

# Einfaches Topic Modeling in Python - Eine Programmbibliothek für Preprocessing, Modellierung und Analyse

**Jannidis, Fotis**

fotis.jannidis@uni-wuerzburg.de  
Universität Würzburg, Deutschland

**Pielström, Steffen**

pielstroem@biozentrum.uni-wuerzburg.de  
Universität Würzburg, Deutschland

**Schöch, Christof**

christof.schoech@uni-wuerzburg.de  
Universität Würzburg, Deutschland

**Vitt, Thorsten**

thorsten.vitt@uni-wuerzburg.de  
Universität Würzburg, Deutschland

Topic Modeling ist eine Methode zur semantischen Erschließung größerer Textsammlungen, die in den letzten Jahren zunehmend in den Fokus der Aufmerksamkeit digital arbeitender Literaturwissenschaftler gerückt ist. Die Methode nutzt probabilistische Verfahren um aus einer Textsammlung eine Reihe von Verteilungen über die Wahrscheinlichkeiten einzelner Wörter zu erzeugen. Diese werden dann als distinkte semantische Gruppen, sogenannte 'Topics', aufgefasst, also als Gruppen inhaltlich zusammenhängender Wörter, die in den einzelnen Texten jeweils mehr oder weniger stark präsent sind (Blei 2012, Steyvers und Griffiths 2006).

Ursprünglich entwickelt, um in größeren Sammlungen kürzerer Fachartikel schnell jene zu identifizieren, die für bestimmte Themen relevant sein könnten, kann diese Methode darüber hinaus für eine Reihe von Problem im Bereich der digitalen Literaturwissenschaft interessante neue Lösungsansätze bieten. Dazu gehört die automatische Identifikation von Romanen, die ähnliche Themen behandeln (wenngleich eine direkte Gleichsetzung probabilistischer 'Topics' mit literarischen 'Themen' durchaus problematisch ist), ebenso wie die Zuordnung zu bestimmten Genres anhand inhaltlicher Aspekte, oder die quantifizierende Betrachtung der zu- und abnehmenden Bedeutung einzelner Themenfelder über den Verlauf eines einzelnen Romans (vgl. Blevins 2012, Jockers 2011, Rhody 2012, Schöch in Vorbereitung).

Mit den Programmen 'Mallet' (vgl. McCallum 2002) und 'Gensim' (vgl. Rehurek 2010) stehen zur Zeit zwei State-of-the-Art Implementierungen von Topic Modeling-Algorithmen zur Verfügung. Um die Methode produktiv einzusetzen, sind aber neben der Erzeugung des Modells weitere Arbeitsschritte notwendig (Abb. 1). Im 'Preprocessing' gilt es zunächst, die Textsammlungen in eine Form zu bringen, in der sie vom Modellierungsprogramm verarbeitet werden können. Darüber hinaus werden die Texte normalerweise durch das Herausfiltern häufiger Funktionswörter auf die potentiell inhaltsrelevanten Wörter reduziert, was in der Regel den vorhergehenden Einsatz von NLP-Tools (Natural Language Processing) erfordert. Sind die 'Topics' dann erst einmal errechnet worden, kann sich eine Visualisierung der Ergebnisse anschließen, oder ihre statistische Evaluierung anhand interner oder externer Kriterien, ein Aspekt dem beim Einsatz von Topic Modeling-Verfahren im DH-Kontext bisher eher zu wenig Beachtung geschenkt wurde.

Ziel unseres Projektes ist es, den Einstieg in aktuelle Topic Modeling-Verfahren für digital arbeitende Literaturwissenschaftler wesentlich zu vereinfachen, indem wir möglichst viele der notwendigen Arbeitsschritte in einer einheitlichen, umfangreichen und gut dokumentierten Programmbibliothek für die unter digital-quantitativ arbeitenden Geisteswissenschaftlern stark verbreitete Programmiersprache Python anbieten. Hierbei sollen Nutzerinnen und Nutzer bei allen Arbeitsschritten auf vorhandene, in einem ausführlichen Tutorial dokumentierte Funktionen zurückgreifen und so weit wie möglich wie mit einem Kommandozeilentool arbeiten können, ohne selbst programmieren zu müssen. Die Anforderungen an die Programmierkenntnisse der Forschenden, die diese Verfahren einsetzen möchten, werden damit minimiert und die Methode wird so einem größeren Nutzerkreis zugänglich gemacht.

Für das NLP-Preprocessing steht mit dem DARIAH-DKPro-Wrapper (DDW) ein komfortables Einheitswerkzeug zur Verfügung, das ein großes Spektrum an NLP-Aufgaben abdeckt und linguistische Annotationen in einem Python-Pandas-kompatiblen Ausgabeformat erzeugt. Ein Ziel unserer Bibliothek ist die direkte Anbindung des DDW-Outputs an existierende Implementierungen verschiedener etablierter Varianten von Topic Modeling-Algorithmen.

