`topicwizard` - a Modern, Model-agnostic Framework for Topic Model Visualization and Interpretation

Márton Kardos
Aarhus University
martonkardos@cas.au.dk

Kenneth C. Enevoldsen
Aarhus University
kenneth.enevoldsen@cas.au.dk

Kristoffer Laigaard Nielbo
Aarhus University
kln@cas.au.dk

Abstract

Topic models are statistical tools that allow their users to gain qualitative and quantitative insights into the contents of textual corpora without the need for close reading (Nielbo et al., 2024). They can be applied in a wide range of settings from discourse analysis (Bednarek, 2024), through pretraining data curation (Peng et al., 2025), to text filtering (Ma et al., 2016). Topic models are typically parameter-rich, complex models, and interpreting these parameters can be challenging for their users. It is typical practice for users to interpret topics based on the top 10 highest ranking terms on a given topic. This *list-of-words* approach, however, gives users a limited and biased picture of the content of topics (Gillings and Hardie, 2022). Thoughtful user interface design and visualizations can help users gain a more complete and accurate understanding of topic models' output. While some visualization utilities do exist for topic models, these are typically limited to a certain type of topic model. We introduce `topicwizard` [1], a framework for model-agnostic topic model interpretation, that provides intuitive and interactive tools that help users examine the complex semantic relations between documents, words and topics learned by topic models.

1 Introduction

Topic models are statistical instruments, which have been developed to uncover human-interpretable topics in corpora of text (Blei, 2012). These methods have allowed analysts gain insights into the contents of large corpora, the manual reading of which would be impractical or impossible. Topic models also often offer a more impartial account of a corpus' content (Nielbo et al., 2024).

Typically, topic models' outputs are presented to users in the form of the highest-ranking words and perhaps documents on a given topic. While this allows users to gain a superficial understanding of a topic, one might miss crucial details, and a lot of nuances, when topic models are exmined this way (Gillings and Hardie, 2022). We suggest that topic models capture more detailed information about topics than simple word lists convey, and that carefully designed interfaces can help users better explore this complexity.

1.1 Topic Models are Diverse

While topic models all carry out a similar task, they can also be very different from each other in how they conceptualize topic discovery.

Topic models originally relied on a bag-of-words model of documents where they are represented as sparse vectors of word-occurrence counts, with an optionally applied weighting scheme, such as tf-idf. Most commonly, these models either discover topics by matrix factorization (Gillis and Vavasis 2014, Kherwa, Pooja and Bansal, Poonam 2017) or by fitting a probabilistic generative model over these representations (Blei et al. 2003, Yin and Wang 2014, Hofmann 1999) or biterms (Yan et al., 2013).

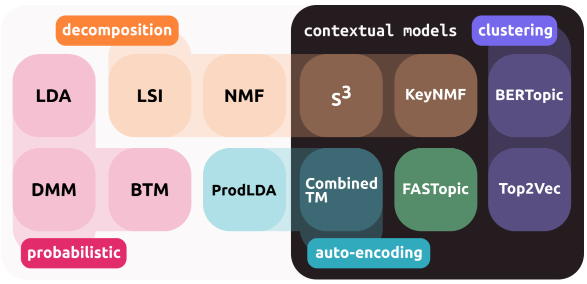

Figure 1: A Simplified Taxonomy of Topic Models

More recent topic models, however, also rely on context-sensitive, dense text representations from neural networks (Reimers and Gurevych, 2019). These models can conceptualize topic discovery as document clustering and post-hoc term importance

[1] https://github.com/x-tabdeveloping/topicwizard

estimation (Grootendorst 2022, Angelov 2020), document generation with amortized variational inference (autoencoders) (Bianchi et al. 2021a, Bianchi et al. 2021b), semantic relation reconstruction (Wu et al., 2024), or semantic decomposition (Kardos et al. 2025a, Kristensen-McLachlan et al. 2024).

1.2 Topic Models are Alike

Despite these differences, all topic models have a lot in common. Topic models, in essence, learn a three-way relationship between `words`, `documents` and `topics`.

