Exploring the Evolution of Science with Pivot Topic Graphs

Ke Li
LIP6, CNRS, Sorbonne Université
Paris, France
ke.li@lip6.fr

Hubert Naacke
LIP6, CNRS, Sorbonne Université
Paris, France
hubert.naacke@lip6.fr

Bernd Amann
LIP6, CNRS, Sorbonne Université
Paris, France
bernd.amann@lip6.fr

ABSTRACT

In this article we propose a data model for the visualisation and exploration of topic evolution networks representing the research progress in scientific document archives. Our model is independent of a particular topic extraction and alignment method and proposes a set of semantic and structural metrics for characterizing and filtering meaningful topic evolution patterns. These metrics are particularly useful for the visualization and the exploration of large topic evolution graphs. We also present a first implementation of our model on top of Apache Spark and experimental results obtained for three well-known document archives.

KEYWORDS

Topic Modeling, LDA, Science Evolution, Big data

1 INTRODUCTION

There is an increasing demand for practical tools to explore the evolution of scientific research published in bibliographic archives such as the Web of Science (WoS), ISTEX, arXiv or PubMed. Revealing meaningful evolution patterns from document archives has many applications and can be used to synthesize narratives from datasets across multiple domains, including news stories, research papers, legal cases and works of literature [16].

The *cognitive view* of scientific evolution emphasizes the shared knowledge and the change of ideas present in the document contents [13], whereas the *social view* takes account of authorship information and social interactions represented, for example, in co-authorship and citation graphs [8, 17]. Bibliographic archives often include both kinds of information and there also exist methods which combine both views to study science evolution [9]. In the interdisciplinary EPIQUE project[1], we adopt the cognitive view for modeling science evolution and assume that the evolution only depends on the textual document contents (title, abstract, main contents). This choice reduces the expressive power of our evolution model, but it also decreases the "social" bias and detects more easily possible interactions between scientific ideas and contributions, independently of any particular scientific community.

Graph-based topic evolution analysis builds on topic evolution networks which track complex temporal evolution dynamics by periodical topic discovery and similarity-based topic alignment. Figure 1 shows two snippets of a single topic evolution graph extracted from the arXiv[2] corpus. The graph covers the period between 2000 and 2006 decomposed into three overlapping time periods (3 year periods with one year overlap). Each topic is represented by a rectangle containing the top-10 topic terms obtained by a simple NLP document pre-processing step. Emerging terms are shown in green, decaying term boxes are colored in red, stable terms which exist both, in ancestor topics and in descendant topics, are grouped in a blue boxe and specific terms which appear only in the current topic are in white. The thickness of the alignment edges reflects the similarity of the connected topics. Several topics in both subgraphs contain the term "database" and we can observe different evolution patterns. The left subgraph shows that in period 2002 – 2004, topic 77 ("databases, queries, optimization, integration") splits in two research directions "databases, queries and constraints" (topics 100, 188) and "prediction, probability, random" (topics 104, 191, 152). The right subgraph covers the same period with topics related to "data mining" (83), "data access interfaces" (90), "information retrieval" (92), "logics, semantics" (80) and "knowledge, reasoning" (54). The first three topics converge in 2002 – 2004 into a single topic on "object, xml, store, data mining" (146) which splits in the period of 2004 – 2006 into "storage servers" (170), "data mining and management" (158) and "knowledge and ontologies" (150).

Building such meaningful topic evolution networks is still difficult and needs an important expertise in statistical text mining. A first challenge for domain experts is to correctly tune method specific hyper parameters with respect to a given dataset and an expected output. A second challenge concerns the visual exploration of large topic evolution networks. Whereas existing graph visualisation standards and tools like Gephi [3] or Graphviz [4] can be used to generate high-quality visualisations, their use for exploring large graphs and identifying meaningful evolution patterns is still limited. In this article we propose a generic evolution network computation and visualization framework which combines a high-level data model with big data technology for extracting and exploring topic evolution networks. The graph model relies on the notion of *pivot topic graphs*, which describe the contents and the evolution dynamics of topics at different levels of detail. The model also includes a number of high-level semantic metrics which enable domain experts to specify meaningful topic evolution patterns (queries) for exploring large topic evolution networks.

