Visualising Scientific Topic Evolution

Panagiotis Deligiannis
IMSI, ATHENA RC
deligianp@athenarc.gr

Thanasis Vergoulis
IMSI, ATHENA RC
vergoulis@athenarc.gr

Serafeim Chatzopoulos
Univ. of the Peloponnese
schatzop@uop.gr

Christos Tryfonopoulos
Univ. of the Peloponnese
trifon@uop.gr

ABSTRACT

The automatic extraction of topics is a standard technique for summarizing text corpora from various domains (e.g., news articles, transport or logistic reports, scientific publications) that has several applications. Since, in many cases, topics are subject to continuous change there is the need to monitor the evolution of a set of topics of interest, as the corresponding corpora are updated. The evolution of scientific topics, in particular, is of great interest for researchers, policy makers, fund managers, and other professionals/engineers in the research and academic community. In this work, we demonstrate a prototype that provides intuitive visualisations for the evolution of scientific topics providing insights about topic transformation, merging, and splitting during the recent years. Although the prototype works on top of a scientific text corpus, its implementation is generic and can be easily applied on texts from other domains, as well.

CCS CONCEPTS

• **Information systems → Document topic models**;

KEYWORDS

Topic modeling, topic evolution, visualisation

ACM Reference Format:
Panagiotis Deligiannis, Thanasis Vergoulis, Serafeim Chatzopoulos, and Christos Tryfonopoulos. 2021. Visualising Scientific Topic Evolution. In *Companion Proceedings of the Web Conference 2021 (WWW '21 Companion), April 19–23, 2021, Ljubljana, Slovenia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3442442.3451371

1 INTRODUCTION

As science evolves through the publication of new research results, scientific topics are subject to continuous change. Monitoring the evolution of such topics in the recent years (e.g., investigating which of them remained almost intact, which evolved into new ones, etc.) is of great interest for professionals in the wide research and academic community. For instance, researchers may have the opportunity to better configure their plans for future research, while research fund managers may succeed in identifying interesting research areas and develop new calls for projects in these areas.

In recent years, various tools to explore scientific literature have been introduced (e.g., Google Scholar, AMiner [15], Semantic Scholar[1], BIP! Finder [16]). Their motivation was to facilitate the work of researchers and other professionals in identifying valuable research, a very tedious task due the exponential increase in the number of scientific publications during the recent years [3]. The focus of such systems is to provide keyword-based search functionalities to their users and they do not usually provide advanced topic monitoring options. Even those systems that provide such features usually simply present the evolution of the number of publications for a set of topics or keywords (e.g., AMiner's Trend[2] or Dimensions[3]). Additionally, other tools, like SciTo [4], also provide topic trends based on the aggregated impact of the publications of each topic. Nevertheless, none of the aforementioned tools provides the option to monitor how each scientific topic has evolved over time. This is an important issue since topics should not be considered as static entities as time goes by; new topics emerge due to new needs or scientific advancements, older topics are significantly being transformed, while some of them even cease to exist.

In this work, we demonstrate a prototype[4] that provides intuitive visualisations regarding scientific topic evolution. It is based on a large, multidisciplinary collection of scientific publication abstracts, gathered from Crossref [8]. The tool first calculates and then consults two topic models, one for the publications of the current period (last 5 years) and another one for publications of a past period (5 years before the current period). Its objective is to model and visualise the recent evolution of the main scientific topics based on the literature included in the selected corpus. The code of the topic evolution visualisations is provided as an open source software while a dataset, that contains the topics produced for both time periods, has been made openly available (see Section 3).