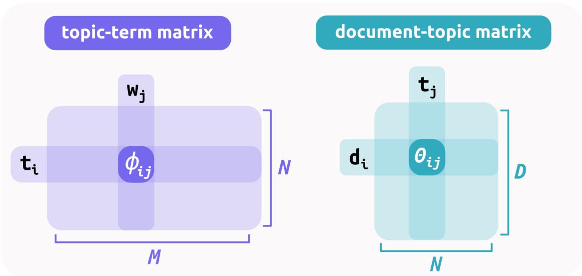

Figure 2: Common Components Computed by Topic Models

All topic models have a method for extracting the K most relevant words from the discovered topics. These top K words are calculated from a **topic-term matrix** (ϕ), which is either inferred as part of topic discovery. This matrix has N rows, corresponding to the number of topics, and M columns corresponding to the size of the model's vocabulary. In addition, models compute a **document-topic-matrix** (Θ), where rows represent the D documents in the corpus, while the N columns represent topics. This matrix contains the importance/relevance of a topic in a document.

1.3 Contribution

We introduce `topicwizard`, a model-agnostic topic model visualization framework that allows users to investigate complex semantic relations between words, documents and topics in their corpora. `topicwizard` is natively compatible with topic modelling libraries, which use the scikit-learn API (Pedregosa et al., 2011), such as tweetopic (Kardos, 2022) and Turftopic (Kardos et al., 2025b) and comes with compatibility layers for Gensim and BERTopic.

2 Related Work

Due to Latent Dirichlet Allocation's (LDA) popularity, a considerable amount of work has been dedicated to visualizing and interpreting its outputs. Chuang et al. (2012b) discuss best practices and design considerations for visualization and interpretation systems for LDA. Chuang et al. (2012a) introduced the Termite system for interactively visualizing and interpreting LDA output. The main visualization in Termite is a stylized version of the topic-term matrix (see Figure 8), where circles of different size are at the intersection of terms and topics indicating their importance. The authors also propose a scheme for selecting the most topically salient words, since displaying all words in the corpus would not be feasible. As a consequence, Termite can only display a limited number of words. Additionally, Termite is no longer under active maintenance [2].

LDAvis (Sievert and Shirley, 2014) is an interactive visualization R package for LDA (see Figure 9). LDAvis combines elements of previous topic visualization systems, including an inter-topic distance map, term distribution plots, and a term-weighting scheme to show only the most specific and (*relevant*) terms. Similar to Termite, the original LDAvis package is no longer maintained. Its Python port, PyLDAvis, receives occasional updates, but does not enjoy feature parity with the original package.

Notable visualization utilities are also included in the BERTopic library (Grootendorst, 2022), which boasts model-specific plotting functions, such as an inter-topic map, document cluster visualizations, and term distribution bar-charts. Similarly, Turftopic (Kardos et al., 2025b) also contains model-specific visualization utilites for a number of models, including cluster maps, concept compasses for S^3 (Kardos et al., 2025a) and interactive timeline plots for dynamic topic models. While these visualizations are useful, they are typically of limited interactivity, and are limited to a particular type of model.

3 `topicwizard`

To address these challenges, we outline `topicwizard`, a novel system for topic model interpretation. Our framework is model-agnostic,

[2] The Termite repository on Github was last committed to 11 years prior to the writing of this article

Figure 3: An overview of visualizations and pages in the `topicwizard` framework
All visualizations were produced using KeyNMF (Kristensen-McLachlan et al., 2024)

allows users to investigate topic models from a number of distinct perspectives, and is highly interactive, thereby providing a more complete picture of topic models' output,

3.1 Topic Models Learn Topic Representations

Topic models' primary objective is to discover latent themes in a corpus. Being able to understand what concepts make up such topics, and how these topics are related is perhaps the most important aspect of interpreting topic models.

In `topicwizard` (see Figure 3a), similar to Sievert and Shirley (2014) an `inter-topic map` is displayed, which shows the relative distances of topics to each other. While Sievert and Shirley (2014) utilize PCA for this visualization, projections in `topicwizard` are calculated with UMAP (McInnes et al., 2018), since it is better at capturing local structure. The size of the topics on the graph is determined by a *topic importance* score. This score, and thereby the size on the graph indicates how prevalent a given topic is in the corpus overall, also taking into account the length of the documents. Topic importance is calculated in the following manner:

$$s_t = \sum_{d}^{D} \Theta_{dt} \cdot |d|$$

where Θ_{dt} is the importance of topic t and document d and $|d|$ is the number of terms in a given document, and D is the size of the corpus.

To provide users with insights about topics' word content, the `topic-word plot` displays the distribution of the highest ranking words for a given topic, and also how globally prevelant these words are across topics [3]. Since 10-20 words are rarely enough to give a complete picture of the words relevant to a topic, a more comprehensive `topic wordcloud` is also displayed To aid further analysis, users can also manually name topics on this page.