The remainder of this paper is organized as follows. The next section introduces the related work on topic evolution models and is followed by Section 3 which defines the EPIQUE topic evolution model including the evolution pattern metrics and a simple query language. Section 4 describes our workflow and gives an outline of the algorithms for building topic evolution networks. Section 5 illustrates some experimental results obtained by applying our evolution pattern metrics on three different document archives. The final section presents our conclusions and outlines future work.

[1]This work was funded by French ANR-16-CE38-0002-01 project EPIQUE
[2]https://arxiv.org/

[3]https://gephi.org/
[4]https://www.graphviz.org/

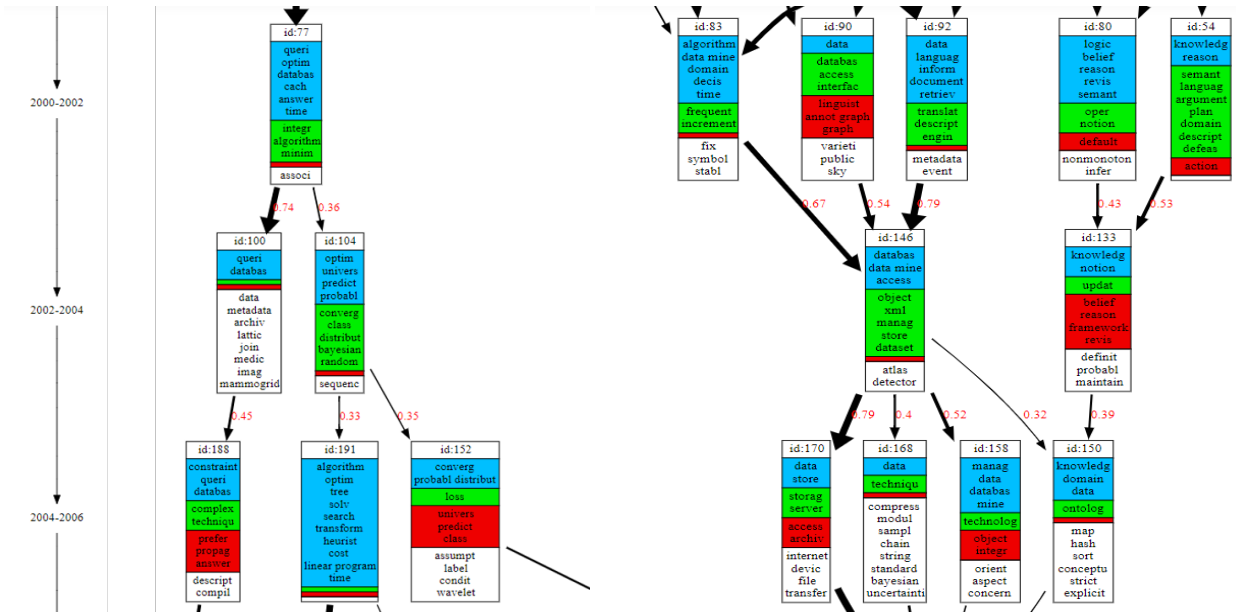


Figure 1: Pivot topics containing term "database" extracted from arXiv, green = emerging terms, blue = stable terms, red = decaying terms

2 RELATED WORK

Topic modeling is a text mining task which consists in extracting a compact representation of the content of a collection of documents. Statistical topic models like LDA [4] define a probabilistic procedure to generate documents as mixtures of a low-dimensional set of topics. The goal of dynamic topic model [3, 19, 20] is to capture the evolution of topics in a sequential document corpus. They generally achieve better predictive accuracy than static topic models which ignore the temporal dimension. In our work we use a simple topic evolution model where documents are organized into possibly overlapping time periods and LDA is applied on each document sub-collection independently from the other time slices. As our experiments show, the results we obtain with this strategy are already of good quality and the integration of dynamic topic models is an open future work.

