderung durch Verteilung seiner Services auf im Prinzip unbegrenzt viele Server, und zwar so, dass eine $n:m$-Relation von Diensten und Servern entsteht. Dieser Ansatz erlaubt das Ein- und Ausschalten bzw. Readressieren immer neuer Server und Services zur Laufzeit. Infolge dieser Dynamisierung muss keiner der Server mehr zusammen mit dem orchestrierenden System gehostet werden. Infrastrukturen dieser Art können folglich als Schnittstellen zu Multi-Server-Systemen ausgebaut werden, die aufgrund dezentraler Initiativen und Weiterentwicklungen stetig wachsen, ohne das eingangs erwähnte Kapazitätsproblem aufzuwerfen. Das ermöglicht es schließlich, dass Nutzer eigene Dienste, die auf ihren Servern liegen, mit Diensten etwa von TextImager kommunizieren lassen, um die verteilte Erledigung ihrer Annotationsaufgaben mittels dieser Infrastruktur zu verwalten. Dabei geht es auch um nicht frei zugängliche Ressourcen (Korpora, Tools etc.), die auf proprietären Servern verbleiben, während sie im Verbund mit frei zugänglichen Ressourcen orchestriert werden. Darüber hinaus adressiert dieser Ansatz die Verarbeitung von *Big Data*, indem er die verteilte Verarbeitung gleichzeitiger Anfragen einer Vielzahl von Usern ermöglicht.
5. *Multi-Datenbank-Systeme:* Ein Spiegelbild des Methodenzuwachses für immer mehr Annotationsaufgaben besteht in der Diversität resultierender

Repräsentationen ob nun in Form dokumentorientierter (XML-basierter) Baumstrukturen, datenorientierter Graphstrukturen oder numerischer Verteilungen. Um auch in diesem Bereich Skalengewinne zu erzielen, bedarf es eines Multi-Datenbank-Systems, das mehrere Datenbankmanagementsysteme (DBMS), welche für die Eigenheiten solcher Repräsentationen optimiert sind, verwalten kann. So eignen sich beispielsweise Graphdatenbanken wie Neo4j für die Repräsentation simpler gerichteter Graphen, während MongoDB für die Verwaltung dokumentorientierter Strukturen besser ausgerüstet ist. Blazegraph wiederum eignet sich für die Verwaltung RDF-basierter Daten, wofür das Wikidata-Projekt ein anwendungstechnisches Aushängeschild ist. TextImager adressiert auch diesen Aspekt der Dynamisierung. Leitbild hierfür ist jene Offenheit, die bereits Multi-Server-Systeme charakterisiert: Es geht darum, immer neue Datenbanken in das System integrieren zu können, um noch immer bestehende Repräsentationsschranken – etwa im Hinblick auf die Analyse von Hypergraphen – angehen zu können. Die dahinter stehende Erwartung betrifft die Erschließung von Skaleneffekten, welche aus der gleichzeitigen Analyse unterschiedlicher Daten und Annotationen derselben sprachlichen Einheiten resultieren. Dahinter steht die Prognose, dass künftige NLP-Infrastrukturen in diesem Sinne hochgradig multimodal sein werden.

6. *Verzahnung:* Indem sie „*Machine Learning*-lastiger" werden, transformieren sich die Digital Humanities sozusagen zu Computational Humanities (Biemann et al. 2014). Je herausfordernder jedoch die geisteswissenschaftliche Interpretationsarbeit, desto größer der Aufwand der „manuellen" Annotation von Trainings- und Testdaten, die dazu benötigt werden, *Machine Learning*-Verfahren für die Annotation von Textsegmenten zu trainieren, auf die die anvisierten Interpretationen bezogen oder potentiell beziehbar sind. Diese Aufgabe erfordert eine enge Verzahnung von automatischer Analyse und manueller Annotation. Es geht darum, letztere Prozesskette dahingehend zu dynamisieren, dass der Prozess der manuellen Annotation durch das NLP massiv unterstützt wird, etwa durch Recommender-Systeme zur fortlaufenden Generierung von Annotations- und Vervollständigungsvorschlägen, die der Annotator idealerweise nur auszuwählen braucht, um seine Arbeit zu erledigen. Eingedenk dieser Dynamisierungsanforderung integriert TextImager Werkzeuge für die Annotation von rhetorischen und propositionalen Strukturen, um die Verschmelzung von *Human Computation* und *Machine Learning* voranzutreiben. Dabei geht es letztlich um die Aufhebung der statischen Abfolge von manueller Annotation, automatischer Analyse und nachfolgender Interpretation.

Diese sechs Bezugsgrößen skizzieren Entwicklungsperspektiven von Infrastrukturen, die bei aller Unabdingbarkeit noch immer auf expertenseitige Nutzer fokussieren und damit eine *informatics literacy* einfordern, wie sie von der Mehrzahl ihrer Nutzer noch lange nicht (wenn überhaupt jemals) erbracht (werden) wird. Infolge der linearen Ordnung von automatischer Analyse und intellektueller Interpretation lassen solche Systeme zudem jene Form von artifizieller Interaktivität zwischen Menschen und Maschinen vermissen, die auf die stetige Anpassung und Optimierung der involvierten Maschinen gerichtet ist (Mehler 2010). Klassische Systeme wie Voyant Tools oder TextImager sind genauer insofern nicht-interaktiv, als die Art und Weise der Interaktionen, die sie implementieren, nicht die Dispositionen der eingesetzten Algorithmen für zukünftige Interaktionen mit ihren menschlichen Nutzern oder gar untereinander verändert (Mehler 2010). Anders ausgedrückt: Die Verwendung von TextImager setzt keine Lernprozesse aufseiten seiner Algorithmen in Gang. Folglich bedarf es – ganz im Sinne von Kriterium (6) – eines Systems, das sehr viel stärker die Interaktionsmöglichkeiten mit menschlichen Interpreten automatisch erstellter „Interpretations-Vorprodukte“ in den Vordergrund rückt und dabei gleichzeitig der Gefahr methodischer Blackboxes (siehe Sektion 1) begegnet, indem es seinen Rezipienten weitreichende Interaktionsmöglichkeiten bietet, um letztere Informationsangebote zu diskutieren, zu adaptieren und gegebenenfalls zu optimieren. Dies ist die Aufgabe von Wikidition.

