Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2016"

Moscow, June 1–4, 2016

TOPIC MODELING OF SCHOLARLY ARTICLES: INTERACTIVE TEXT MINING SUITE

Scrivner O. (obscrivn@indiana.edu),
Davis J. (majdavis@iu.edu)

Indiana University, Bloomington, IN, USA

Access to a large amount of scholarly publication presents new opportunities to researchers. Recent advances in data visualization techniques allow for automated content analysis, topic modeling and classification as well as research trend and scientific network analyses. Many probabilistic models and text-mining tools have been put forth to help analyze and explore large collections of documents. The use of these tools in mainstream scholarly research, however, remains limited to machine learning and natural language processing fields, as the researcher is often challenged by the technological hurdle of command-line tools. We address this issue by introducing a user-friendly application that allows researchers to visually explore scholarly articles. Written in R with the Shiny web app, this application not only provides a web-interactive interface, but also allows researchers to implement state-of-the-art topic modeling and visualization tools. Finally, the accessibility of our web application facilitates data analysis, as researchers are not constrained by memory limitation or platform dependency.

Keywords: topic modeling, data mining, content analysis, visualization

1. Introduction

With the increasingly large collections of digital scholarly publications, text browsing and keyword searches have become both time-consuming and inefficient for content analysis. In contrast, text-mining has evolved into a powerful tool for automated content analysis. In recent years, many probabilistic models and algorithms have been put forward to automatically analyze large collections, detect research trends and map documents into clusters or networks, e.g. knowledge discovery (Karanikas and Theodoulidis 2001), topic modeling (Blei et al. 2003, Blei and Lafferty 2009) and network analysis of articles (Tonta and Darvish 2010, Duvvuru et al. 2012). These models reveal underlying document structure by analyzing word patterns and linking documents with similar patterns, namely topics (Blei et al. 2003). In addition, inherent metadata information from documents, such as author and time stamp, makes it possible to perform diachronic topic analyses. Topic modeling has been successfully applied to various text genres and document domains, e.g. news articles (Blei et al. 2003, Wei and Croft 2006, Newman et al. 2006), scientific abstract (Griffiths and Steyvers 2004), scientific papers (Blei and Lafferty 2007), digital libraries (Jockers 2013, Ngueyen 2014) and twitter (Ramage et al. 2010, Hong and Davidson 2010).

At present, several state-of-the-art tools, such as LDA and MALLET, allow for topic modeling. While these command-line tools offer new insights and perspectives for textual analysis, their use often remains limited to machine learning and natural language processing fields. First, there is no simple topic modeling stand-alone application to implement this analysis and researchers from other fields would inevitably have to acquire new programming skills, which is time-consuming and often challenging. Secondly, social science and humanities researchers seek more interactive control of modeling, which can serve as “holistic support for exploratory analysis” (Klein and Eisenstein 2013). In this paper, we propose to address these issues by integrating micro and macro-analyses with a dynamic interactive interface in which a researcher has control over text analysis and visualization. The development of our toolkit, Interactive Text Mining Suite, is guided by the following questions:

1. Which existing topic modeling and visualization tools provide an interactive control of modeling?
2. Can we build an application for a wide range of documents, such as scholarly articles and digital literary collections?
3. Can we incorporate various methods for topic modeling?
4. How can we optimize this application for a wide range of users?

The remainder of this paper is organized as follows. In Section 2, we describe topic modeling and review existing topic modeling tools. In Section 3, we introduce Interactive Text Mining Suite (ITMS). Section 4 details our case study. In Section 5, we summarize our contributions and present our future directions.

2. Topic Modeling and Topic Modeling Tools

Topic modeling refers to an algorithm that identifies short and informative descriptions of each document in a large collection that can further be used for various text-mining tasks, e.g. document classification, summarization, trend detection and corpus exploration (Blei et al. 2003, Blei 2012). The premise of this model is that text collections are “represented as random mixtures over latent topics, where each topic is characterized by a distribution over words” (Blei et al. 2003:996). Several algorithms have been put forth to build a probabilistic topic model, such as mixture-of-unigram (Nigam et al. 2000), Latent Semantic Indexing (Deerwester et al. 1990, Hofmann 1999) and Latent Dirichlet Allocation (Blei et al. 2003). The latter model, often referred to as LDA, has become a standard tool for topic identification.

