Aufdecken von “versteckten” Einflüssen: Teil-Automatisierte Textgenetische Prozesse mit Methoden der Computerlinguistik und des Machine Learning

Ullrich, Sabine
sabine.ullrich@campus.lmu.de
Ludwig-Maximilians-Universität München, Deutschland

Bruder, Daniel
dmb77@cam.ac.uk
Universität Cambridge, Vereinigtes Königreich

Hadersbeck, Maximilian
maximilian@cis.uni-muenchen.de
Ludwig-Maximilians-Universität München, Deutschland

Einleitung

Am Beispiel von Ludwig Wittgensteins Nachlass (Pichler et al. 2009; Wittgenstein 1996) wird ein Tool vorgestellt und erläutert, welches die Digital Humanities durch einen computer-unterstützten Prozess für textgenetische Aufgaben bereichern soll.

Ludwig Wittgensteins Nachlass umfasst etwa 20.000 Seiten, in welchen er eine Menge Zitate aus der Weltliteratur verwendet, diese aber nicht notwendigerweise explizit als solche kennzeichnet. Es scheint, als verzichte Wittgenstein auf explizite Quellenangaben bei Autoren, von welchen er annimmt, sie seien Teil eines „allgemeinen” „kulturellen Horizonts“, und lässt bei seinen Zitaten in Fällen wie z.B. Goethe die Namen der Autoren weg.

In einer sinnvollen Zusammenarbeit im Rahmen der Digital Humanities kann die Informatik im Allgemeinen und die Computerlinguistik im Speziellen der Philologie unterstützende Werkzeuge anbieten, die dem Geisteswissenschaftler helfen sollte, derartige mühevolle Prozesse der Zitat-Aufdeckung teil-automatisieren zu können.

In diesem Poster wird ein Tool vorgestellt, welches dem Philologen mit Methoden des Machine Learning beim Aufspüren von Einflüssen aus Zitaten in textgenetischen Prozessen unterstützen kann, indem es erlaubt, „ähnliche” Textabschnitte, d.h. potentielle Zitate, vorzufiltern und zu sortieren, um diese im nächsten Schritt einer genaueren Untersuchung zu unterziehen.

Bezug zum aktuellen Forschungsstand

Die Plagiatsaufdeckung ist die Bestimmung von „ähnlichen“ Texten, d.h. die Aufdeckung von (ungekennzeichneten) Zitaten. Eine Möglichkeit zur Aufdeckung von Ähnlichkeiten ist dabei der Vergleich von syntaktischen Merkmalen zweier oder mehrerer Texte. Diese Charakteristika umfassen beispielsweise die Berücksichtigung von Part-of-Speech Tags, Lemmata und Wortpositionen (Ekbahl et al. 2012). Um zusätzlich modifizierte, aber dennoch semantisch gleiche Texte zu identifizieren, müssen auch Synonyme in Betracht gezogen werden (Abdalgader et al. 2010). Eine Vorverarbeitung, welche u.a. das Tokenisieren der Texte, die Entfernung von Stoppwörtern und eine Sprachidentifizierung beinhaltet, hilft zudem, redundante Wörter zu ignorieren und damit genauere Ergebnisse zu erzielen. Weitere Methoden zur Erkennung von ähnlichen Texten beinhalten u.a. subjektbasierte Graphen (Tomita et al., 2004) und *Document Fingerprinting* (Sadowski and Lewin, 2007; Kent et al. 2010). Überraschenderweise wurde bislang kein Versuch unternommen, die oben genannten linguistischen Features aus dem Bereich der Syntax und der Semantik zu kombinieren, um die Performanz der Ähnlichkeitssuche weiter zu verbessern. Dass eine solche Kombination sinnvoll ist und besonders gute Ergebnisse leisten kann, zeigt Ullrich (2017).

Forschungsproblem

Solche Ansätze könnten – durch entsprechende Umwidmung und in koordinierter Zusammenarbeit mit Philologen – in den Digital Humanities sinnvoll zur Anwendung gebracht werden, um textgenetische Prozesse teil-automatisiert zu unterstützen und Zeit für intensivere Analysen zu gewähren.

Die Idee ist, das bestehende Tool zur Ähnlichkeitsbestimmung aus Ullrich (2017) so umzufunktionieren, dass nicht nur die Ähnlichkeit *einer* gegebene Texteingabe mit *einer* anderen gegebenen Eingabe bestimmt werden kann, sondern eine *Sortierung* (*ranking*) von ähnlichen Texten vorgenommen werden kann. Zunächst werden dafür die Texte in kürzere Abschnitte – bei Wittgenstein „Bemerkungen“ genannt – geteit. Nun wird eine *Sammlung* dieser Abschnitte mit einer anderen *Sammlung* an Abschnitten verglichen, um dann die potentiell Ähnlichsten in einer Art Hitliste auszugeben. Eine derartige Vorsortierung könnte es dem Philologen besonders erleichtern, potentielle Zitate, Einflüsse und Verweise eines Autors innerhalb seines Werkes und im Bezug auf die Literatur seiner Zeit aufzuspüren.