3 Wikidition

Oberhalb der Ebene von Einzeltexten geht es bei linguistischen Netzwerken darum, die kohärenzstiftenden Verknüpfungsrelationen ganzer Systeme solcher Einheiten horizontal (d. h. von Einheiten derselben Ebene) wie auch vertikal (d. h. von Einheiten verschiedener Ebenen) sichtbar zu machen, um schließlich Ausblicke auf Prozesse der Sprachevolution im Kontext sozialsemiotischer Netzwerke (siehe Sektion 1) zu gewinnen. Angesichts des in Sektion 1 diagnostizierten Methodenzuwachses ist dabei zu fragen, wie immer neue Methoden nutzbar werden sollen, um solche Vernetzungsrelationen zu identifizieren und zu annotieren, und zwar so, dass Rezipienten die Annotationsergebnisse interaktiv weiterverarbeiten können. Zur Beantwortung dieser Frage rekurrieren wir auf Wikidition (Mehler et al. 2015) als Beispiel für eine Infrastruktur, die die automatische Annotation mit dem Wiki-Prinzip (Leuf & Cunningham 2001) verbindet und folglich eine bei weitem stärkere Verflechtung von *artificial* und *human computation* realisiert, als sie mit TextImager und verwandten Systemen derzeit gegeben ist (siehe Sektion 2). Die Kernfunk-

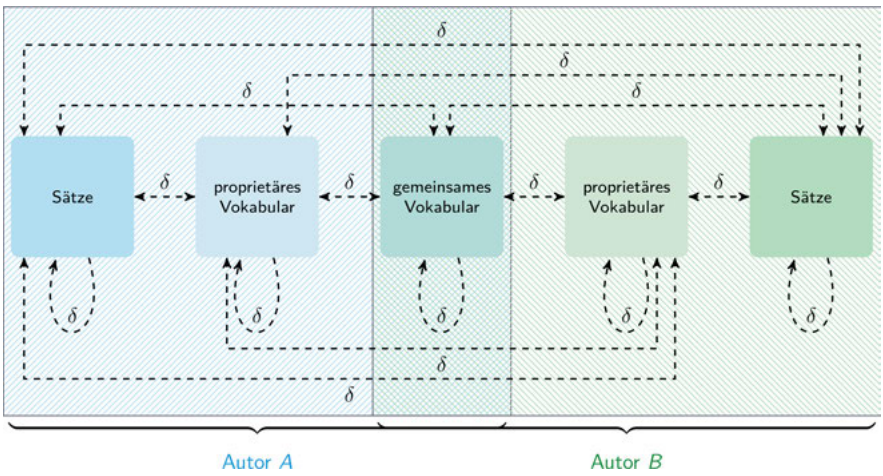

Abb. 7.3: Polypartite Mehrebenen-Netzwerke als das Ergebnis von Wikidition.

tionen von Wikidition lassen sich wie folgt zusammenfassen (zu den Details siehe Mehler et al. 2015):

1. *Linkifizierung:* Ausgehend vom jeweiligen Inputkorpus (Mehler, Wagner & Gleim (2016) unterscheiden neun Varianten solcher Korpora) wird jedes seiner Segmente auf Wort-, Satz- und Textebene durch einen separaten Wiki-Artikel repräsentiert, seine syntagmatischen Kontiguitäts- und paradigmatischen Similaritätsrelationen exploriert und schließlich mittels Hyperlinks manifestiert. Auf diese Weise entsteht ein Mehrebenennetzwerk, in dem jedes Token mit seinem Type und jeder Type mit allen seinen Token verlinkt ist, so dass das resultierende Netzwerk im Sinne kommutierender Diagramme aus jederlei Perspektive (der Wort-, Satz- oder Textebene) in jederlei Richtung traversierbar wird.
2. *Lexikonisierung:* Voraussetzung für die Linkifizierung ist die automatische Extraktion eines Korpus-spezifischen Lexikons, das Wikidition in Form eines eingebetteten Wiktionarys bereithält. Auf diese Weise werden autorenspezifische Eigenheiten der im Inputkorpus enthaltenen Texte zugänglich. Dies bringt Abbildung 7.3 zum Ausdruck, bei der das *gemeinsame Vokabular* auf mehrere Sätze verlinkt, die von mindestens zwei Autoren stammen, während *proprietäres Vokabular* je auf Sätze eines Autors bezogen ist.
3. *Interaktivität:* Da Wikidition im technologischen Sinne ein MediaWiki darstellt, das um Visualisierungsmethoden von TextImager erweitert ist, sind alle seine Inhalte durch alle dazu autorisierten Nutzer änderbar und erwei-

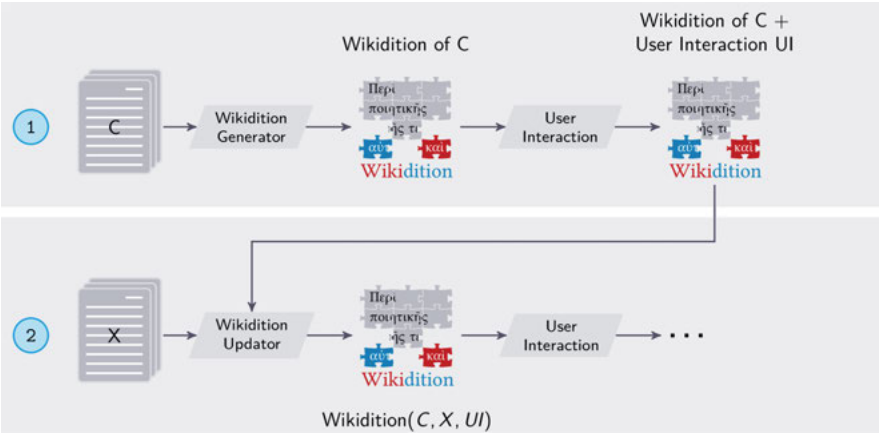

Abb. 7.4: Zur Verschränkung von *artificial* und *human computation* in Wikidition ausgehend von Korpus *C* und seiner Ergänzung um Korpus *X* (vgl. Mehler et al. 2015).

terbar. Das hat den Vorteil, dass alle automatisch erzeugten Annotationen Revisionen unterzogen werden können, und zwar so, dass diese Revisionen nach dem Wiki-Prinzip interaktiv optimiert werden können. Wikidition ermöglicht folglich die Implementation eines *Human-based Genetic Algorithm* im Sinne von Kosorukoff (2001) (siehe Tab. 7.1). Anders als der nachfolgend vorzustellende Ansatz wirkt sich die hierdurch implementierte Interaktivität jedoch nicht auf die fortschreitende Anpassung der involvierten NLP-Routinen aus: Wikidition ermöglicht derzeit daher keine artifizielle Interaktivität im Sinne von Mehler (2010).

4. *Reproduzierbarkeit:* Ergänzt werden die Kernfunktionen durch die gleichzeitige Verfügbarmachung jener Modelle, die zwecks Generierung der Wikidition berechnet wurden. Auf diese Weise werden die betroffenen Modelle auch außerhalb der jeweiligen Wikidition nachnutzbar und reproduzierbar.