The LDA model is currently available in several programming languages and environments. There exist two packages in R (R Development Core Team 2011), namely topicmodels (Grün and Hornik 2011) and lda (Chang 2010). LDA has also been implemented in Python (lda[1] and Tethne[2]) as well as in C (hca[3] and lda-c[4]), Java (MALLET[5] and lda-j[6]) and Scala[7] (McCallum et al. 2008). The second type consists of applications providing a graphical user interface (GUI) that facilitates users' interactions with an application. At present, there exist three stand-alone GUI applications for topic modeling: Stanford Topic Modeling Toolbox (TMT[8]), GUI Topic Modeling Tool (GTMT[9]) and TethneGUI[10]. The TMT toolkit was designed for social scientists; in it, input and output files are specifically formatted as a spreadsheet file, a common environment in the social sciences (Ramage et al. 2009). As a text-user interface tool, TMT involves some interaction by means of short scripts written in Scala[11]. This interaction, however, does not require extensive knowledge of Scala and is usually limited to small modifications of template scripts to reflect data, e.g. number of topics or data location (see Figure 1a). The second application, namely GUI Topic Modeling Tool, provides a graphical interface for topic model generation and navigation (see Figure 1b). Written in Java, this application is platform flexible. In addition, it uses a powerful MALLET toolkit for topic modeling as a back-end. The user is provided with several options—e.g. text normalization, stopwords, number of topics—and the output format is available as csv and html types. Finally, the third application, TethneGUI, serves as a graphical interface to the Tethne application. However, it was developed as a teaching tool within the field of history and philosophy of science and is not intended for extensive use.

One of the most common visualization methods in topic modeling involves a list of words associated with each topic and information on the correlation between topics and documents (see Figure 2).

Another approach was introduced by GTMT, where the output is displayed as an HTML output. Topics and documents are hyperlinked, allowing users to navigate between individual documents and related topics (see Figure 3). Finally, LDA

[1] https://pypi.python.org/pypi/lda

[2] http://diging.github.io/tethne/api/index.html

[3] Blei D., N. A. (2003). Latent Dirichlet allocation. Journal of Machine Learning Research, 3, 993–1022. http://www.mloss.org/software/view/527/

[4] http://www.cs.princeton.edu/~blei/lda-c/

[5] http://mallet.cs.umass.edu/about.php

[6] http://www.arbylon.net/projects/

[7] http://factorie.cs.umass.edu/

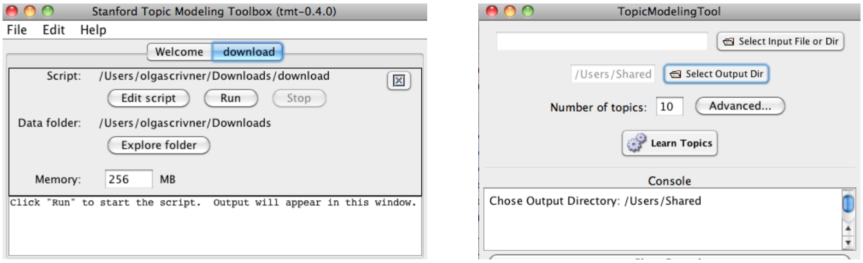
[8] http://nlp.stanford.edu/software/tmt/tmt-0.4/

[9] https://code.google.com/archive/p/topic-modeling-tool/

[10] https://pythonhosted.org/tethne/

[11] TMT was written in 2009 using an old version of Scala that is no longer developed.

topic models can also be represented as a topic-coupling network, where each topic is viewed as a cluster of terms and each term is linked to specific topics. For instance, the Tethne application offers results as a GraphML file that can be visualized in Cytoscape, an open source software platform for network visualization.