2 BACKGROUND

2.1 Preliminaries and Notation

Topic modeling refers to the field that studies text-mining approaches for the discovery of hidden semantic structures in text bodies. In this context, a *document* is a text body, i.e., a set of *words*. The term

WWW '21 Companion, April 19–23, 2021, Ljubljana, Slovenia

ACM ISBN 978-1-4503-8313-4/21/04.
https://doi.org/10.1145/3442442.3451371

[1] https://www.semanticscholar.org
[2] https://trend.aminer.org/
[3] https://app.dimensions.ai
[4] http://83.212.72.177/

corpus refers to a collection of documents, while *vocabulary* (or *dictionary*) to the set of all the distinct words in this collection. Given a corpus of n documents, that is defined based on a vocabulary of m words, we can use an $m \times n$ matrix that contains word frequencies per document (rows=unique words, columns=documents) to represent the corpus. This is known as the *term-document matrix* of the corpus and can be denoted as $C = (c_{i,j})$, where $c_{i,j}$ contains the frequency (e.g., number of appearances, tf-idf score) of the ith word in the vocabulary in the jth document of the corpus with $1 \leq i \leq m$ and $1 \leq j \leq n$.

2.2 Topic modeling approaches

Several approaches have been proposed for the extraction of topics from text corpora. Most of them utilise the statistical attributes of texts and focus on recording term co-occurrence to define models which can learn to identify the discussed topics of a given corpus.

Latent Semantic Analysis [13] (LSA)[5], is a seminal topic modeling approach which is based on applying a linear algebra-based dimensionality reduction technique called *truncated singular value decomposition* (truncated SVD, for brevity) on the term-document matrix of the corpus of interest. The desired number of topics is provided as parameter to the truncated SVD process and the output is a matrix having one line for each identified topic. Each line contains one numerical value for each document; a large value indicates a strong connection of the corresponding document with the topic. The analysis also produces an encoding matrix that captures how strongly each word in the dictionary is 'contributes' to each topic. It should be noted that there are various adaptations of LSA, like the *Probabilistic LSA* [9] (PLSA) approach that replaces the truncated SVD step with a mixture decomposition derived from a latent class model (in particular, the *aspect model*).

The most popular topic modeling approach is *Latent Dirichlet Allocation* [2] (LDA). This is a generative statistical model that allows a set of words to be interpreted by latent topics which can explain the underlying word similarities and connections. Each document is represented as a mixture of topics (the number of topics is a parameter). Each topic is considered to be a mutlinomial distribution over the words in a given vocabulary. LDA is a step forward from PLSA as it incorporates document-topic probabilities in its generative process, by sampling the document-topic multinomial parameters, as well as the topic-term multinomial parameters from Dirichlet distributions.

Since LDA does not explicitly model correlations among topics, this motivated a relevant line of work. Indicatively, *Pachinko Allocation Model* [10] (PAM) captures arbitrary, nested (and probably sparse) correlations between topics using a *directed acyclic graph* (DAG). The leaves of this graph correspond to individual words in the vocabulary, while interior nodes represent topics (modeled as Dirichlet distributions) and, subsequently, correlations between words (leaves) or other topics (other interior nodes).

Finally, there are various works attempting to model the evolution of topics: in [12] the authors exploit paper citations to reveal the hidden structure of topic evolution in a corpus; in [6] a model that combines dynamic LDA and word embeddings is introduced; in [1] a family of probabilistic time series models to analyze topic evolution in large corpora is proposed; [17] gives a nice survey of evolution approaches based on probabilistic topic modeling.

2.3 Topic evolution visualisation

Most existing tools that provide topic evolution visualisations simply present topic evolution according to the number of publications to which each topic relates. Indicatively, AMiner's Trend provides streamgraph-like visualisations that illustrate how many articles, which are related to the sub-topics of a user-selected scientific area, have been published each year. SciTo [4] follows a similar approach, however it is not restricted in providing only the number of publications for each topic; it also provides various aggregated statistics about the impact of each topic.

However, these approaches do not provide insights about how each scientific topic has evolved over time (a topic's terms composition are subject to continuous change). Also, as time goes by, individual topics may be merged into a single larger topic or individual topics may be split into more topics. Finally, topics may cease to exist at some time. To the best of our knowledge there are no systems that provide visualisations for this type of information for scientific topics.

It is worth mentioning that there are various topic evolution tools for other types of text corpora, however most of them share similar characteristics with the aforementioned scientific topic evolution visualisation tools (e.g., [5, 14]). A notable exception is TopicFlow [11], which shares a very similar approach to the one presented in this work; however this tool is focused on small texts from microbloging platforms and it utilises cosine similarity to identify topic similarities. Finally, there are various mature topic visualisation libraries like pyLDAVis[6] or LDAExplore [7], however such libraries do not provide topic evolution functionalities.