Den Ablauf der Erstellung einer Wikidition zeigt Abbildung 7.4: Im ersten Schritt generiert der Wikidition-Algorithmus das oben genannte Mehrebenennetzwerk, das durch nutzerseitige Interaktionen revidiert und erweitert werden kann, ehe Korpora und daraus extrahierte Lexika ergänzt und mit den vorhandenen Ressourcen vernetzt werden. Wikidition ist – wie schon TextImager – sequenziell organisiert: Auf die initiale maschinelle Verarbeitung seiner Textbasis folgt ihre Überarbeitung durch menschliche Agenten, die ergänzen oder korrigieren können, was die maschinelle Verarbeitung unvollständig oder fehlerhaft annotiert hat. Nachfolgend operierende maschinelle Agenten erweitern vorrangig die Textbasis, ohne zuvor erzeugte Annotationen oder ihre eigene

Lernbasis einer maschinellen Revision zu unterziehen. Anders ausgedrückt: Wikidition zielt nicht auf die Optimierung der involvierten NLP-Routinen, sondern auf die andauernde Annotation und Vernetzung der thematisierten Sprachressourcen. Verfahrensinnovationen erfolgen extrinsisch durch Weiterentwicklung von TextImager, auf dem Wikidition basiert. Die verwendeten NLP-Routinen bilden somit einen prozeduralen Input, der mittels einer vorgegebenen Pipeline orchestriert wird. Aus dem Blickwinkel zukünftiger Texttechnologien bietet dieses Szenario eine Reihe von Ansatzpunkten für eine Weiterentwicklung hin zu Verfahren der *verteilten evolutionären Verarbeitung natürlicher Sprachen*, die vor allem auch auf die Optimierung der involvierten NLP-Routinen zielt. Dies wird nachfolgend skizziert.

4 VienNA

Die Grundlage des nun zu skizzierenden Textanalysemodells bildet das Konzept des *Human-based Evolutionary Computing* (HbEC) (Nickerson 2013), das auf menschliche Agenten rekurriert, um Prozessschritte des zugrundeliegenden Evolutionären Algorithmus (EA) zu implementieren bzw. mittels *Human Computation* (HC) (Michelucci 2013) zu ergänzen. Der Vorteil von HbEC besteht in der Integration einer memetischen Komponente (Nickerson 2013), mittels derer die Selektion von Genotypen durch Rekurs auf den Wissens- bzw. Interessenshintergrund von Nutzern gesteuert werden kann. Die Implementation dieser Komponente mittels des für Wikidition konstitutiven Wiki-Prinzips und folglich die Verteilung des HC auf ein Netzwerk interagierender Prosumenten (Tapscott & Williams 2008) – im Folgenden spezieller Wikiditoren genannt – bedeutet dann, dass letzteres Wissen als das Ergebnis eines Wiki-vermittelten Koordinationsprozesses resultiert. Darüber hinaus soll der Koordinationsprozess auf die beteiligten NLP-Komponenten ausgedehnt werden, so dass diese als „eigenständige" Koordinationspartner untereinander ebenso wie mit den beteiligten Prosumenten interagieren. Auf diese Weise soll ein EA implementierbar werden, der Szenario 9 aus Tabelle 7.1 adressiert, in dessen Rahmen alle NLP-Komponenten zu *NLP-Bots* mutieren. Dieser Modellrahmen ist entlang folgender Bezugsgrößen auszuarbeiten:[12]

1. *Integration von Evolutionary und Social Computing*: An erster Stelle geht es um die bereits thematisierte Weiterentwicklung des HbEC zu einem *Distri-*

12 Dabei sprechen wir von Aktualgenese als einem Prozess auf einer kurzfristigeren Zeitskala, und zwar im Verhältnis zur jeweiligen Ontogenese.

buted Human-based Evolutionary Computing (Michelucci 2013) bzw. *Social Computing* (Michelucci 2013), wie es für Wikidition konstitutiv ist.

2. *Ontogenese von NLP-Bots:* An zweiter Stelle geht es um die Dynamisierung der involvierten NLP-Komponenten, die nicht länger einen zwar prozeduralen, aber statischen, im aktualgenetischen Sinne also konstanten Input bilden, sondern selbst einer Evolution unterliegen sollen.
3. *Aktualgenese von Annotationen:* Basierend auf den vorangehenden Dynamisierungsschritten sollen die anvisierten Annotationen der zugrundeliegenden Ressourcen (Texte, Lexika etc.) fortlaufend erzeugt bzw. angepasst werden, und zwar mit dem Ziel ihrer fortgesetzten Optimierung unter Bedingungen, die aus dem Verhalten der beteiligten Prosumenten bzw. kooperierenden oder konkurrierenden Bots resultieren. Annotationsprozesse werden in vergleichsweise kurzen Zeiträumen vollzogen, so dass bei jedesmaliger Verwendung der betroffenen Wikidition deren stetige Veränderung beobachtbar wird.
4. *Genese der Repräsentationen sprachlicher Einheiten:* In Bezug auf letztere Aktualgenese unterscheiden wir zwei Prozessebenen der Repräsentation sprachlicher Einheiten: zum einen die Genese von Merkmalen bzw. Zähleinheiten zu annotierender Objekte und zum anderen die Genese von Annotationseinheiten (als Manifestationen der anvisierten Interpretationen[13]). Man beachte, dass Annotationseinheiten aus der Sicht bestimmter Bots Zähleinheiten aus der Sicht anderer Bots sein können. Um die hiermit verbundene Offenheit des operativen Textrepräsentationsmodells (Mehler 2007) beherrschbar zu machen, gehen wir von einer Grammatik zur Generierung von Datentypen im Rahmen eines verallgemeinerten Klassifikationsszenarios aus (siehe unten).
5. *Genese des Agentennetzwerks:* Die NLP-Bots sind schließlich zu soziotechnischen Gemeinschaften zu integrieren, in denen sie mit ihresgleichen wie auch mit Prosumenten interagieren, und zwar so, dass ihre Orchestrierung selbst zum Gegenstand eines evolutionären Prozesses wird. Auf dieser Grundlage ist zu vermeiden, dass die Orchestrierung etwa in Form einer statischen NLP-Pipeline vorgegeben wird. Das Agentennetzwerk repräsentiert, welcher menschliche oder künstliche Agent mit welchem anderen Agenten auf welche Weise (etwa Daten liefernd oder nachnutzend) interagiert. Das Netzwerk bildet eine emergente Eigenschaft von VienNA, die auf einer Zeitskala zwischen Aktual- und Ontogenese fluktuiert.

13 Im einfachsten Fall der Textklassifikation sind dies Klassenlabels.

6. *Phylogenese von NLP-Bots:* Auf der äußersten (langfristigsten) Zeitskala sind schließlich jene Evolutionsprozesse anzusiedeln, die die Herausbildung von Klassen von NLP-Bots und deren „Abstammungsverhältnisse" betreffen. Dabei handelt es sich um Prozesse, die derzeit noch immer vollständig in Menschenhand liegen, etwa dann, wenn es darum geht, z. B. das Lemmatisieren, PoS-Tagging, *Dependency Parsing* oder das *Argument Mining* als verschiedene, teils interdependente NLP-Aufgaben abzugrenzen und entsprechende Algorithmen zu entwickeln. Aus konzeptioneller Sicht versteht sich unser Ansatz als eine Architektur, in der schließlich auch diese Prozesse zum Gegenstand eines evolutionären Prozesses werden sollen.