3.2 Topic Models Learn Word Embeddings

While topic models' are mainly oriented at discovering topics, they also implicitly learn meaningful representation of words within the corpus. Each column of the topic-term matrix can technically be thought of as a semantic embedding for a given word, with the dimensions being interpretable. This implicit learning of word representations allows us to examine words' relation to each other in a corpus, without explicit reference to the topics.

In `topicwizard` (see Figure 3c), a `word map` is displayed to users, allowing them to quickly and interactively investigate the semantic landscape of words in their corpus. Word positions are calculated by projecting word embeddings to two dimensions using UMAP.

Word embeddings are useful for investigating associative relations in corpora, and have been used for a variety purposes such as query expansion

[3]Unlike LDAvis, we do not compute *relevance* scores, since they rely on the assumption that ϕ contains word probabilities.

(Kuzi et al., 2016), or to uncover authorship patterns in literature (Baunvig, 2024). Clicking on a word on the word map highlights the words most closely related to the selected one and displays the topical distribution of the selected term and its neighbourhood on the `word-topic plot`. Displaying closely associated words with the selected keywords in topic models can give practitioners a more nuanced picture of word use (Liu and Lei, 2018).

3.3 Topic Models Organize Documents

An important aspect of topic models is that they learn a representation of documents in the corpus they are fitted on. Document representations discovered by topic models were historically used for a number of purposes, including retrieval (Yi and Allan, 2009), and studying information dynamics (Barron et al., 2018).

In `topicwizard` (see Figure 3d), a `document map` is displayed, where document's UMAP-projected embeddings can be seen, and documents are coloured based on most prevalent topic. In the case of BoW models, these representations are derived from the document-topic matrix, while with contextual models, the pre-computed sentence embeddings are used.

Secondly, individual documents' contents can be investigated on a `document-topic plot`, which displays the distribution of the most relevant topics, a `document-topic timeline`, which displays how the topical content changes throughout the course of the document and a `document viewer`, where a snippet of the document is displayed, and the most topically relevant words are highighted. The combination of these document inspection utilities can help users ground and verify topic models' output in the documents themselves, which elevates trust (Chuang et al., 2012b). Additionally, this interface encourages close reading, which provides additional insight into the corpus' content.

3.4 Topics Augment User-Defined Groups

Commonly, users of topic models also have some externally defined grouping of documents, which might be relevant for their analyses. This could be binning documents by time period, predefined categories or place of origin. While most topic models do not utilize external labels, meaningful inferences can be made about topics' relation to these labels post-hoc.

An important part of this process is to compute a **group-topic matrix**, the cells of which contain the summed importance of a given topic for documents in a given group:

$$G_{ij} = \sum_{k}^{D} \Theta_{kj} \cdot I(g_k = i)$$

where G_{ij} is the importance of group i for topic j, g_k is the group label of document k, and $I(g_k = i)$ is the indicator function.

In `topicwizard` (see Figure 3b), semantic distances between user-defined groups can be seen on the `group map`, where group-topic representations are projected to 2D space using UMAP. Groups are coloured based on the dominant topic in the group. Topic distributions in groups can be seen on the `group-topic plot`, and groups' lexical content can be examined in detail on the `group wordcloud` to the right.

3.5 Software Design Considerations

The `topicwizard` Python package was designed with both research and enterprise use in mind. As such, our goal was to develop a package that is accessible to new users and sufficiently flexible to accommodate specific use cases – ranging from academic writing and technical reporting to enabeling business analysts to interact with topic models via a web interface.

The **Web Application** (see Figures 4 - 7) was designed to make topic model interpretation as seemless and quick as possible, in as many environments as possible, including Jupyter notebooks, in the browser, or deployed to the cloud. which produces a readily deployable Docker project to a specified folder.

The **Figures API** makes it trivial for our users to produce specific figures tailored to their needs. This is especially crucial for producing publications, since some colour schemes, fonts or aspect ratios, while appropriate for an interactive web application, might not be visually appealing in a static document.

4 Conclusion

This paper introduces `topicwizard`, a comprehensive, interactive, and model-agnostic topic model visualization framework. Our framework is a notable extension over previous topic model visualization systems, thanks to a) supporting a much wider range of models b) allowing users to ground topic models in the corpus, and investigate them from numerous angles and c) being flexible, actively supported, and production-ready. The `topicwizard` software package has so far been downloaded more than 45000 times from PyPI, demonstrating that practitioners have already found it useful.