Topic detection and trend analysis systems [12] aim to identify and follow event-based topics across incoming streams of documents. Usually, a tracking system is given seed documents to monitor the document stream for further documents on the same topic, whereas a detection system performs unsupervised clustering of the incoming document stream. For example, Hu et. al. [10] applied LDA and regression analysis to identify different topic evolution patterns for preprints and papers from arXiv and the Web of Science (WoS) in astrophysics for the last 20 years (1992 – 2011). The paper redefines the notion of topic trend and popularity, and demonstrates that open access preprints have stronger growth tendency as compared to traditional printed publications.

Trend analysis describes the temporal evolution of the popularity, the utility or interest of topics, but does not take account of their structural evolution where one topic can evolve into several sub-topics or several topics can merge into a single topic. Topic evolution networks represent this kind of structural topic evolution and track complex temporal changes by periodic topic discovery and directed acyclic networks aligning topics of different periods. Existing evolution network based frameworks mainly can be distinguished by the chosen topic extraction and alignment methods. [5] comes up with a method to enable a bottom-up reconstruction of the dynamics of scientific fields. They generate topics by word co-occurrence graphs and align inter-temporal topics by Jaccard similarity [11]. [1] generates topics by a Hierarchical Dirichlet Process (HDP) [18] and uses Bhattacharyya similarity [2], representing the gradual speciation and convergence similar to biologic evolution, for identifying topic alignments. The alignment process also applies (asymmetric) Kullback-Leibler divergence (KLD) for detecting topic split and merge. [15] introduces a novel approach to the early detection of research topics by using the Computer Science Ontology[5] to model research topics in the Rexplore system. They apply a Clique Percolation Method (ACPM) for analyzing the dynamics between existent topics. Other examples of science evolution studies explore how "cognitive science" as a field has changed over the last three decades [7] or analyze topic evolution patterns (split, merge and knowledge transfer) in the field of Information retrieval (IR) [6].

3 TOPIC EVOLUTION MODEL

This section presents the topic evolution model implemented in the EPIQUE workflow. The model is based on a multi-stage graph representation of topic evolution networks and introduces the notion of pivot evolution graphs with appropriate evolution pattern metrics. The sections also presents a simple query language for searching meaningful evolution patterns.

Topic Evolution Graphs. We consider a corpus C of time-stamped documents and a list P of periods. C_p denotes the corpus of documents having their timestamp in period $p \in P$. We consider set of topics T and denote $T_p \subseteq T$ the topics extracted from the documents in C_p. A topic $t = (v, p) \in T_p$ is defined by a (sparse) weighted term vector $v \in \mathbb{R}^{|V|}$ where V is a vocabulary of terms. We will denote by $t.v$ the terms and by $t.p$ the period of t. We also define a function $sim : T \times T \rightarrow [0, 1]$ estimating the *similarity* between topics in T. The similarity measure depends on

[5] http://cso.kmi.open.ac.uk/

the term vectors of topics and estimates their semantic proximity. For example in Figure 1, P has 3 periods: $p1$="2000 − 2002", $p2$="2002 − 2004" $p3$="2004 − 2006", T_{p1} contains topics 54 to 92, and topic 92 = $(v, p1)$, where v is a weighted vector containing terms "$queri$", "$optim$", "$databas$", ... with a positive weight. The similarity between topic 77 and topic 100 is $sim(77, 100) = 0.74$.

Based on the previous definitions, we define a *topic evolution graph* as a directed labeled *multistage* graph $\mathcal{G}^\beta = (T, E, sim, \beta)$ over T where the edges E connect topics from consecutive periods and their similarity is higher or equal to some threshold β. That is, $E = \{(t_i, t_j) \in T | sim(t_i, t_j) \geq \beta \wedge t_j.p = t_i.p + 1\}$.