Eine NLP-Architektur, die zumindest die Komponenten (1–5) beinhaltet, und also die Ontogenese von Bots und eine hierauf beruhende Aktualgenese von Textrepräsentationen vorsieht, bezeichnen wir als *v*erteilte evolut*i*onär*e n*euronalplastische *N*LP-*A*rchitektur (VienNA). Unter Hinzuziehung eines Modells der Phylogenese von Bots thematisiert VienNA folglich mehrere Zeitskalen, deren Dynamiken zum gegenwärtigen Stand des NLP mit Ausnahmen im Bereich der Aktualgenese und teils der Ontogenese noch immer fest in Menschenhand liegen.

Eine fundamentale Herausforderung der Umsetzung von VienNA bildet die Begrenzung des Textrepräsentations- bzw. Annotationsraums, den die beteiligten NLP-Bots infolge ihrer andauernden Arbeiten aufspannen. Es geht dabei unter anderem darum, zu verhindern, dass die Bots diesen Raum stetig vergrößern – auch im Hinblick auf Annotationen, die die involvierten Wikiditoren positiv evaluieren. Diese Problematik kann am Beispiel der Wikidition aus Sektion 3 erläutert werden, deren lexikalische Einheiten im Schnitt über hunderte Links mit Einheiten derselben oder anderer Sprachebenen verknüpft werden. Würden Bots in einem solchen Szenario weitgehend ungehindert intra- und interrelationale Relationen annotieren, erreichte deren Zahl schnell eine Größe, die die zugrundeliegende DB nicht mehr erfassen kann und aus der Sicht der Prosumenten unzumutbar, da nicht rezipierbar wäre. Folglich ist in VienNA die Selektion von Annotations-Genotypen an eine oder mehrere Fitness-Funktionen zu binden, die unter anderem die Raumkomplexität, die Rezipierbarkeit sowie die Filter- und Kontextualisierbarkeit von Genotypen ebenso reflektiert wie deren Prosumenten-orientierte Interpretierbarkeit, Nützlichkeit[14] oder deren Innovationsgrad. Man kann sich die entsprechen-

14 Was interpretierbar ist, ist nicht notwendigerweise auch nützlich.

den Fitness-Funktionen als ein System vorstellen, das einen *Constraint Satisfaction Process* (CSP) induziert: Erzeugt ein Bot einen Annotationsvorschlag (möglicherweise infolge seiner evolutionären Weiterentwicklung), so hat sich dieser etwa unter Beachtung der für den CSP geltenden Raumkomplexitätsschranke dahingehend zu bewähren, dass er fitter ist als mindestens eine der bereits bestehenden Annotationen, deren Speicherbedarf ersetzungshalber zu beanspruchen wäre. Hierzu kann die Fitnessfunktion auf die sich wandelnden Interessen der Wikiditoren rekurrieren, um eine Erstarrung der Wikidition zu verhindern: fitter ist sozusagen das, was die je aktuellen, prinzipiell aber wechselnden Interessen der Prosumenten besser bedient.

Um VienNA als Instanz eines EA zu konstruieren, der zumindest auf den zwei genannten Zeitskalen operiert, orientieren wir uns an der Wiedergabe eines EA nach Nickerson (2013). Dieser beinhaltet zunächst genau eine Zeitskala evolutionärer Dynamik (siehe Algorithmus 1). Ziel ist es, die Verfahrensschritte eines solchen EA im Sinne der anvisierten Dynamisierung zu konkretisieren. Hierzu ist anzugeben, wie sich die Zeitskalen von VienNA in Algorithmus 1 einbetten lassen. Um angesichts der Vielfalt von NLP-Aufgaben die Machbarkeit der Darstellung zu wahren, konzentrieren wir uns beispielgebend auf zwei Klassen von Aufgaben: Textkategorisierung (Joachims 2002) und Textähnlichkeitsmessung (Bär et al. 2012). Wir beginnen mit der Textkategorisierung, mit deren Hilfe einstellige Relationen gelernt werden.

Algorithmus 1: Schema eines *Evolutionären Algorithmus* (EA) nach Nickerson (2013).

```
erzeuge eine initiale Generation G von Lösungen;
evaluiere G;
while G lässt sich verbessern do
    erzeuge eine Menge K von möglichst vielen Paaren von Elementen
      aus G;
    foreach Paar aus K do
        erzeuge Nachkommen mittels Kombination dieser Paare;
        mutiere die Nachkommen;
        evaluiere die Nachkommen;
    end
    selektiere die nächste Generation aus der Menge der Eltern und
      Nachkommen;
end
return Menge der besten Lösungen;
```

4.1 Einstellige Relationen

Textkategorisierung ist als Konkatenation

$$g \circ f : C \to \mathscr{L} \tag{1}$$

zweier Funktionen aufzufassen, bei der $f : C \to X$ jedem Text(segment) $t \in C$ des Korpus C eine Textrepräsentation (etwa einen Vektor im Sinne des Vektorraummodells) als Element der Repräsentationsmenge X zuordnet, die die Funktion $g : X \to \mathscr{L}$ auf Klassenlabels der Menge $\mathscr{L}$ abbildet. Man beachte, dass die Konstituenten der Repräsentation $f(t)$ nicht t entstammen müssen.[15] f und g werden i. d. R. getrennt implementiert. Beide Funktionen besitzen daher unterschiedliche Parameterräume, die nachfolgend separiert betrachtet werden.

- *Zum Parameterraum von g:* Wir gehen zunächst davon aus, dass g als SVM implementiert wird. Der g-induzierte Parameterraum beinhaltet dann Klassen von Kernels (z. B. *linear*, *polynomial*, ..., *string kernel*, *tree kernel*, *graph kernel*), die jeweils Kernel-spezifische Parameter induzieren. Ein solcher Parameterraum lässt sich unmittelbar zum Gegenstand eines EA machen, dessen Lösungsraum aus Binärvektoren besteht, und zwar indem die Kombination von Lösungspaaren (Algorithmus 1, Zeile 6) ebenso wie deren Mutation (Zeile 7) mittels Bitoperationen implementiert wird, wobei Parameter und Dimensionen solcher Vektoren eineindeutig aufeinander abgebildet sind. Demgegenüber gestaltet sich die Suche in den Räumen jener Parameter als schwieriger, die die Gestalt von g festlegen. Am Beispiel mehrschichtiger Feedforward-Netze lässt sich ein solcher Raum als mehrdimensionales Grid auffassen, in dem zum Zwecke der Optimierung ein *Random Walk* durchgeführt wird. Im zweidimensionalen Fall lassen sich so etwa Kombinationsmöglichkeiten der Parameter *i-ter Layer* und *Anzahl Neuronen* abbilden. Mit Hilfe solcher Zufallsbewegungen lassen sich Rekombinationen und Mutationen im Sinne von Zeile 6 bzw. 7 aus Algorithmus 1 abbilden. Wir unterteilen folglich den Parameterraum von g in den Raum R_1 jener Parameter, die die Klasse von Funktionen (linearer Kernel vs. String-Kernel etc.), denen g angehört, festlegen, und den Raum R_2 jener Parameter, die den Phänotyp von g als Instanz dieser Klasse bestimmen. Rekombinationen und Mutationen in R_1 bedingen dann solche in R_2 – an dieser Stelle fehlt der Platz, dies ausführlich darzustellen.