Limitations

While `topicwizard` is the most comprehensive topic model visualization tool to date, it still lacks coverage of a number of aspects of topic modelling. It, for instance, does not have visualization utilities for dynamic, hierarchical and supervised topic models. This is a clear limitation and will have to be addressed in future package releases.

Our framework, as of now, does not provide any utilities for comparing outputs from different topic models either. This is yet another aspect that future work should address.

Furthermore, while we consider model-angosticity to be one of the strengths of our approach, it does, to an extent, limit its usefulness for certain models. Certain visualizations, such as concept compasses, might be highly useful tools for examining the output of Semantic Signal Separation, but their utility might be limited for clustering topic models. We encourage our users, therefore, to use `topicwizard` in tandem with model-specific interpretation utilities from libraries such as BERTopic or Turftopic.

References

Dimo Angelov. 2020. Top2vec: Distributed representations of topics. *Preprint*, arXiv:2008.09470.

Alexander T. J. Barron, Jenny Huang, Rebecca L. Spang, and Simon DeDeo. 2018. Individuals, institutions, and innovation in the debates of the french revolution. *Proceedings of the National Academy of Sciences*, 115(18):4607–4612.

Katrine Frøkjær Baunvig. 2024. *Grundtvig og Monstrene*. Center for Grundtvigforskning, Aarhus Universitet.

Monika Bednarek. 2024. Topic modelling in corpus-based discourse analysis: Uses and critiques. *Discourse Studies*.

Federico Bianchi, Silvia Terragni, and Dirk Hovy. 2021a. Pre-training is a Hot Topic: Contextualized Document Embeddings Improve Topic Coherence. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 759–766, Online. Association for Computational Linguistics.

Federico Bianchi, Silvia Terragni, Dirk Hovy, Debora Nozza, and Elisabetta Fersini. 2021b. Cross-lingual Contextualized Topic Models with Zero-shot Learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1676–1683, Online. Association for Computational Linguistics.

David M. Blei. 2012. Probabilistic topic models. *Commun. ACM*, 55(4):77–84.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.

Jason Chuang, Christopher D. Manning, and Jeffrey Heer. 2012a. Termite: visualization techniques for assessing textual topic models. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, AVI '12, page 74–77, New York, NY, USA. Association for Computing Machinery.

Jason Chuang, Daniel Ramage, Christopher Manning, and Jeffrey Heer. 2012b. Interpretation and trust: designing model-driven visualizations for text analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, page 443–452, New York, NY, USA. Association for Computing Machinery.

Mathew Gillings and Andrew Hardie. 2022. The interpretation of topic models for scholarly analysis: An evaluation and critique of current practice. *Digital Scholarship in the Humanities*, 38(2):530–543.

Nicolas Gillis and Stephen A. Vavasis. 2014. Fast and Robust Recursive Algorithmsfor Separable Nonnegative Matrix Factorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(4):698–714.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *Preprint*, arXiv:2203.05794.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, page 50–57, New York, NY, USA. Association for Computing Machinery.

Márton Kardos, Jan Kostkan, Kenneth Enevoldsen, Arnault-Quentin Vermillet, Kristoffer Nielbo, and Roberta Rocca. 2025a. S^3 - Semantic Signal Separation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 633–666, Vienna, Austria. Association for Computational Linguistics.

Márton Kardos. 2022. tweetopic: Blazing fast topic modelling for short texts. *GitHub repository*.

Márton Kardos, Kenneth C. Enevoldsen, Jan Kostkan, Ross Deans Kristensen-McLachlan, and Roberta

Rocca. 2025b. Turftopic: Topic modelling with contextual representations from sentence transformers. *Journal of Open Source Software*, 10(111):8183.

Kherwa, Pooja and Bansal, Poonam. 2017. Latent semantic analysis: An approach to understand semantic of text. In *2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC)*, pages 870–874.

Ross Deans Kristensen-McLachlan, Rebecca Marie Matouschek Hicke, Márton Kardos, and Mette Thunø. 2024. Context is Key(NMF): Modelling Topical Information Dynamics in Chinese Diaspora Media. In *Proceedings of the Computational Humanities Research Conference 2024*, volume 3834 of *CEUR Workshop Proceedings*, pages 829–847, Germany. CEUR-WS.

Saar Kuzi, Anna Shtok, and Oren Kurland. 2016. Query expansion using word embeddings. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, page 1929–1932, New York, NY, USA. Association for Computing Machinery.