Script: /Users/olgascrivner/Downloads/download

Data folder: /Users/olgascrivner/Downloads

Memory: 256 MB

```
Click "Run" to start the script.  Output will appear in this window.
```

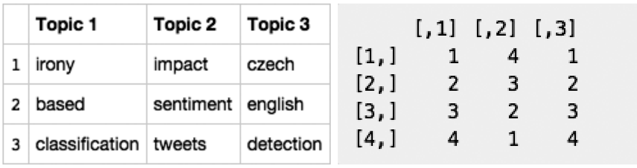
/Users/Shared

Number of topics: 10

```
Chose Output Directory: /Users/Shared
```

(a) TUI Interface for Standford TMT (b) GUI Topic Modeling Tool

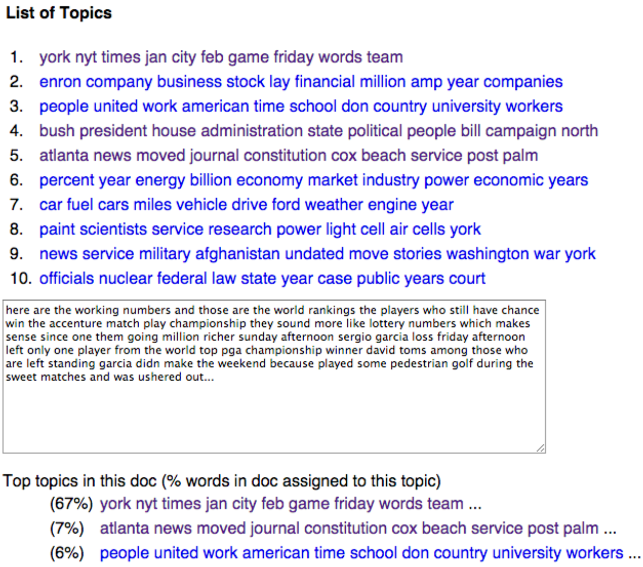
Figure 1. Graphical user interfaces for topic modeling applications



	Topic 1	Topic 2	Topic 3
1	irony	impact	czech
2	based	sentiment	english
3	classification	tweets	detection

```
     [,1] [,2] [,3]
[1,]    1    4    1
[2,]    2    3    2
[3,]    3    2    3
[4,]    4    1    4
```

Figure 2. Common topic visualization: list of words per topic (left) and topic-document matrix (right)



List of Topics

1. york nyt times jan city feb game friday words team
2. enron company business stock lay financial million amp year companies
3. people united work american time school don country university workers
4. bush president house administration state political people bill campaign north
5. atlanta news moved journal constitution cox beach service post palm
6. percent year energy billion economy market industry power economic years
7. car fuel cars miles vehicle drive ford weather engine year
8. paint scientists service research power light cell air cells york
9. news service military afghanistan undated move stories washington war york
10. officials nuclear federal law state year case public years court

here are the working numbers and those are the world rankings the players who still have chance win the accenture match play championship they sound more like lottery numbers which makes sense since one them going million richer sunday afternoon sergio garcia loss friday afternoon left only one player from the world top pga championship winner david toms among those who are left standing garcia didn make the weekend because played some pedestrian golf during the sweet matches and was ushered out...

Top topics in this doc (% words in doc assigned to this topic)

(67%) york nyt times jan city feb game friday words team ...
(7%) atlanta news moved journal constitution cox beach service post palm ...
(6%) people united work american time school don country university workers ...

Figure 3. GUI TMT HTML visualization

Most topic visualization techniques, however, remain static. As Blei (2012) points out, the developing of interactive user interfaces with topic visualization is a future direction in the topic modeling field. In addition, the social sciences and humanities express a need for the use of topic modeling as a means of exploratory analysis and require more interactive control of modeling (Klein and Eisenstein 2013). One of the recent attempts to construct such an interactive visualization is Paper Machines, a plug-in for the bibliographic management software Zotero[12]. Built with the Django web framework, Paper Machines allows for queries from Zotero library by time period, document title or location and displays the proportion of topics over time. The second project, TOME (successor of TopicViz (Einstein et al. 2012)), which is still under the development, aims to overcome the limitations of previous tools by combining topic and keyword searches (Klein and Eisenstein 2013). The graphical output represents a spatial layout of documents organized by topic, where documents and topics can be re-assembled interactively. In this approach, users can add or remove topics or documents, thus refining their research. Though these promising new tools offer novel insights by unveiling hidden patterns in large collections, they present some limitations. Paper Machines requires Zotero library and is mainly built for bibliographic management, whereas TOME and TopicViz are not released for public use. In addition, they would require an installation. Finally, users' metadata is not fully integrated into text analysis, "limiting the utility of the topics to facilitate additional research" (Klein and Einstein 2013). Thus, the question raised by Klein and Eisenstein (2013) remains: "How to integrate topic models and related automated text-mining tools with existing modes of scholarship"? We propose to reduce this gap by integrating topic modeling analysis into a broader interactive and exploratory web application.