15 Etwa bei der Merkmalsexpansion in Bezug auf ein Lexikon. Alternativ, wenn t ein Segment eines Texts t' ist, kann $f(t)$ Merkmale aus der Umgebung von t in t' beinhalten.

– *Zum Parameterraum von f:* Aufgabe der Parameteroptimierung ist nun die Erzeugung von Merkmalen zur Charakterisierung von Einheiten aus C. Wir gehen davon aus, dass diese Merkmale mittels einer Grammatik rekursiv aufzählbar sind. Da die Spezifikation einer solchen Grammatik die Grenzen des Beitrags sprengte, beschränken wir uns auf einen skizzenhaften Modellausschnitt. Unser Ansatz besteht darin, (1) Textsegmente als Mengen, Pfade, Bäume oder allgemeinere Graphen zu repräsentieren, um (2) hieraus häufige Teilmengen (z. B. *frequent itemsets*, Aggarwal & Han 2014), Teilsequenzen (z. B. *k*-Skip-*n*-Gramme, Guthrie et al. 2006), Teilbäume (Zaki 2005) oder Teilgraphen (z. B. *motifs*, Milo et al. 2002) zu extrahieren, so dass (3) diese im Folgenden als Motive bezeichneten Muster ihrerseits in ihrer Mengen, Pfad-, Baum- oder Graph-basierten Zusammenhangsstruktur untersuchbar werden, und zwar mittels aggregierter Motive.[16] Die Motive, die auf der jeweiligen Rekursionsstufe extrahiert werden, bilden Merkmale in Form von Ereignissen, die sich aus zugrundeliegenden Zähleinheiten (wie etwa Lexeme) zusammensetzen. Hieraus resultieren Häufigkeitsverteilungen der Motive über den Instanzen des beobachteten Segmenttyps (Merkmalsträger erster Ordnung wie z. B. Sätze), die mit Hilfe von Funktionen (wie Lage, Streuungs-, Konzentrations- oder Gestaltparameter) schließlich auf die Ausprägungen entsprechender Merkmale von Einheiten $x \in C$ des Zieltyps (Merkmalsträger zweiter Ordnung wie z. B. Texte) abgebildet werden können. Dabei müssen Merkmalsträger zweiter Ordnung nicht jene erster Ordnung enthalten[17] und können sogar mit diesen zusammenfallen. In diesem Ansatz spannen unter anderem die Auswahlgesamtheit für Merkmalsträger erster Ordnung, die Länge bzw. Größe von Motiven, Signifikanzschranken für deren Häufigkeit, die Rekursionstiefe zur Berechnung aggregierter Motive sowie Funktionen zur Charakterisierung von Häufigkeitsverteilungen von Merkmalen einen Parameterraum auf, der wieder zum Gegenstand eines *Random Walks* gemacht werden kann, wobei für jede solcherart „aktivierte" Parameterkonstellation die Merkmalsausprägungen der Einheiten aus C zu berechnen sind. Nachfolgend denotieren wir diesen Raum mit R_3. Offenbar ist R_3 derart groß und letztere Berechnungen dermaßen komplex, dass eine Zufallsbewegung, die mit dem ein-

16 Dabei bilden Pfade etwa Tokenfolgen ab, während Bäume zur Abbildung von Dependenzstrukturen und ((un-)gerichtete) Graphen zur Abbildung von Assoziationsnetzwerken sprachlicher Einheiten (z. B. der Wortebene) dienen. Aggregationen von Motiven werden beispielsweise in Mehler (2006) beschrieben.

17 Im umgekehrten Fall werden Merkmale aus der Umgebung eines Segments auf dieses abgebildet. Dies geschieht beispielsweise bei der Lemmatisierung.

fachsten Setting in Form eines *Bag-of-elementary-Features*-Modell) beginnt, die einzig effiziente Möglichkeit seiner Traversierung sein dürfte.

Dieser Ansatz deckt eine recht allgemeine Klasse von NLP-Aufgaben ab, wie z. B. Lemmatisierung, PoS-Tagging, Sentimentanalyse oder Textkategorisierung. Er beinhaltet das klassische *Bag-of-words*-Modell ebenso wie Modelle basierend auf Grammen höherer Ordnung oder auf Mustern in Form von Teilgraphen. Seine Grundidee besteht darin, jedwedes Muster (mengenmäßiger, sequenzieller oder graphenmäßiger Art) auf numerische Merkmalsvektoren abzubilden, die zum Input von Neuronalen Netzwerken ebenso gemacht werden können wie von SVMs. An dieser Stelle stellt sich die Frage, warum man nicht einfach auf Neuronale Netze rekurriert, um das hier skizzierte Merkmalsauswahlproblem zu lösen. Der Grund besteht darin, dass unser Ansatz insofern transparenter ist, als er darauf zielt, Merkmale mittels Bitoperationen explizit „ein- oder auszuschalten", so dass man Lösungen als Ergebnisse entsprechender Optimierungsdurchläufe unmittelbar ablesen kann, welche Merkmalskonstellationen für sie ausschlaggebend waren.

4.2 Binäre Relationen

Anders als die Kategorisierung ist die Ähnlichkeitsmessung insofern *relational*, als entsprechende Funktionen $g \circ f : C \times C \to \mathbb{R}$, $f : C \times C \to X \times X$, $g : X \times X \to \mathbb{R}$, Paare von Textsegmenten auf Ähnlichkeitswerte abbilden. Der Einfachheit halber nehmen wir an, dass diese Werte im Intervall $I = [0,1]$ liegen und sich wie Zugehörigkeitswerte einer unscharfen Menge lesen lassen. Dieses Szenario lässt sich unmittelbar als Kategorisierungsaufgabe rekonstruieren und erlaubt daher den zuvor erarbeiteten evolutionären Zugriff. Dazu besteht die Möglichkeit einer Benennung von Teilintervallen des Intervalls I etwa mit Elementen der Menge $\mathscr{L}' = \{$*ähnlich*, *eher ähnlich*, *unentscheidbar*, *eher unähnlich*, *unähnlich*$\}$, so dass die Ähnlichkeitsmessung als Kategorisierungsfunktion $g' \circ f'$: $C \times C \to \mathscr{L}'$ notierbar wird. Man beachte, dass sich unüberwachte Verfahren der Textähnlichkeitsmessung dazu eignen, die durch $\mathscr{L}'$ benannte Partition mit Beispielen aufzufüllen – andernfalls ist mit intellektuell erzeugten Trainingsbeispielen zu operieren. Auf der Basis einer solchen Rekonstruktion kann nicht nur die Ähnlichkeitsmessung – wie bereits die Textkategorisierung – zum Gegenstand eines EA gemacht werden. Nach diesem Muster können vielmehr alle Klassen von Aufgaben, die binäre Relationen über Textsegmenten lernen (z. B. *Textual Entailment*), zum Gegenstand von VienNA gemacht werden.