Pivot Evolution Graphs. Threshold β strongly influences the complexity of the obtained evolution graphs. It is easy to see that $\mathcal{G}^{\beta'}$ is a subgraph of $\mathcal{G}^\beta$ for all $\beta' \geq \beta$ and $\mathcal{G}^0$ is the complete graph connecting all topics of two consecutive periods. More exactly, higher β values generate more "linear" graphs with many isolated topics, whereas lower values generate more complex and heterogeneous graphs containing a variety of potentially interesting structures. Analyzing science evolution by using topic evolution graphs then becomes a complex task which consists in computing and visually exploring multiple graphs for different β values.

To solve this problem, we propose a different approach which allows users to formulate *queries* for characterising and extracting interesting *subgraphs* from a set of evolution graphs defined by a set of β thresholds. For this, we decompose topic evolution graphs into the set of all connected subgraphs defined by all paths containing a given topic t (one graph per topic). More formally, a *pivot evolution graph* $\mathcal{G}^\beta(t) = (T', E', sim, \beta)$ *of topic* t in $\mathcal{G}^\beta$ is the subgraph of $\mathcal{G}^\beta$ which contains t and all ancestors and descendants of t. The subgraph of $\mathcal{G}^\beta(t)$ containing all nodes which are reachable from t is called the *future* of t, denoted by $\mathcal{F}^\beta(t)$, and the subgraph of nodes which reach t is called the *past* of t, denoted by $\mathcal{P}^\beta(t)$. The couple (t, β) is called a *pivot topic* with pivot graph $\mathcal{G}^\beta(t)$, future $\mathcal{F}^\beta(t)$ and past $\mathcal{P}^\beta(t)$. It is easy to see that if t_1 appears in the future (past) of t_2, then the future (past) of t_1 is a subgraph of the future (past) of t_2 and t_2 appears in the past (future) of t_1. This property can be exploited to filter topics *wrt.* future and past topics (see the definition of Path Filters below).

The evolution of topics within their evolution graphs can be characterized by the following metrics:

- The *liveliness* $live(\mathcal{G}^\beta(t))$ of a pivot topic (t, β) is defined by the diameter of its pivot graph $\mathcal{G}^\beta(t)$.

$$live(\mathcal{G}^\beta(t)) = max\{length(p) | p = \text{path in } \mathcal{G}^\beta(t)\}$$

  A high liveliness value describes a long living topic, whereas a value equal to 0 corresponds to an isolated topic without ancestors and descendants.
- The *relative evolution degree* $revol(\mathcal{G}^\beta(t))$ of a pivot topic (t, β) is defined by the average topic dissimilarity (edge) weight in $\mathcal{G}^\beta(t)$.

$$revol(\mathcal{G}^\beta(t)) = 1 - avg_{(t_i,t_j)\in E'}(sim(t_i, t_j))$$

  A low relative evolution degree states that most topics evolve slowly. On the other hand, a high value signifies that most topics have an important "semantic gap". By definition, we have $revol(\mathcal{G}^\beta(t)) \leq 1 - \beta$.
- The *pivot evolution degree* $pevol(\mathcal{G}^\beta(t))$ of a pivot topic (t, β) is defined by the average dissimilarity of all topics in $\mathcal{G}^\beta(t)$ with respect to the pivot topic t.

$$pevol(\mathcal{G}^\beta(t)) = 1 - avg_{t_i\in T'}(sim(t, t_i))$$

  A low pivot evolution degree signifies that the pivot topic does not evolve much (all other topics are similar), whereas a high value indicates that the pivot topic evolves rapidly .
- The *split degree* $split(\mathcal{G}^\beta(t))$ of a pivot topic (t, β) is defined by the average outdegree of $\mathcal{G}^\beta(t)$.

$$split(\mathcal{G}^\beta(t)) = \frac{|E'|}{|\{t_i | t_i \in T \wedge outdeg(t_i) > 0\}|}$$

  A low value signifies that the topics evolve along linear paths and a high value signifies that the topics split into several future sub-topics.
- The *convergence degree* $conv(\mathcal{G}^\beta(t))$ of a pivot topic (t, β) is defined by the average indegree of $\mathcal{G}^\beta(t)$.

$$conv(\mathcal{G}^\beta(t)) = \frac{|E'|}{|\{t_i | t_i \in T \wedge indeg(t_i) > 0\}|}$$

  A low value signifies that many topics depend on a single parent topic and a high value signifies that many topics are the result of the fusion of past topics.