3. Interactive Text-Mining Suite

The purpose of Interactive Text-Mining Suite (ITMS) is to provide a dynamic exploration of text collections. The ITMS application is built with R as a back-end and Shiny app as a front-end. In the back-end, Shiny app consists of two R scripts, namely server.R and ui.R. Server.R hosts all functions, whereas ui.R provides a graphical user interface. The use of the Shiny web framework for text mining tools has several advantages. First, as a web application, ITMS is platform independent and does not require installation, as compared to other topic modeling tools. Second, as an R application, ITMS has access to a range of state-of-the-art text analytical, statistical and graphical packages, e.g. lda, topicmodels, ggplot and tm. Finally, Shiny app is designed to build a dynamic and user-friendly interface (see Figure 4).

The ITMS interactive aspect allows users to have dynamic control of the application. For instance, users can upload their own metadata in csv format or extract existing metadata from files. Similarly, stop words can be added interactively or in a separate file. In addition, the user has a choice to specify a cutoff limit for frequency analysis. Furthermore, the ITMS application is not limited to a bibiliographic research, as it aims

[12] http://papermachines.org/

to assist researchers with a wide variety of data collections. The data format can vary from pdf to text files as well as a csv format for metadata. In contrast to other tools, ITMS also provides various topic modeling methods and interactive parameter tuning. Currently, we have included the following methods: 1) Collapsed Gibbs Sampling Method (package lda), 2) Latent Dirichlet Allocation (package topicmodels) and 3) Structural Topic Modeling (package stm). Parameter tuning includes number of topics, iteration, alpha and eta parameters. To help researchers determine the best number for topics, we have also incorporated a log likelihood function (topicmodels package) (Figure 5).

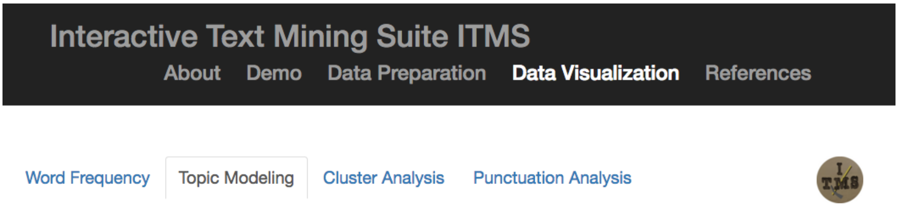
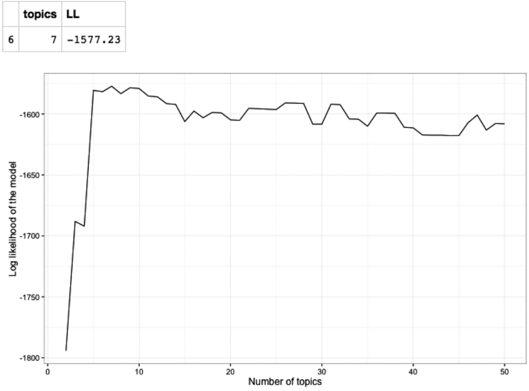


Figure 4. Interactive Text Mining Suite



	topics	LL
6	7	-1577.23

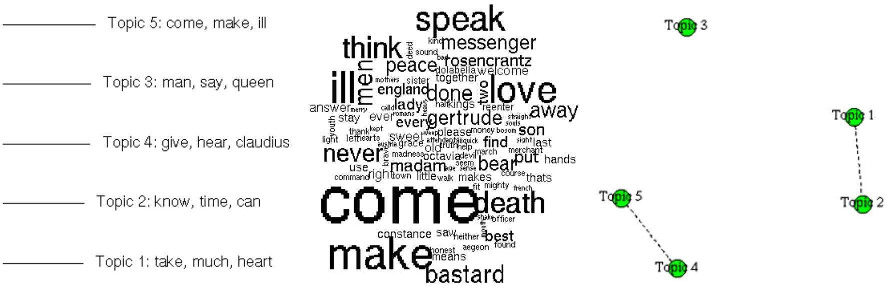
Figure 5. Log likelihood of the model

Various examples of topic visualization for Shakespeare's selected plays are shown in Figure 6. Topics can be shown as a list (Figure 6a), word cloud (Figure 6b) and topic clusters (Figure 6c). The extracted metadata (e.g. author and timestamp) is also available for topic analysis.