3 ARCHITECTURE

The architecture of our prototype for the visualisation of scientific topic evolution is illustrated in Figure 1. The tool consists of four major components: the *Data Collector & Cleaner*, which is responsible of gathering and cleaning article abstracts and metadata from Crossref, the *Topic Trainer*, which undertakes the task of training the required topic models, the *Evolution Modeler*, which generates the evolution data for each topic, and the *Front-end UI*, which implements the user interface. The code of the latter is available based on a GNU/GPL license[7]. In the next sections we elaborate on the technical details behind of the aforementioned components.

3.1 Data Collector & Cleaner

This component undertakes all tasks which are relevant to collecting and cleaning data from external sources and preparing the input required by other components. Our tool gathers publication data from Crossref [8], in particular from the public data files which are made available occasionally[8]. From these files, we keep only the publication year and the abstract of each publication, only for those records that both these metadata are not null (the rest are discarded).

[5] It is also known as *Latent Semantic Indexing* (LSI).

[6] pyLDAVis: https://pyldavis.readthedocs.io/en/latest/readme.html

[7] Front-end: https://github.com/deligianp/sci-k-topic-evolution-frontend, back-end: https://github.com/deligianp/sci-k-topic-evolution.

[8] For example, here: https://doi.org/10.13003/83B2GP.

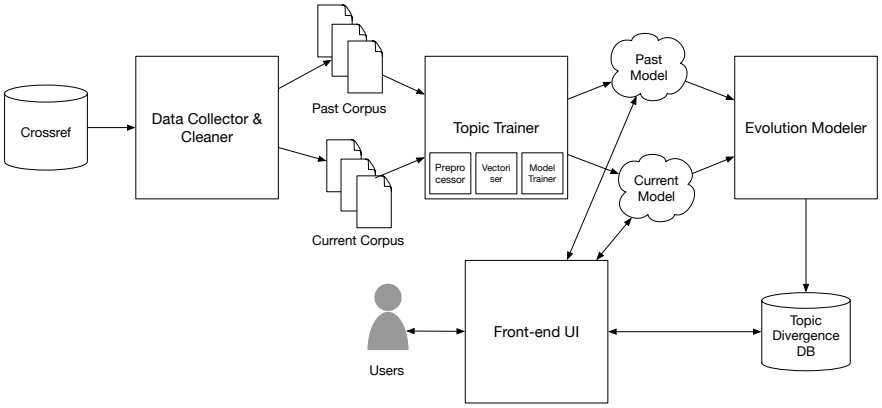

Figure 1: The architecture of the tool.

In addition, to avoid the use of a noisy dataset, which may affect the performance of the produced models, an initial filtering procedure is performed: utilising the *langdetect*[9] Python library, any articles with abstracts written in other languages than English are removed from the dataset. Finally, we keep only publications into the time period between 2011 and 2020, and we split these records into two corpora: one containing all articles published between $2011 - 2015$ and another one containing all articles between $2016 - 2020$. This is done to support the topic evolution analysis: it is required for this analysis to train two distinct topic models, one using the articles of a past period and another one using the articles of the current period. In our case, each period is 5 years long (an arbitrary, however fair decision, since the period is not too large to miss significant intermediate topic modifications).

3.2 Topic Trainer

Our prototype relies on training two LDA [2] topic models, one for each of the two article corpora generated by the Topic Collector & Cleaner (Section 3.1). The exact same training workflow is followed for both corpora. This workflow is essentially one of the components of our tool's architecture and we refer to it as the *Topic Trainer* component. Its output is given to the Evolution Modeler component and, at the same time, is saved in JSON files to be published as an open dataset on Zenodo[10]. In the next paragraphs we describe the technical details of its individual sub-modules.