Topic labeling. All topics t of some evolution graph $\mathcal{G}^\beta$ are labeled by the top-k highest weighted terms $t.l$ in the topic term vector $t.v$. Let $t.l_p \subseteq t.l$ and $t.l_f \subseteq t.l$ be the sets of past and future terms which appear, respectively, in the ancestor topics and in the descendant topics of t. Then, the terms in some topic vector $t.l$ are partitioned into the following four subsets of :

- *emerging* future terms $t.l_e = t.l_f - t.l_p$ which do not exist in past topics,
- *decaying* past terms $t.l_d = t.l_p - t.l_f$ which do not exist in future topics,
- *stable* terms $t.l_g = t.l_p \cap t.l_f$ which exist in the past and the future topics of t, and
- *specific* terms $t.l_s = t.l - (t.l_p \cup t.l_f)$ which neither exist in the past nor in the future topics of t.

The quadruple $[t.l_e, t.l_d, t.l_g, t.l_s]$ is called the *term label* of t.

Pivot Topic Query Language. Liveliness, relative evolution degree, pivot evolution degree, split degree and convergence degree allow to characterize the amount and complexity of the evolution of a topic in some evolution graph $\mathcal{G}^\beta$. Combined with other filters on the topic labels and the graph structure, it is possible to filter pivot topics satisfying rich evolution patterns within a set of evolution graphs $\mathcal{G}^{\beta_i}$, $1 \leq i \leq n$.

Let DB be the union of all future and past pivot topic graphs $\mathcal{F}^\beta(t)$ and $\mathcal{P}^\beta(t)$. Operators can be composed by concatenation (similar to Scala methods). For each attribute A we define a filter $A(X)$ where X is a valid value, and for each ordinal attribute the filter $A(X, Y)$ contains a second attribute Y restricting X to be the minimal ($Y = 0$) and maximal ($Y = 1$) value respectively.

- Term Filters select pivot graphs with respect to the pivot topic labels. In particular, they can be applied to filter pivot graphs *wrt.* to their emerging, decaying, stable, and specific terms. *Find all pivot topics where the term "deep learning" is emerging and the term "big data" is decaying*:

```
DB.Emerge("deep learning").Decay("big data")
```

- Temporal Filters allow the expert to filter all topics situated within a certain time period. *Find all topics between 2012 and 2017*:

```
DB.Period(2012,0).Period(2017,1)
```

- Pattern Filters can filter topics by their pivot graph structure along their liveliness, split degree and convergence degree. *Find all pivot graphs which cover at least 6 periods where all topics split into at least three subtopics in average*:

```
DB.Live(6,0).Split(3,0)
```

- Evolution Filters are applied to filter topics by their relative and pivot evolution degrees. *Find all topics which are evolving "slowly"*:

```
DB.Revol(0.3,1)
```

The previous filters are applied to sets of pivot topics and can be combined with other operators:

- Temporal Projection allows to project the pivot graph to its past and future. *Find all pivot topics with a linear future of a minimal length of 3 periods*:

```
DB.Future.Live(3,0).Split(1,1)
```

- Set Operations (union, intersection, minus) combine sets of topics. *Find all topics with decaying term "big data" and without emerging term "deep learning"*:

```
DB.Decay("big data")
  .Minus(DB.Emerge("deep learning"))
```

- Path Filters select pivot topics by the existence of a path to/from other topics. *Find all topics with an emerging term "deep learning" where the future contains a path to a topic with the decaying term "big data"*:

```
DB.Emerge("deep learning")
  .Future.Path(DB.Decay("big data"))
```

- Ordering: *Find all topics about "big data" ordered by period*:

```
DB.Term("big data").Sort("Period","asc")
```

4 IMPLEMENTATION

This section presents the implementation of the topic evolution model in the context of the EPIQUE project. Our implemented workflow is able to handle any scientific document corpus where each document has a publication date and some text content (title, abstract, keywords, ...).