Dilin Liu and Lei Lei. 2018. The appeal to political sentiment: An analysis of donald trump's and hillary clinton's speech themes and discourse strategies in the 2016 us presidential election. *Discourse, Context & Media*, 25:143–152.

Jialin Ma, Yongjun Zhang, Jinling Liu, Kun Yu, and XuAn Wang. 2016. Intelligent sms spam filtering using topic model. In *2016 International Conference on Intelligent Networking and Collaborative Systems (INCoS)*, pages 380–383.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. 2018. Umap: Uniform manifold approximation and projection. *J. Open Source Softw.*, 3:861.

Kristoffer L. Nielbo, Folgert Karsdorp, Melvin Wevers, Alie Lassche, Rebekah B. Baglini, Mike Kestemont, and Nina Tahmasebi. 2024. Quantitative text analysis. *Nature Reviews Methods Primers*, 4(1).

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jiahui Peng, Xinlin Zhuang, Qiu Jiantao, Ren Ma, Jing Yu, Tianyi Bai, and Conghui He. 2025. Unsupervised Topic Models are Data Mixers for Pre-training Language Models. *Preprint*, arXiv:2502.16802.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Carson Sievert and Kenneth Shirley. 2014. LDAvis: A method for visualizing and interpreting topics. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, pages 63–70, Baltimore, Maryland, USA. Association for Computational Linguistics.

Xiaobao Wu, Thong Thanh Nguyen, Delvin Ce Zhang, William Yang Wang, and Anh Tuan Luu. 2024. FASTopic: Pretrained Transformer is a Fast, Adaptive, Stable, and Transferable Topic Model. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Xiaohui Yan, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. 2013. A biterm topic model for short texts. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, page 1445–1456, New York, NY, USA. Association for Computing Machinery.

Xing Yi and James Allan. 2009. A comparative study of utilizing topic models for information retrieval. In *Advances in Information Retrieval*, pages 29–41, Berlin, Heidelberg. Springer Berlin Heidelberg.

Jianhua Yin and Jianyong Wang. 2014. A dirichlet multinomial mixture model-based approach for short text clustering. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, page 233–242, New York, NY, USA. Association for Computing Machinery.

A Appendix

See Figures 4-7 for screenshots of `topicwizard`, Figure 9 for LDAvis and Figure 8 for Termite.