Es muss betont werden, dass mit einem derartigen Werkzeug die Arbeit des Philologen lediglich unterstützt aber keinesfalls ersetzt werden kann. Im Wesentlichen erlaubt eine teil-automatisierte Vorfilterung von „ähnlichen" Textstellen eine drastische Reduktion des „Suchraums".

Um die Leistung eines Philologen wie Hans Biesenbach (2014) zu illustrieren: Wittgensteins Nachlass umfasst 20.000 Seiten. Rechnet man mit 5 Bemerkungen pro Seite und die angestrebte Ähnlichkeitssuche beschränkt sich auf 20.000 Seiten der „Weltliteratur" mit ebenfalls 5 Abschnitten pro Seite, dann gäbe es mathematisch 100.000 x 100.000, also 10 Milliarden mögliche Beeinflussungen, die manuell zu prüfen wären. Selbst für besonders leistungsfähige Rechner werden hier Grenzen erreicht, die nach geeigneten NLP Methoden verlangen.

Angewendete Methode

Mit Hilfe computerlinguistischer Methoden berechnet man für jeden Abschnitt eines Textes seinen „charakteristischen" Vektor oder, intuitiv gesprochen, seinen linguistischen „Fingerabdruck". Dieser automatisierte Prozess kann unabhängig im Voraus berechnet werden um spätere Prozesse zu vereinfachen und beschleunigen. Dieser „Fingerabdruck" beinhaltet die oben genannten syntaktischen, sowie semantischen Informationen.

Wird eine Suchanfrage zum Auffinden ähnlicher Abschnitte gestartet, lässt sich der charakteristische Vektor des eingegebenen Abschnitts berechnen und daraufhin die Vektoren mit dem geringsten Abstand im multi-dimensionalen Raum bestimmen. Diese Vektoren verweisen auf die ähnlichsten Textabschnitte, die dann dem Philologen zur genaueren Prüfung in einer Hitliste vorgeschlagen werden.

Die bereits erfolgreiche Ähnlichkeits *bestimmung* in Ullrich (2017) soll zu einem Ähnlichkeits *rankingtool*weiterentwickelt werden, um sie vor allem für textgenetische Prozesse in digitalen Editionen nutzbar zu machen. Sobald diese Weiterentwicklung abgeschlossen ist, soll sie die Wittgenstein Advanced Search Tools (Hadersbeck et al. 2014) in der Suchmaschine WiTTFind (siehe: http://wittfind.cis.lmu.de) erweitern, welche am Centrum für Informations- und Sprachverarbeitung der Universität München entwickelt wurde.

Bibliographie

Abdalgader, Khaled / Skabar, Andrew(2010): "Short-text similarity measurement using word sense disambiguation and synonym expansion.", in: *Australasian Joint Conference on Articial Intelligence*, Springer 435-444.

Biesenbach, Hans (2014): *Anspielungen und Zitate im Werk Ludwig Wittgensteins*, Sofia University Press.

Ekbal, Asif / Saha, Sriparna / Choudhary, Gaurav (2012): "Plagiarism detection in text using vector space model.", in: *Hybrid Intelligent Systems* (HIS), 2012 12th International Conference on, pages 366-371. IEEE.

Hadersbeck, Maximilian / Pichler, Alois / Fink, Florian / Gjesdal, Øyvind, L. (2014): *Wittgenstein's Nachlass: WiTTFind and Wittgenstein Advanced Search Tools (WAST).* DATeH. Madrid.

C.K. Kent / N. Salim (2010): „Features based text similarity detection.", in: *Journal of Computing*, 2 (1).

Pichler, Alois / Krüger, Heinz W. / Smith, D. / Bruvik, Tone / Lindebjerg, Anne / Olstad, Vemund (Hrsg.) (2009): *Wittgenstein Source Bergen Facsimile (BTE).* Wittgenstein Source Bergen.

Sadowski, Caitlin / Levin, Greg (2007): *Simhash: Hash-based similarity detection.*

Tomita, Junji / Nakawatase, Hidekazu / Ishii, Megumi (2004): „Calculating similarity between texts using graph-based text representation model", in: *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 248-249. ACM.

Ullrich, Sabine (2017): *Evaluation of Existing Plagiarism Research for the Optimisation of NLP-based Similarity Detection using Ludwig Wittgenstein's Remarks*, Bachelor thesis, Ludwig-Maximilians-Universität München.

Wittgenstein, Ludwig / Nedo, Michael (Hrsg.) (1996): *Wiener Ausgabe*. Band 1-5.