4.3 Einbettung

An dieser Stelle sind mehrere Möglichkeiten denkbar, die Optimierung der Funktionen f und g in das Schema von Algorithmus 1 einzubetten. Wir beschränken uns auf die Variante, dass Algorithmen (R_1) und deren Parameter (R_2) gleichzeitig mit den vektoriellen Repräsentationen Input-bildender Texte (bzw. Textsegmente) (R_3) optimiert werden. Wegen der Größe von R_3 ist es wenig sinnvoll, die Suche in diesem Raum jener in R_1 (bzw. R_2) rekursiv unterzuordnen: Die evolutionäre Dynamik bliebe mutmaßlich auf R_3 beschränkt. Es besteht jedoch die Möglichkeit, in den Zeilen 6 und 7 aus Algorithmus 1 Bewegungen in R_3 wahrscheinlicher zu machen. Generationen von Lösungen (Zeile 1) entsprechen dann stets Parameterkonstellationen in allen drei Räumen (R_1 ... R_3) und also Algorithmen und Repräsentationsmodellen, die in mehreren Teilpopulationen organisiert sein können, ob nun im Wettbewerb miteinander oder nicht (etwa dann, wenn sie dieselben oder verschiedene Klassifikationsaufgaben adressieren). Der entscheidende Punkt dabei ist, dass Parameterkonstellationen, die zu Klassifikationen von Textsegmenten und entsprechenden Annotationen führen, selbst zum Gegenstand der Merkmalsbildung im Sinne von Sektion 4.1 und 4.2 werden können.[18] Die Evaluationen der Zeilen 2 und 8 sind schließlich mittels einschlägiger Evaluationsmaße (z. B. *F*-Score) implementierbar, was initial bereitzustellende Trainings- und Testdaten voraussetzt.[19] Zeile 10 bietet den Einstiegspunkt für Wikiditoren, die nun alte oder neue Lösungen verwerfen bzw. auswählen können, was zu Revisionen von Evaluationen aus Zeile 8 führen kann. Die Selektionsfunktion aus Zeile 10 steht genauer in Zusammenhang mit drei Szenarien:

1. *Szenario 1:* Dies betrifft zunächst die bereits erläuterte Variante, bei der Wikiditoren die Selektion durchführen, z. B. zwecks Bedienung einer memetischen Funktion (siehe oben). Dieses Szenario entspricht einem Annotationsspiel, das auf zwei Ebenen überwacht ist: auf der Ebene der eingebauten Fitting-Funktionen (Zeile 2 und 8) und der durch Vorwissen oder Interessen geleiteten Selektion (Zeile 10), wobei durch Wikiditoren getätigte Bewertungen etwa in Form von Korrekturen das zugrundeliegende Trainings- und Testmaterial stetig erweitern bzw. verändern. Ein solches Anno-

18 Dadurch können beispielsweise Annotationen von Wortarten als Ergebnis des PoSTaggings anstelle der betroffenen Lexeme im Hinblick auf sequenzielle Muster betrachtet werden, die entsprechende Tokenfolgen erzeugen.

19 Von den hiermit zusammenhängenden Parameterräumen abstrahieren wir aus Platzgründen. Ferner sei darauf hingewiesen, dass dies zugleich der Ansatzpunkt für die anfangs der Sektion aufgezählten Fitness-Funktionen ist, welche weit über Maße wie *F*-Score hinauszugehen erfordern.

tationsspiel zielt auf folgende Leitfrage: *Inwieweit lösen die involvierten, fortwährend nachjustierten Verfahren und errechneten Repräsentationen die gestellten Aufgaben im Sinne der beteiligten Wikiditoren?*

2. *Szenario 2:* Demgegenüber besteht die Möglichkeit, die Selektion ebenfalls zu automatisieren, was zunächst auf Szenario 1 aus Tabelle 7.1 hinausliefe. Es besteht nun aber die Möglichkeit, die gestellten Annotationsaufgaben als Bots zu konzipieren, die sich variable Manifestationen in den Räumen R_1 ... R_3 selbstlernend suchen und einander dahingehend bewerten, inwieweit sie ihre Aufgabenbewältigung wechselseitig selbstorganisierend unterstützen. Ein solches Annotationsspiel zielt auf folgende Leitfrage: *Zu welchen Annotation führt „das freie Spiel" der Bots, die sich kooperationshalber mit welchen anderen Bots auf welche Weise vernetzen, so dass die Reihenfolge, in der diese agieren, zur variablen Modellgröße wird und nicht mehr als NLP-Pipeline vorgegeben zu werden braucht?*
3. *Szenario 3:* Ein drittes Szenario betrifft Mischformen letzterer beider Annotationsspiele, so dass Wikiditoren neben Bots als Selektoren in Wikidition agieren. Dieses Szenario zielt auf die Beantwortung folgender Leitfrage: *Ab welchem Zeitpunkt besteht die Möglichkeit des Übergangs zu Szenario 2, ohne dass Wikiditoren länger in den Selektionsprozess eingreifen müssen, um die evolutionäre Dynamik jenen Zustand approximieren zu lassen, auf den Szenario 1 zielt?* Szenario 3 adressiert sozusagen den Übergangsbereich von Szenario 1 zu 2.

Idealerweise liefe Szenario 3 darauf hinaus, die Annotation sprachlicher Einheiten, wie sie die Informatik und verwandte Disziplinen bislang unterstützen, soweit zu automatisieren, dass sie als Aufgabenstellung weitgehend erledigt wäre. Angesichts der Offenheit des menschlichen Interpretationsuniversums kann unmittelbar die Unmöglichkeit eines vollständigen Unterfangens dieser Art konstatiert werden. Dies wirft jedoch die Frage auf, wieweit man mit einer solchen Automatisierung gehen kann, inwieweit sich also die Arbeit der Computerlinguistik als wissenschaftliche Disziplin automatisieren lässt.

5 Schlussfolgerung

In diesem Beitrag haben wir vier Entwicklungslinien aufgezeigt, die einen erheblichen Anpassungsdruck auf die Weiterentwicklung von Infrastrukturen für die Digital Humanities erzeugen. Dies nahmen wir zum Anlass, mit Text-Imager, Wikidition und VienNA Beispiele für drei Klassen von Infrastrukturen zu betrachten, die sich unter anderem durch die Interaktionsmöglichkeiten

unterscheiden, die sie ihren Nutzern bieten. Mit VienNA haben wir an dritter Stelle eine neuartige Architektur für eine hochgradig interaktive Infrastruktur skizziert, die auf die vollständige Automatisierung der Annotation sprachlicher Einheiten zielt, so dass Interaktion letztlich die Kooperation selbstorganisierter, selbstlernender NLP-Bots meint. Zukünftige Arbeiten sind einer experimentellen Erprobung dieser Architektur gewidmet.