EPIQUE Workflow. Figure 2 details the main steps of the EPIQUE workflow. The workflow starts with a standard document preprocessing step composed of some lexical analysis, stop-word removal, stemming, index term generation and term selection. The main goal of this step is to extract for each document a term index which precisely describes the scientific document contents. The document preprocessing step is followed by corpus periodization step which decomposes the document collection according to several continuous, possibly overlapping, time windows, *i.e.*, the same document may appear in two periods. Each time window defines a *corpus period*, which is the subset of documents published during the corresponding time period. The choice of the window size and sliding step depends on the granularity of the document time-stamps (year, month, day) and on the number of available documents in each period.

In the following step, each corpus period is analyzed by a topic model. In our implementation, we use LDA [4] to extract the topics of each corpus period. The main output of LDA is a topic-term matrix describing each topic as a weighted term vector. LDA requires to set the number of topics to be generated in advance. Tuning this parameter is important and subtle because it strongly influences the diversity of the topics generated for each time

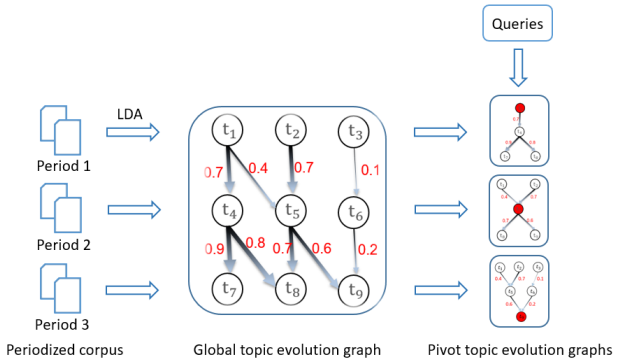


Figure 2: Topic evolution model of EPIQUE

period. We will illustrate in our experiments how experts can be assisted in choosing the right number of topics.

The topic-term vectors generated by the topic extraction step are combined with an appropriate similarity measure for aligning topics from different periods. In our experiments, we use cosine similarity which performs well in measuring the correlations between sparse vectors. Observe that the choice of LDA and cosine similarity does not exclude the use of other topic models and similarity measures like Jaccard similarity [11] or Battacharya similarity [2].

The next step produces instances of topic evolution graphs following the model introduced in Section 3. The topics are aligned to generate a single topic evolution graph $\mathcal{G}^{\beta_0}$ for some small alignment threshold β_0 (see the central part of Figure 2). This global evolution graph is then transformed into n families of pivot evolution graphs defined by a set of alignment thresholds $\beta_i > \beta_0$, $1 \leq i \leq n$. Each family contains the pivot graphs $\mathcal{G}^{\beta_i}(t)$ of all pivot topics (t, β_i). We only consider pivot graphs with at least one edge and ignore isolated pivot topics (single node graphs). The final database then contains at most $n \times |T|$ pivot graphs where $|T|$ is the number of topics in $\mathcal{G}^{\beta_0}$. These graphs can then be queried using the filters defined in Section 3 and we will illustrate the result of some queries in Section 5.