Für die Untersuchung der resultierenden Modelle möchten wir verschiedene Evaluierungsverfahren anbieten, sowohl interne Verfahren wie z.B. das Perplexity-Maß, als auch externe Verfahren, wie z.B. die Weglänge zwischen zwei Begriffen in einem Wörterbuch. Hieran schließen sich verschiedene Optionen zur Visualisierung der Ergebnisse an.

Im Fokus der Entwicklung steht die Gestaltung schlüssig aufeinander aufbauender Programmbefehle, die einer einheitlichen Syntax folgen und deren Funktion sich schnell erschließen lässt. Sie sollen sich ohne längere Einarbeitung nutzen und zu einer Pipeline zusammenfügen lassen, die die spezifischen Arbeitsschritte eines bestimmten Topic

Modeling-Projektes umsetzt. Hierbei können Nutzerinnen und Nutzer auf detaillierte Anleitungen aus einem umfangreichen Tutorial zurückgreifen, in dem alle Funktionen, alle Outputs, und potentielle Kombinationen detailliert dokumentiert und anhand von Beispielen erläutert werden.

Die Entwicklung der Programmbibliothek kann auf Erfahrungen mit einer vorhandenen, Python-basierten Implementierung eines entsprechenden Workflows aufbauen, die allerdings eher "proof of concept"-Character hat (Topic Modeling Workflow "tmw", vgl. Schöch 2015 und <http://github.com/cligs/tmw> ).

Language Text Processing System: CRITIQUE“, in: *Proceedings of the Second Conference on Applied Natural Language Processing* 195–202.

**Schöch, Christof** (in Vorbereitung): „Topic Modeling Genre: An Exploration of French Classical and Enlightenment Drama“, in: *DHQ: Digital Humanities Quarterly* <http://digitalhumanities.org/dhq> . Preprint: <https://zenodo.org/record/48356> .

**Steinberger, Mark / Griffiths, Tom** (2006): „Probabilistic Topic Models“, in: Landauer, T. / McNamara, D. / Dennis, S. / Kintsch, W.: *Latent Semantic Analysis: A Road to Meaning*. Laurence Erlbaum.

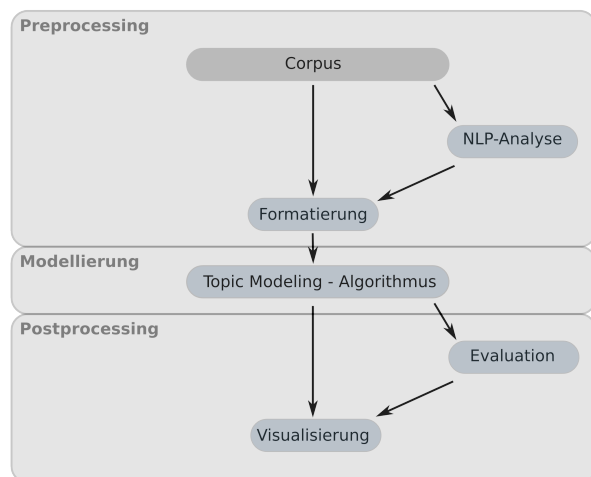


Abbildung 1: Workflow eines Topic Modeling-Projektes

## Bibliographie

**Blei, David M.** (2012): „Probabilistic Topic Models“, in: *Communication of the ACM* 55 (4): 77–84 [10.1145/2133806.2133826](https://doi.org/10.1145/2133806.2133826).

**Blevins, Cameron** (2010): „Topic Modeling Martha Ballard’s Diary“, in: *Historying* . <http://historying.org/2010/04/01/topic-modeling-martha-ballards-diary/> .

**Jockers, Matthew L.** (2013): *Macroanalysis - Digital Methods and Literary History*. Champaign, IL: University of Illinois Press.

**McCallum, Andrew K.** (2002): *MALLET: A Machine Learning for Language Toolkit* <http://mallet.cs.umass.edu> .

**Rehurek, Radim / Sojka, Petr** (2010): „Software framework for topic modelling with large corpora“, in: *Proceedings of LREC 2010*.

**Rhody, Lisa M.** (2012): „Topic Modeling and Figurative Language“, in: *Journal of Digital Humanities* 2 (1) <http://journalofdigitalhumanities.org/2-1/topic-modeling-and-figurative-language-by-lisa-m-rhody/> .

**Richardson, Stephen D. / Braden-Harder, Lisa** (1988): „The Experience of Developing a Large-Scale Natural