3.2.1 Preprocessor. This module carries out the required operations that transform each text to a more machine-readable format. Initially, any potential hypertext artifacts are removed. Then, the text is passed through a *Part-of-Speech* (PoS) tagger[11] to distinguish nouns, verbs and adjectives and pass them through a lemmatizer, while the rest of the words remain intact. The resulting set of words is then decapitalized and words with less than three characters or words detected as stopwords, are dropped. Furthermore, any punctuation, with the exception of hyphens(-), is removed. Finally, from the transformed texts we keep only those containing more than 10 words, since smaller texts fail to capture enough content represent any useful topic.

3.2.2 Vectoriser. This module exploits the transformed corpus to determine the topic model's vocabulary. Initially, a term-document matrix is constructed, which is used to generate a vocabulary of at maximum $100,000$ terms taking into account each term's frequency and salience. In particular, the module accepts terms that appear, in at least 5 different documents, but not in more than half of the documents of the corpus. The inferred vocabulary is finally used to convert each transformed text into a Bag-of-Words vector.

3.2.3 Model Trainer. This module implements an LDA [2] topic modeling training process. It gets as input a set of vectorised texts produced by the Vectoriser. Through an iterative process, the model attempts to find the probability distributions for each topic, that maximise the likelihood of observing the training corpus. The number of topics as well as the number of iterations, are part the model's hyperparameters that are set by the modeler. Implementations of LDA models provide various additional hyperparameters that allow for fine-tuning and controlling the training process. We use *gensim's*[12] LDA implementation to train our models for $k = 500$ topics, on the vectorised corpus produced in the previous step.

3.3 Evolution Modeler

Evolution Modeler attempts to capture the evolution of the topics of the past time period into the models of the current period. To do so, the component calculates the *Jacard similarities* between each of the past topics with all the current topics taking into consideration the top-w words of each topic[13]. A benefit of using Jaccard similarities is that it focuses only on the co-existence of words in the topics; this is convenient because the topics of each model are built on different vocabularies (since they originate from distinct sets of texts). Any pair of topics (coming from different time periods) that is identified to have similarity which is larger than a predefined threshold are considered to be related. It is possible that one past topic will be similar to more than one current topics. This means

[9] https://pypi.org/project/langdetect/
[10] Our dataset: https://doi.org/10.5281/zenodo.4560609.
[11] We chose the PoS tagger and lemmatizer implementation provided by Python's package *nltk*.
[12] https://radimrehurek.com/gensim/
[13] Intuitively, good topics will assign large probability only to a small subset of the vocabulary. Topics with greater sparsity in their probability distribution, usually are too general or ambiguous. Thus, a topic's top-w words provides a sufficient representation.

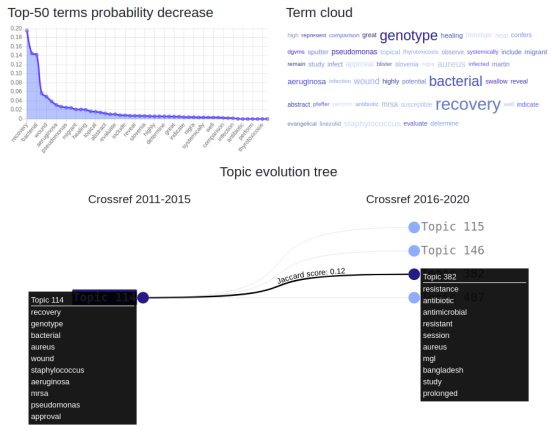

Figure 2: Topic information and visualisation page.

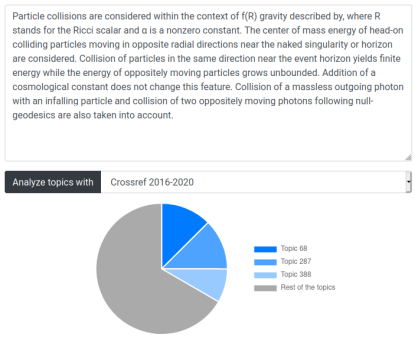

Figure 3: On-demand topic modeling.

that the former topic evolved into the latter ones, i.e., it was split into multiple topics. Similarly, multiple past topics may evolve into one particular current topic (topic merging). Finally there are past topics with no similar current topics (i.e., topics that seized to exist) and current topics with no similar past ones (i.e., novel topics).