Literatur

Abokhodair, Norah, Daisy Yoo & David W. McDonald (2015): Dissecting a social botnet: Growth, content and influence in Twitter. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing* (CSCW '15), 839– 851. New York: ACM. doi: 10.1145/2675133.2675208.

Aggarwal, Charu C. & Jiawei Han (Hrsg.) (2014): *Frequent pattern mining*. Cham, Heidelberg, New York u. a.: Springer.

von Ahn, Luis (2008): Human computation. In *IEEE 24th International Conference on Data Engineering (ICDE 2008)*, 1–2. doi: 10.1109/ICDE.2008.4497403.

Bär, Daniel, Chris Biemann, Iryna Gurevych & Torsten Zesch (2012): UKP: Computing semantic textual similarity by combining multiple content similarity measures. In *Proceedings of SemEval '12*, 435–440. Stroudsburg: Association for Computational Linguistics.

Beißwenger, Michael & Angelika Storrer (2008): Corpora of computer-mediated communication. In Anke Lüdeling & Merja Kytö (Hrsg.), *Corpus Linguistics. An International Handbook of the Science of Language and Society*, Kap. 21, 292–309. Berlin/New York: de Gruyter.

Biemann, Chris, Gregory R. Crane, Christiane D. Fellbaum & Alexander Mehler (2014): Computational humanities – bridging the gap between computer science and digital humanities (Dagstuhl Seminar 14301). *Dagstuhl Reports* 4(7), 80–111. doi: 10.4230/DagRep.4.7.80.

Boshmaf, Yazan, Ildar Muslukhov, Konstantin Beznosov & Matei Ripeanu (2012): Key challenges in defending against malicious socialbots. In *Proceedings of the 5th USENIX Conference on Large-Scale Exploits and Emergent Threats* (LEET '12), 12–16. Berkeley: USENIX Association.

Brandes, Ulrik, Patrick Kenis, Jürgen Lerner & Denise van Raaij (2009): Network analysis of collaboration structure in Wikipedia. In *Proceedings of the 18th International Conference on World Wide Web (WWW '09)*, 731–740. New York, NY: ACM.

Eckart de Castilho, Richard & Iryna Gurevych (2014): A broad-coverage collection of portable NLP components for building shareable analysis pipelines. In Nancy Ide & Jens Grivolla (Hrsg.), *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT (OIAF4HLT) at COLING 2014*, 1–11. Dublin: Association for Computational Linguistics and Dublin City University.

Cheney-Lippold, John (2017): *We are data: Algorithms and the making of our digital selves*. New York: New York University Press.

Cresci, Stefano, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi & Maurizio Tesconi (2017): The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race. In *Proceedings of the 26th International Conference on World Wide Web*

Companion WWW '17 Companion, 963–972. Geneva: International World Wide Web Conferences Steering Committee. doi: 10.1145/3041021.3055135.

Cunningham, Hamish (2002): GATE, a general architecture for text engineering. *Computing and the Humanities* 36, 223–254.

Diesner, Jana (2014): ConText: Software for the integrated analysis of text data and network data (paper presented at the Social and Semantic Networks in Communication Research. Preconference at Conference of International Communication Association (ICA)), Seattle.

Ertek, Gurdal, Dilek Tapucu & Inanc Arin (2013): Text mining with RapidMiner. In Markus Hofmann & Ralf Klinkenberg (Hrsg.), *RapidMiner: Data Mining Use Cases and Business Analytics Applications*, 241–264. Boca Raton: CRC Press.

Everett, Daniel (2013): *Die größte Erfindung der Menschheit: Was mich meine Jahre am Amazonas über das Wesen der Sprache gelehrt haben*. DVA.

Ferrara, Emilio, Onur Varol, Clayton Davis, Filippo Menczer & Alessandro Flammini (2016): The rise of social bots. *Communications of the ACM* 59(7), 96–104. doi: 10.1145/2818717.

Ferron, Michela & Paolo Massa (2014): Beyond the encyclopedia: Collective memories in Wikipedia. *Memory Studies* 14(1), 22–45.

Freitas, Carlos, Fabrício Benevenuto, Adriano Veloso & Saptarshi Ghosh (2016): An empirical study of socialbot infiltration strategies in the Twitter social network. *Social Network Analysis and Mining* 6(1), 23.

Geiger, R. Stuart (2014): Bots, bespoke, code and the materiality of software platforms. *Information, Communication & Society* 17(3), 342–356.

Geiger, R. Stuart & David Ribes (2010): The work of sustaining order in Wikipedia: the banning of a vandal. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*, 117–126. ACM.

Gilani, Zafar, Liang Wang, Jon Crowcroft, Mario Almeida & Reza Farahbakhsh (2016): Stweeler: A framework for Twitter bot analysis. In *Proceedings of the 25th International Conference Companion on World Wide Web* (WWW '16 Companion), 37–38. Geneva: International World Wide Web Conferences Steering Committee. doi: 10.1145/2872518.2889360.

Gleim, Rüdiger, Alexander Mehler & Alexandra Ernst (2012): SOA implementation of the eHumanities Desktop. In *Proceedings of the Workshop on Service-oriented Architectures (SOAs) for the Humanities: Solutions and Impacts, Digital Humanities*, 1–6. Hamburg: Digital Humanities 2020. http://www.clarin-d.de/images/workshops/proceedingssoasforthehumanities.pdf (letzter Zugriff: 12. 12. 2017).

Guthrie, David, Ben Allison, Wei Liu, Louise Guthrie & Yorick Wilks (2006): A closer look at skip-gram modelling. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC-2006)*, 1222–1225.

Halfaker, Aaron & John Riedl (2012): Bots and cyborgs: Wikipedia's immune system. *Computer* 45(3), 79–82.

Hearst, Marti A. (1999): Untangling text data mining. In *Proceedings of ACL '99: the 37th Annual Meeting of the Association for Computational Linguistics, University of Maryland*, 3–10. Stroudsburg, PA: Association for Computational Linguistics.

Hearst, Marti A. (2017): eGaia, ein verteiltes, technisch-soziales, geistiges System. In John Brockmann (Hrsg.), *Was sollen wir von künstlicher Intelligenz halten? Die führenden Wissenschaftler unserer Zeit über intelligente Maschinen*, 338–339. Frankfurt a. M.: Fischer.

Hemati, Wahed, Tolga Uslu & Alexander Mehler (2016): TextImager: a distributed UIMAbased system for NLP. In *Proceedings of the COLING 2016 System Demonstrations*, 59–63. Osaka: Association for Computational Linguistics.

Hemati, Wahed, Tolga Uslu & Alexander Mehler (2017): TextImager as an interface to BeCalm. In Martin Krallinger & Alfonso Valencia (Hrsg.), *BioCreative V.5. Proceedings*, 47–53. Madrid: CNIO Centro Nacional de Investigaciones Oncológicas.

Hinrichs, Erhard, Marie Hinrichs & Thomas Zastrow (2010): Weblicht: Web-based LRT services for German. In *Proceedings of the ACL 2010 System Demonstrations* (ACLDemos '10), 25–29. Stroudsburg: Association for Computational Linguistics. http://dl.acm.org/citation.cfm?id=1858933.1858938.