In addition to topic analysis, ITMS allows for a wide range of text mining tasks, such as frequency, cluster and punctuation analyses. Cluster analysis explores how individual texts are grouped, thus revealing their structural similarities in clusters (Jockers 2014). An example of clustering of Shakespeare's selected plays is illustrated in Figure 7. In contrast, punctuation analysis examines a superficial text structure of individual passage. Inspired by Adam Calhoun's blog[13], the punctuation analysis

[13] https://medium.com/@neuroecology/punctuation-in-novels-8f316d542ec4

is rendered as a heat map in where periods, question marks and exclamation marks are red, commas and quotation marks are green, and semicolons and colons are blue.



(a) Topic list (b) Topic cloud (c) Topic links

Figure 6. Topic visualization

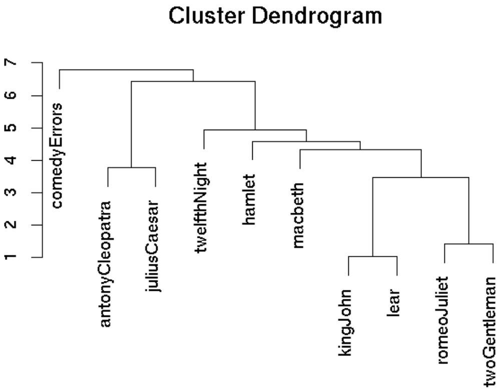


Figure 7. Cluster analysis of the Shakespeare’s selected plays

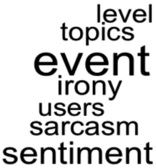
The ITMS application is currently hosted on the Shinyapps server[14] and is freely available to public. However, the performance is somewhat limited by a standard subscription to this service and for researchers with large collections we could provide our application for local installation.

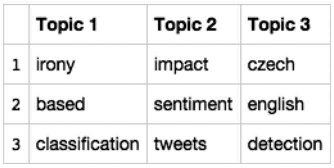
4. Case Study

We demonstrate the ITMS application by conducting a case study on recent papers analyzing Twitter data. This case study was guided by the following questions: 1) What were the main trends in research on Twitter during 2010–2015 and 2) Did these trends

[14] https://languagevariationsuite.shinyapps.io/TextMining/

change overtime? The data come from the ACL anthology website[15]. The ACL anthology hosts papers on the study of natural language processing from various ACL proceedings and workshops. The articles were searched with the keywords "twitter" and "tweet(s)", and they represent time periods between 2010 and 2015. We further followed the methodology from Griffiths and Steyvers (2004) and analyzed abstracts instead of full texts. For each article, the ITMS extracted its abstract and metadata. We removed stop words using an ITMS built-in default ("SMART" from tm package). In addition, we dynamically removed words, e.g. tweet(s), twitter, twitting, dataset. Word distribution and the list of topics are shown in Figure 8. The main themes are sentiment analysis (irony, sarcasm), topic modeling and twitter events. However, given the very small size of extracted articles (87), we are using this small corpus for illustration purposes only without making any claims about observed trends.





	Topic 1	Topic 2	Topic 3
1	irony	impact	czech
2	based	sentiment	english
3	classification	tweets	detection

(a) Word distribution (b) List of topics

Figure 8. Exploratory analysis

The relation between topics and documents can be examined in a multidimensional scaling graph (Figure 9). This graph provides a visual representation of the pattern of proximities.