3.4 Front-end UI

This component implements the Web-based UI that supports all the functionalities provided. It's implementation follows the MVC design pattern so that the code can be easily maintained and extended. The application's backend is implemented with a REST API though Python's *Django REST Framework*. The topic divergence database contains information regarding the models and their divergence. The user interface accessible to users, is developed with *React*.

4 FUNCTIONALITY

Our tool provides two main functionalities: (a) the option to monitor the evolution of all identified scientific topics and (b) the option to identify the topics of a user provided text, based on one of the already trained models.

Regarding the topic evolution monitor, the main page in the Web UI displays term clouds for all the identified topics. By default, the topics of the current corpus are displayed, but the user can select to also display the topics of the past corpus, as well. Each term cloud includes a button that redirects to a page containing information and visualisations about the corresponding topic (see Figure 2). This page contains the topic's term cloud and a diagram displaying the contribution probabilities for each topic term. Finally, the evolution of the topic is illustrated though an interactive graph that displays topic evolution relationships according to the Jaccard similarities. Old topics are displayed at the left, while new topics at the right.

Figure 3 illustrates the form that can be used to perform a topic analysis for a user-provided text. The user inserts their text in the form and selects the topic model to be used. After clicking on the analysis button, the tool responds with the top topics that are expressed by the text (a pie chart that shows the top-3 topics and their respective probabilities is displayed). For the depicted example, an abstract of a paper discussing particle collisions near naked singularities was used. The tool identified the connection to topic 287, which is very relevant (its predominant terms include *collision*, *photon*, *gravity* and *particle*, among others); the other two topics (68 and 388) seem to partially capture the notion of particle movement and cosmological constants.

5 DEMONSTRATION

During the workshop, the audience will be able to interact with our tool to examine its full capabilities. We will also demonstrate some interesting scenarios we have identified. Indicatively, as a first interesting case, a member of the audience browses the topic term clouds of the 2011 – 2015 period and, then, selects to display more details about Topic 114. The user examines each term's contribution reveals the topic has been split into four new topics in the model of the 2016 – 2020 period. In a second scenario, a member of the audience wants to reveal the topics of a paper of interest according to the model trained from the Crossref articles of the period 2016 – 2020. They insert the abstract of the paper in the on-demand topic modeling form, select the aforementioned model, and hit the analysis button. The top related topics are displayed in the screen in the form of a pie chart.

6 CONCLUSION

In this work, we demonstrated a prototype for the visualisation of scientific topic evolution based on multidisciplinary publication abstracts from Crossref. The tool first calculates two topic models based on these data, one for publications of the current 5 year period (2016 – 2020) and another one for those of a past period (2011 – 2015). Then, it consults the produced topic models to model and visualise the recent evolution of the main scientific topics of the captured literature. We plan to extend this preliminary work by investigating alternative configurations of LDA or even alternative topic modeling approaches. We also plan to implement extra visualisations and to evaluate their performance in terms on their usefulness.

ACKNOWLEDGMENTS

This research was partially funded by project ENIRISST under grant agreement No. MIS 5027930 (co-financed by Greece and the EU through the European Regional Development Fund).

REFERENCES

[1] David M. Blei and John D. Lafferty. 2006. Dynamic topic models. In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006 (ACM International Conference Proceeding Series)*, William W. Cohen and Andrew W. Moore (Eds.), Vol. 148. ACM, 113–120. https://doi.org/10.1145/1143844.1143859

[2] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *JMLR* (2003).

[3] Lutz Bornmann and Rüdiger Mutz. 2015. Growth rates of modern science: a bibliometric analysis based on the number of publications and cited references. *JASIST* 66, 11 (2015), 2215–2222.

[4] Serafeim Chatzopoulos, Panagiotis Deligiannis, Thanasis Vergoulis, Ilias Kanellos, Christos Tryfonopoulos, and Theodore Dalamagas. 2019. SciTo Trends: Visualising Scientific Topic Trends *(Lecture Notes in Computer Science)*, Vol. 11799. 393–396. https://doi.org/10.1007/978-3-030-30760-8_41

[5] Tommy Dang, Huyen N. Nguyen, and Vung Pham. 2019. WordStream: Interactive Visualization for Topic Evolution. In *21st Eurographics Conference on Visualization, EuroVis 2019 - Short Papers, Porto, Portugal, June 3-7, 2019*. Eurographics Association, 103–107.