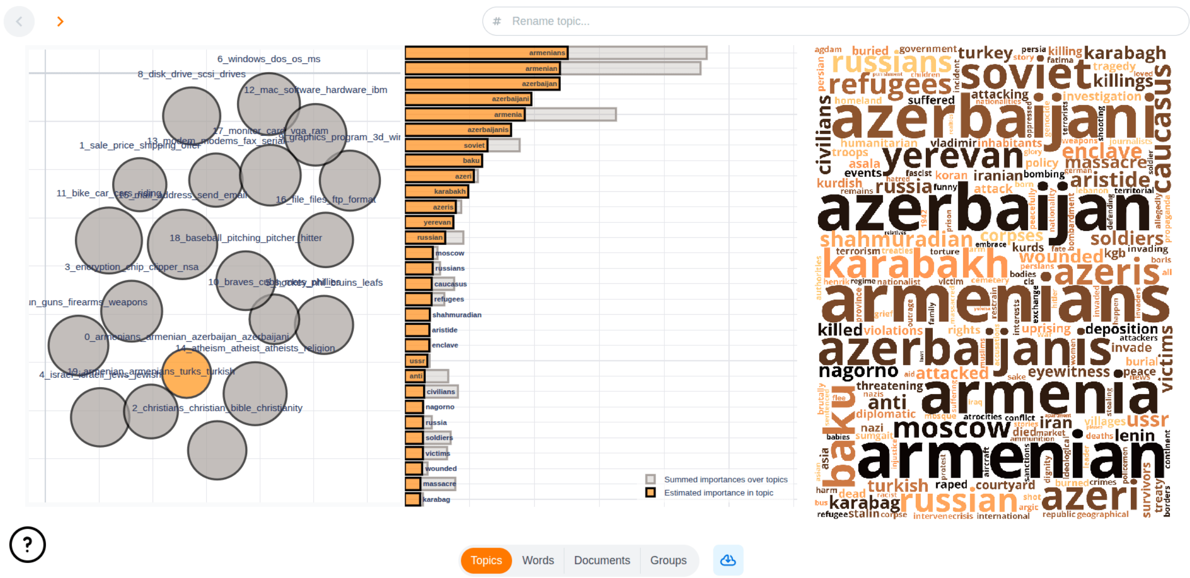

Figure 4: Screenshot of the `Topics` page in the `topicwizard` Web Application

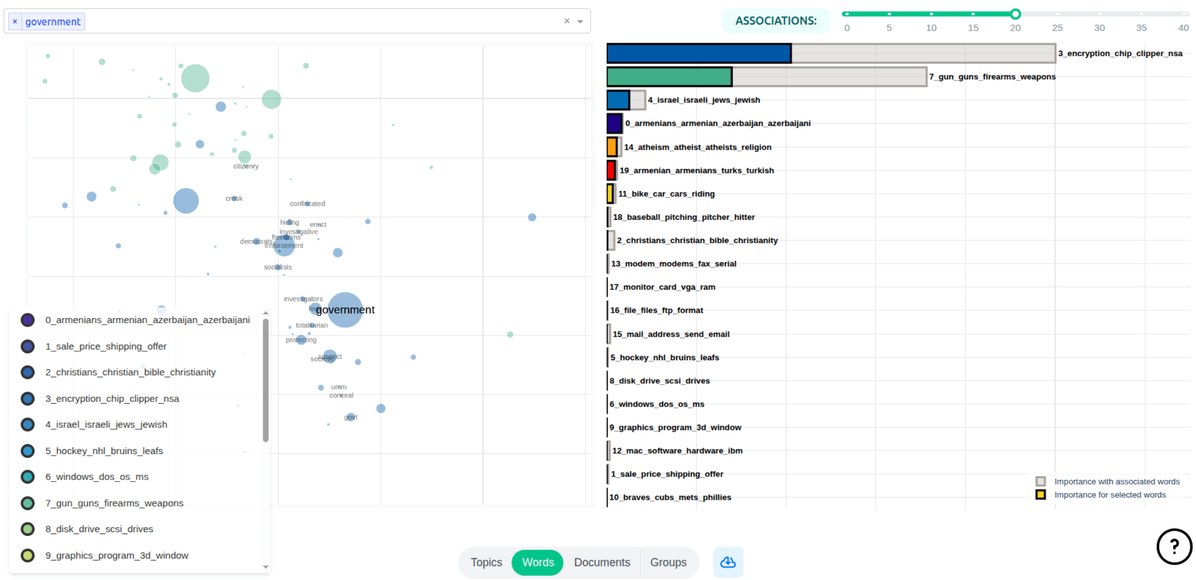

Figure 5: Screenshot of the `Words` page in the `topicwizard` Web Application

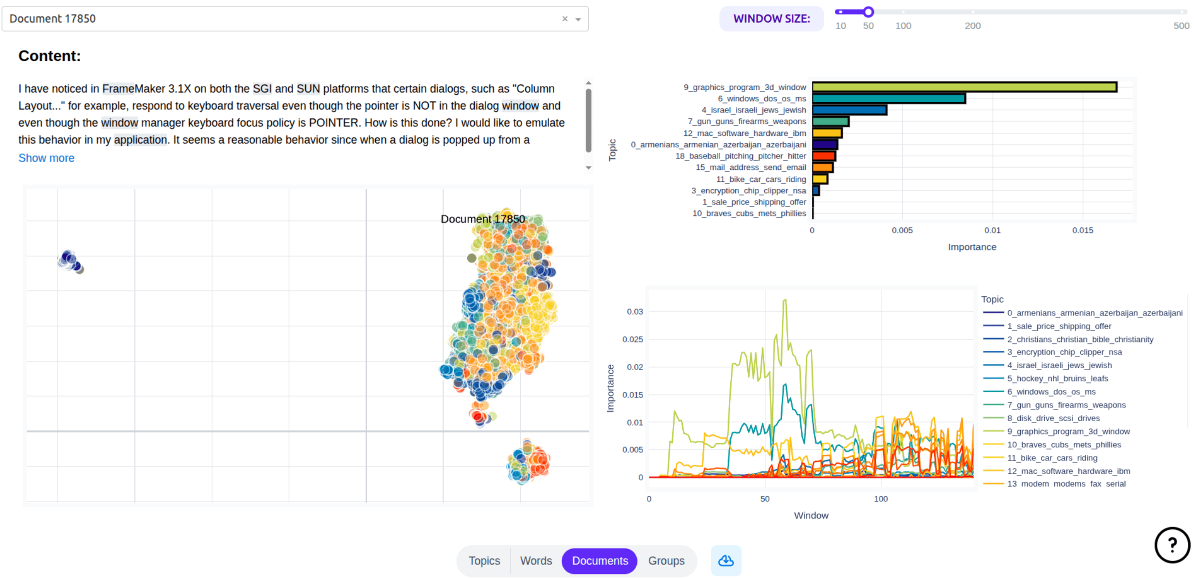

Figure 6: Screenshot of the `Documents` page in the `topicwizard` Web Application