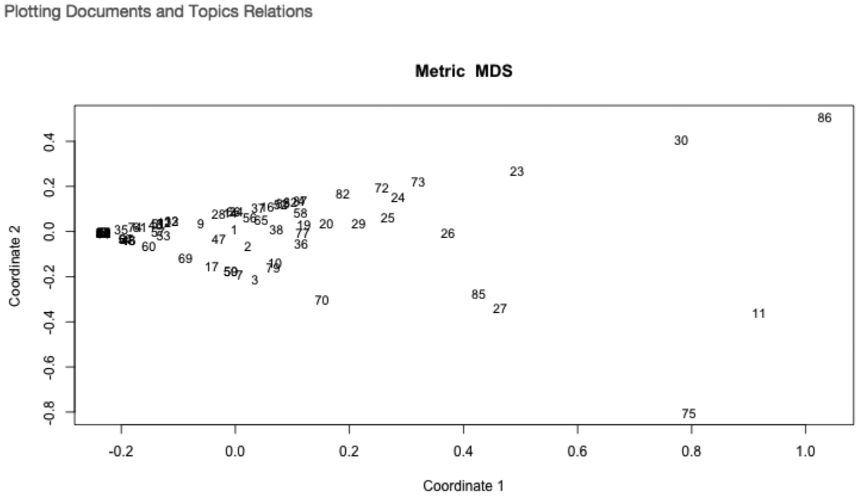


Figure 9. Multidimensional Scaling

[15] http://www.aclweb.org/anthology/

The numbers in Figure 9 correspond to document numbers. While most articles are clustered, several articles stand out, namely 86, 11, 30 and 75. These articles are titled as follows: “Task Alternation in Parallel Sentence Retrieval for Twitter Translation”, “A Unified Model for Topics, Events and Users on Twitter”, “Self-Disclosure Topic Model for Twitter Conversations”, “A Quantitative and Qualitative Study Based on Twitter”. Finally, the diachronic analysis in Figure 10 shows that topic 3 (sentiment analysis) prevails across time in this collection, as compared to the other topics.

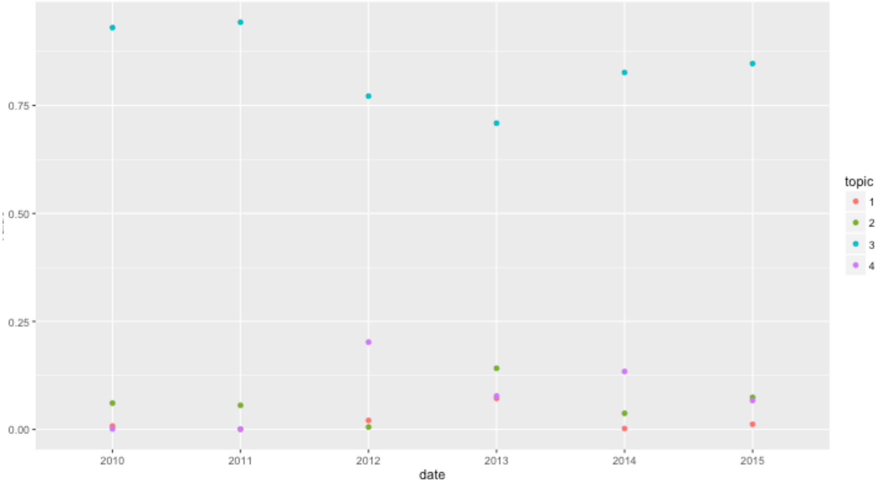


Figure 10. Diachronic trends

5. Conclusion and Future Work

In recent years, we have seen growing interest in the construction of text-mining and visualization tools for text collections. Many scholars, however, have expressed the need for a more integrated approach, namely the “synthesis of computational and humanistic modes of inquiry” (Klein and Einstein 2013). To incorporate this approach, the authors of this article have proposed to develop a user-friendly application for exploratory analysis providing a wide range of analytical and graphical tools. In addition, the accessibility of our web application facilitates data analysis, as researchers are not constrained by programming skills, memory limitation or platform dependency.

There are several developments that we see in the future for ITMS. First, from a visualization standpoint, the exploratory analysis will be enhanced with dynamic network graphs and dynamic diachronic mapping (e.g. igraph and GoogleViz packages). Second, supported files should include xml and html formats as well as urls. In addition, more types of exploratory analysis for social and humanities fields should be included, such as KWIC and correspondence analysis, among others. Furthermore, corpus annotation would be equally important, as linguistic analysis allows for a data structure and language use exploration. Finally, to increase the effectiveness and optimization of our application, we plan to host it on a Hadoop server.