Hollan, James, Edwin Hutchins & David Kirsh (2000): Distributed cognition: Toward a new foundation for human-computer interaction research. *ACM Transaction on ComputerHuman Interaction* 7(2), 174–196.

Hoque, Enamul & Giuseppe Carenini (2015): ConVisIT: Interactive topic modeling for exploring asynchronous online conversations. In *Proceedings of the 20th International Conference on Intelligent User Interfaces* (IUI '15), 169–180. New York: ACM. doi: 10.1145/2678025.2701370.

Iosub, Daniela, David Laniado, Carlos Castillo, Mayo Fuster Morell & Andreas Kaltenbrunner (2014): Emotions under discussion: Gender, status and communication in online collaboration. *PloS ONE* 9(8), 1–23. doi: 10.1371/journal.pone.0104880.

Joachims, Thorsten (2002): *Learning to classify text using support vector machines*. Boston: Kluwer.

Kosko, Bart (2017): Denkmaschinen = alte Algorithmen auf schnelleren Computern. In John Brockmann (Hrsg.), *Was sollen wir von künstlicher Intelligenz halten? Die führenden Wissenschaftler unserer Zeit über intelligente Maschinen*, 338–339. Frankfurt a. M.: Fischer.

Kosorukoff, Alex (2001): Human based genetic algorithm. In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, Band 5, 3464–3469. Tucson, AZ: Institute of Electrical and Electronics Engineers (IEEE). http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=972056 (letzter Zugriff: 12. 12. 2017).

Larsson, Anders Olof & Moe Hallvard (2015): Bots or journalists? News sharing on Twitter. *Communications* 40(3), 361–370.

Leuf, Bo & Ward Cunningham (2001): *The Wiki way. Quick collaboration on the web*. Boston: Addison Wesley.

Lobin, Henning (2014): *Engelbarts Traum: Wie der Computer uns Lesen und Schreiben abnimmt*. Frankfurt a. M.: Campus Verlag.

Lokot, Tetyana & Nicholas Diakopoulos (2016): News bots: Automating news and information dissemination on Twitter. *Digital Journalism* 4(6), 682–699.

Mehler, Alexander (2006): In search of a bridge between network analysis in computational linguistics and computational biology – a conceptual note. In Hamid R. Arabnia & Homayoun Valafar (Hrsg.), *Proceedings of the 2006 International Conference on Bioinformatics & Computational Biology (BIOCOMP '06), June 26, 2006, Las Vegas*, 496–500. Las Vegas: World Academy of Science.

Mehler, Alexander (2007): Compositionality in quantitative semantics. A theoretical perspective on text mining. In Alexander Mehler & Reinhard Köhler (Hrsg.), *Aspects of Automatic Text Analysis* (Studies in Fuzziness and Soft Computing), 139–167. Berlin, New York: Springer.

Mehler, Alexander (2010): Artifizielle Interaktivität. Eine semiotische Betrachtung. In Tilmann Sutter & Alexander Mehler (Hrsg.), *Medienwandel als Wandel von Interaktionsformen – von frühen Medienkulturen zum Web 2.0*, 107–134. Wiesbaden: Verlag für Sozialwissenschaften.

Mehler, Alexander, Rüdiger Gleim, Tim vor der Brück, Wahed Hemati & Tolga Uslu (2015): Wikidition: Automatic lexiconization and linkification of text corpora. Submitted.

Mehler, Alexander, Rüdiger Gleim, Wahed Hemati & Tolga Uslu (2018): Skalenfreie online soziale Lexika am Beispiel von Wiktionary. In Stefan Engelberg, Henning Lobin, Kathrin Steyer & Sascha Wolfer (Hrsg.), *Jahrbuch des Instituts für Deutsche Sprache 2017*, Berlin: de Gruyter.

Mehler, Alexander, Serge Sharoff & Marina Santini (Hrsg.) (2010): *Genres on the web: Computational models and empirical studies*. Dordrecht: Springer.

Mehler, Alexander, Benno Wagner & Rüdiger Gleim (2016): Wikidition: Towards a multilayer network model of intertextuality. In *Proceedings of Digital Humanities 2016*, Kraków: Alliance of Digital Humanities Organizations.

Michelucci, Pietro (2013): Synthesis and taxonomy of human computation. In Pietro Michelucci (Hrsg.), *Handbook of Human Computation*, 83–86. New York: Springer.

Milo, R., S. Shen-Orr, S. Itzkovitz, N. Kashtan & D. Chklovskiiand U. Alon (2002): Network motifs: Simple building blocks of complex networks. *Science* 298(5594), 824–827.

Nickerson, Jeffrey V. (2013): Human-based evolutionary computing. In Pietro Michelucci (Hrsg.), *Handbook of Human Computation*, 641–648. New York: Springer.

Patel, Kayur, Naomi Bancroft, Steven M. Drucker, James Fogarty, Andrew J. Ko & James Landay (2010): Gestalt: Integrated support for implementation and analysis in machine learning. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology* (UIST '10), 37–46. New York: ACM. doi: 10.1145/1866029.1866038.

Peng, Roger D. (2011): Reproducible research in computational science. *Science* 334(6060), 1226–1227.

Rockwell, Geoffrey & Stéfan Sinclair (2016): *Hermeneutica: Computer-assisted interpretation in the humanities*. Cambridge, MA: MIT Press.

Stegbauer, Christian (2009): *Wikipedia: Das Rätsel der Kooperation*. Wiesbaden: Verlag für Sozialwissenschaften.

Steiner, Thomas (2014): Telling breaking news stories from Wikipedia with social multimedia: A case study of the 2014 winter olympics https://arxiv.org/abs/1403.4289 (letzter Zugriff: 12. 12. 2017).

Subrahmanian, V. S., Amos Azaria, Skylar Durst, Vadim Kagan, Aram Galstyan, Kristina Lerman, Linhong Zhu, Emilio Ferrara, Alessandro Flammini & Filippo Menczer (2016): The DARPA Twitter bot challenge. *Computer* 49(6), 38–46.

Tapscott, Don & Anthony D. Williams (2008): *Wikinomics: How mass collaboration changes everything*. New Jork: Portfolio.

Uslu, Tolga, Wahed Hemati, Alexander Mehler & Daniel Baumartz (2017): TextImager as a generic interface to R. In *Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2017)*, 17–20. Valencia: Association for Computational Linguistics.

Wilks, Yorick, Roberta Catizone, Simon Worgan, Alexiei Dingli, Roger Moore, Debora Field & Weiwei Cheng (2010): A prototype for a conversational companion for reminiscing about images. *Computer Speech and Language* 25, 140–157.

Zaki, Mohammed J. (2005): Efficiently mining frequent trees in a forest: Algorithms and applications. *IEEE Transactions on Knowledge and Data Engineering* 17(8), 1021–1035. doi: 10.1109/TKDE.2005.125.