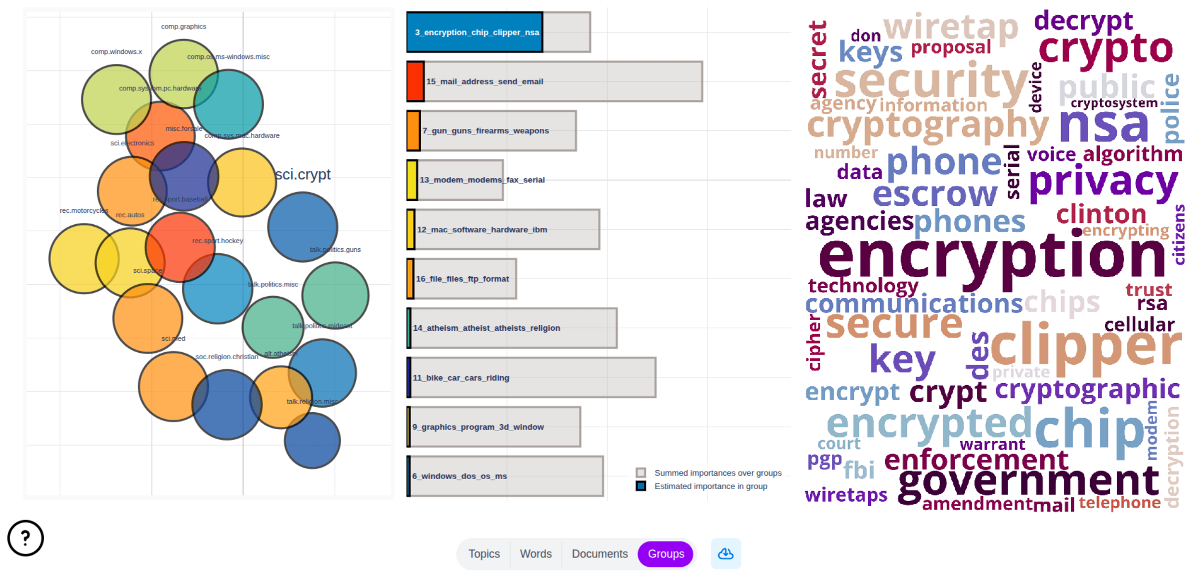

Figure 7: Screenshot of the `Groups` page in the `topicwizard` Web Application

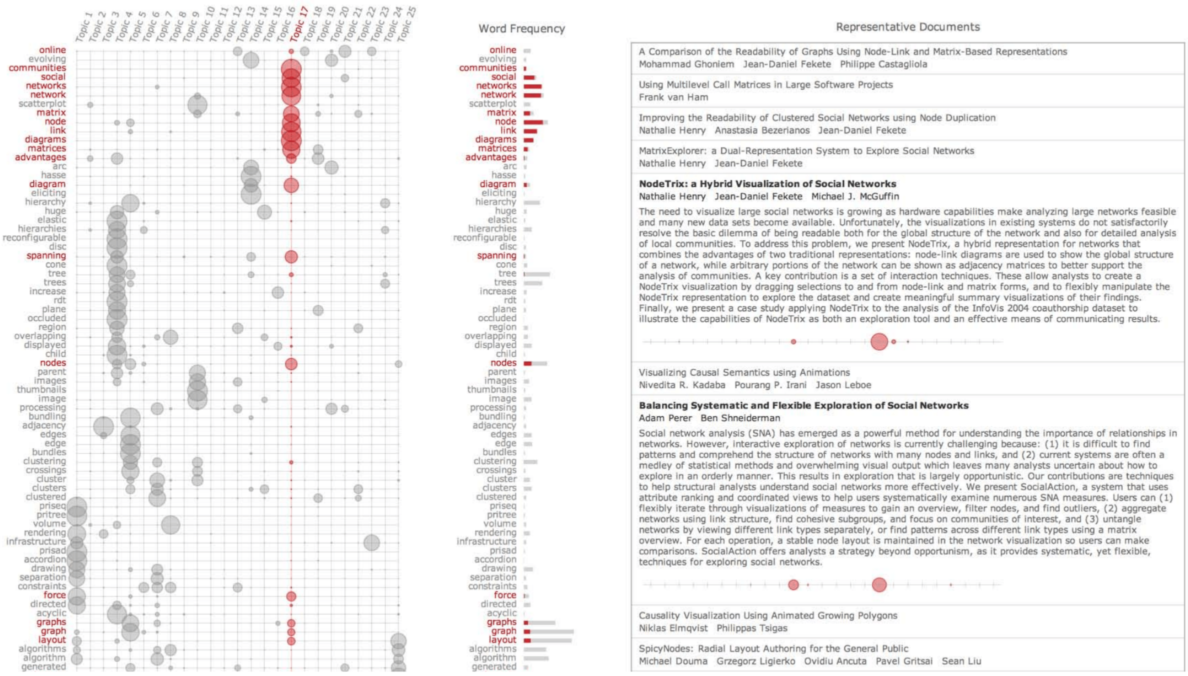