[6] Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. 2019. The Dynamic Embedded Topic Model. *CoRR* abs/1907.05545 (2019). arXiv:1907.05545 http://arxiv.org/abs/1907.05545

[7] Ashwinkumar Ganesan, Kianté Brantley, Shimei Pan, and Jian Chen. 2015. LDAExplore: Visualizing Topic Models Generated Using Latent Dirichlet Allocation. *CoRR* abs/1507.06593 (2015). arXiv:1507.06593 http://arxiv.org/abs/1507.06593

[8] Ginny Hendricks, Dominika Tkaczyk, Jennifer Lin, and Patricia Feeney. 2020. Crossref: The sustainable source of community-owned scholarly metadata. 1, 1 (2020), 414–427. https://doi.org/10.1162/qss_a_00022

[9] Thomas Hofmann. 1999. Probabilistic Latent Semantic Analysis. In *UAI '99: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, Stockholm, Sweden, July 30 - August 1, 1999*, Kathryn B. Laskey and Henri Prade (Eds.). Morgan Kaufmann, 289–296.

[10] Wei Li and Andrew McCallum. 2006. Pachinko allocation: DAG-structured mixture models of topic correlations. In *Machine Learning, Proceedings of ICML 2006 (ACM International Conference Proceeding Series)*, William W. Cohen and Andrew W. Moore (Eds.), Vol. 148. ACM, 577–584.

[11] Sana Malik, Alison Smith, Timothy Hawes, Panagis Papadatos, Jianyu Li, Cody Dunne, and Ben Shneiderman. 2013. TopicFlow: visualizing topic alignment of Twitter data over time. In *Advances in Social Networks Analysis and Mining 2013, ASONAM '13, Niagara, ON, Canada - August 25 - 29, 2013*. ACM, 720–726.

[12] Thuc Nguyen and Phuc Do. 2018. Discovering Topic Evolution in Heterogeneous Bibliographic Network. In *10th International Conference on Knowledge and Systems Engineering, KSE 2018, Ho Chi Minh City, Vietnam, November 1-3, 2018*, Nguyen Thanh Thuy, Satoshi Tojo, Tan Hanh, Minh Le Nguyen, Tu Minh Phuong, and Vo Nguyen Quoc Bao (Eds.). IEEE, 91–96.

[13] Christos H. Papadimitriou, Prabhakar Raghavan, Hisao Tamaki, and Santosh S. Vempala. 1998. Latent Semantic Indexing: A Probabilistic Analysis. In *Proceedings of the ACM SIGACT-SIGMOD-SIGART Symposium*. ACM Press, 159–168.

[14] Feipeng Sun, Yanyan Li, and Zhiqiang Zhang. 2013. A Tool for Visualizing Topic Evolution in Large Text Collections. In *IEEE 13th International Conference on Advanced Learning Technologies, ICALT 2013, Beijing, China, July 15-18, 2013*. IEEE Computer Society, 53–54. https://doi.org/10.1109/ICALT.2013.21

[15] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. ArnetMiner: Extraction and Mining of Academic Social Networks. In *ACM SIGKDD*. 990–998.

[16] Thanasis Vergoulis, Serafeim Chatzopoulos, Ilias Kanellos, Panagiotis Deligiannis, Christos Tryfonopoulos, and Theodore Dalamagas. 2019. BIP! Finder: Facilitating Scientific Literature Search by Exploiting Impact-Based Ranking. In *Proceedings of CIKM 2019, 2019*. ACM, 2937–2940.

[17] Houkui Zhou, Huimin Yu, and Roland Hu. 2017. Topic evolution based on the probabilistic topic model: a review. *Frontiers Comput. Sci.* 11, 5 (2017), 786–802. https://doi.org/10.1007/s11704-016-5442-5