References

1. *Blei D., Ng A., Jordan M.* (2003) Latent Dirichlet allocation. Journal of Machine Learning Research, Vol. 3, pp. 993–1022.
2. *Blei D.* (2012), Probabilistic Topic Models. Communcations of the ACM, Vol. 55, Issue 4, pp. 77–84.
3. *Buntine, W. L. and Mishra, S.,* (2014), Experiments with Non-parametric Topic Models. In Proceeding of 20th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.
4. *Chang J.* (2010). lda: Collapsed Gibbs Sampling Methods for Topic Models. R package version 1.2.3.
5. *Deerwester S., Dumais S. T., Furnas G. W., Landauer T. K., Harshman R.* (1990). Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science, Vol. 41, Issue 6, pp. 391–407.
6. *Duvvuru, A., Kamarthi, S., Sultornsanee, S.,* (2012), Undercovering research trends: Network analysis of keywords in scholarly articles, in Computer Science and Software Engineering (JCSSE), 2012 International Joint Conference, pp. 265–270.
7. *Eisenstein J., Chau D. H., Kittur A., Xing E. P.* (2011), TopicScape: Semantic Navigation of Document Collections, CoRR abs/1110.6200.
8. *Eisenstein J., Chau D. H., Kittur A., Xing E. P.* (2012), TopicViz: Interactive Topic Exploration in Document Collections, CHI 2012, Austin, Texas.
9. *Grün B, Hornik K* (2011). topicmodels: An R Package for Fitting Topic Models. Journal of Statistical Software, Vol. 40, Issue 13, pp. 1–30.
10. *Haralampos Karanikas and Babis Theodoulidis Manchester,* (2001), Knowledge Discovery in Text and Text Mining Software, Centre for Research in Information Management, UK
11. *Hofmann T.* (1999), Probabilistic latent semantic indexing. In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 50–57.
12. *Jockers M. L.* (2013), Macroanalysis: Digital Methods and Literary History, University of Illinois Press.
13. *Jockers M. L.* (2014), Text Analysis with R for Students of Literature. Quantitative Methods in the Humanities and Social Sciences. Springer International Publishing, Cham.
14. *Klein L., Eisenstein J.* (2013), Reading Thomas Jefferson with TopicViz: Towards a Thematic Method for Exploring Large Cultural Archives. Scholarly and Research Communication, Vol. 4, Issue 3 http://src-online.ca/index.php/src/article/view/121/259.

15. *McCallum, A,* (2002), MALLET: A Machine Learning for Language Toolkit. http://mallet.cs.umass.edu.
16. *McCallum A., Rohanemanesh K., Wick M., Schultz K., Singh S.* (2008), FACTORIE: Efficient Probabilistic Programming for Relational Factor Graphs via Imperative Declarations of Structure, Inference and Learning. NIPS Workshop on Probabilistic Programming, (NIPS WS).
17. *McCallum A., Schultz K., Singh S.* (2009), FACTORIE: Probabilistic Programming via Imperatively Defined Factor Graphs. Neural Information Processing Systems (NIPS), 2009.
18. *Nigam K., McCallum A. K., Thrun S., Mitchell T.* (2000), Text classification from labeled and unlabeled documents using em. Machine learning, Vol. 39, Issue 2–3, pp. 103–134.
19. *Ngueyen E.* (2014), Text Mining and Network Analysis of Digital Libraries in R. In Yanchang Zhao, Yonghua Cen (Eds), Data Mining Applications with R, pp. 95–116
20. *Ramage D., Rosen E., Chuang J., Manning C. D., McFarland A. D.* (2009). Topic Modeling for the Social Sciences. Workshop on Applications for Topic Models, NIPS, http://idl.cs.washington.edu/papers/topic-modeling-social-sciences.
21. *Ramage D., Dumais S., Liebling D.* (2010), Characterizing microblogs with topic models. In International AAAI Conference on Weblogs and Social Media, 2010.
22. *Tonta Y., Darvish H. R.* (2010), Diffusion of latent semantic analysis as a research tool: A social network analysis approach, Journal of Informetrics, Vol. 4, Issue 2, pp. 166–174.