Figure 8: Screenshot of the Termite System
Figure from (Chuang et al., 2012a)

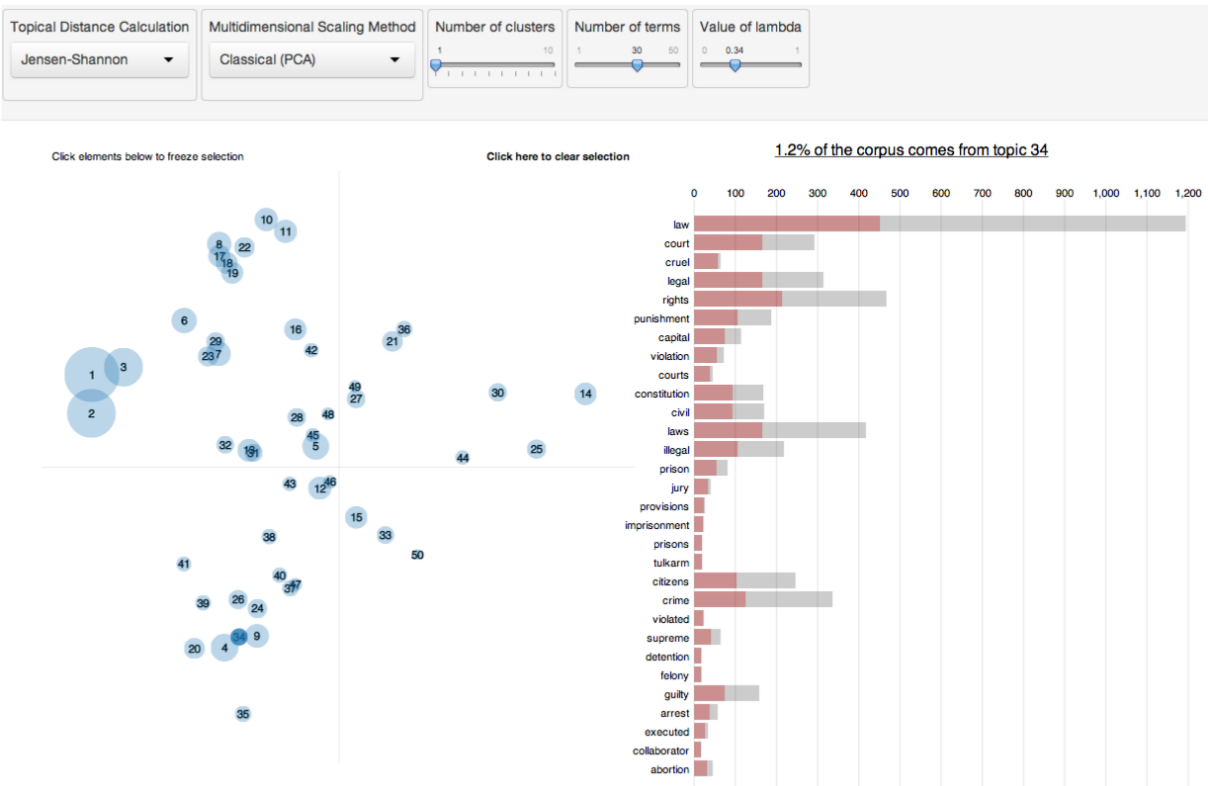

Figure 9: Screenshot of LDAvis
Figure from (Sievert and Shirley, 2014)