Pivot Graph Computation. We first present an algorithm that computes $\mathcal{G}^{\beta_i}$ for a sequence of relative evolution bounds β_0, β_1, ..., β_n where $\beta_i < \beta_{i+1}$. The basic idea of this algorithm is to exploit the monotonicity property that $\mathcal{G}^{\beta_{i+1}}$ is a subgraph of $\mathcal{G}^{\beta_i}$ for all $0 \leq i < n$. The input of the algorithm is a set of time-stamped topics T, a similarity function *sim* and a sequence of β_i values. Topics are represented as a binary table $Topics(t, a)$ storing the topics t and their periods a, the sequence of β_i values is defined as a binary table $Beta(b, i)$ where $b = \beta_i$ and the similarity function is defined as a table $Sim(x, y, s)$ where $s = sim(x, y)$. The following recursive Datalog program computes all $\mathcal{G}^{\beta_i}$ as a relational table $Graph(x, y, s, a, i)$ connecting all topics x of period a to all topics y of period $a + 1$ where there exists an evolution edge of similarity $s = sim(x, y) \geq \beta_i$.

```
Graph(x,y,s,a,0):- Topics(x,a), Topics(y,a+1),
                   Sim(x,y,s), Beta(b,0), s>=b
Graph(x,y,s,a,i):- Graph(x,y,s,i-1),
                   Beta(b,i), s>=b
```

Starting from *Graph* we can compute all pivot topic evolution graphs for all topics in T and beta value β_i. This is done by generating first a table $TC(x, y, s, l, a, i)$ containing the transitive

closure of graph $\mathcal{G}^{\beta_i}$ where l is the distance between x and y[6], a is the period of x and s is the similarity between x and y.

$$\begin{aligned}\mathbf{TC}(x,y,s,1,a,i) \quad &:-\mathbf{Graph}(x,y,s,a,i)\\ \mathbf{TC}(x,y,s,l+1,a,i) &:-\mathbf{TC}(x,z,\_,l,a,i),\\ &\quad\ \mathbf{Graph}(z,y,\_,\_,i),\mathbf{Sim}(x,y,s)\end{aligned}$$

This table can then be used to compute the future and the past graph for each pivot topic p.

$$\begin{aligned}\mathbf{Future}(p,x,y,rs,ps,l,a,i) &:-\mathbf{TC}(p,x,\_,\_,a,i),\\ &\quad\ \mathbf{TC}(p,y,ps,l,\_,i),\mathbf{Graph}(x,y,rs,\_,i)\\ \mathbf{Past}(p,x,y,rs,ps,l,a,i) &:-\mathbf{TC}(x,p,ps,l,a,i),\\ &\quad\ \mathbf{TC}(y,p,\_,\_,\_,i),\mathbf{Graph}(x,y,rs,\_,i)\end{aligned}$$

Graphs *Past* and *Future* contain the pivot topic evolution graphs of all topics where a tuple $(p, x, y, rs, ps, l, a, i)$ represents an edge (x, y) in $\mathcal{G}^{\beta_i}(p)$ for pivot p in period a with relative evolution similarity rs, pivot evolution similarity ps of x (*Past*) and y (*Future*) and distance l of x (*Past*) and y (*Future*) from p.

Table size estimation. We can estimate the size of each table as follows. Let p be the number of periods, t be the number of topics per period and k be the maximal outdegree and indegree in $\mathcal{G}^{\beta_i}$. Then the size of *Graph* is bound by $|Graph| \leq k * t * (p-1)$ (remind that *Graph* is a multistage graph). The size of the transitive closure *TC* is bound by $|TC| \leq k*t*p*(p-1)/2$. Finally, both graphs *Future* and *Past*, contain for each tuple (x, y, s, l, a, i) in the transitive closure *TC* (x and y are connected by a path of length l in $\mathcal{G}^{\beta_i}$), at most k tuples $Future(x, y, \_, \_, \_, \_)$ and at most k tuples $Past(y, x, \_, \_, \_, \_)$. Therefore, the size of *Future* and *Past* is bound by k times the size of the transitive closure: $|Future| \leq k^2 * t * p * (p-1)/2$. As our experiments show, even for small β-thresholds ($\beta = 0.2$) the maximum indegree and outdegree of a topics is smaller than $k = 10$ and we generally assume about $t = 100$ topics over $p = 20$ periods. Then, the size of the transitive closure is $|TC| \leq 10 * 100 * 10 * 19 = 1.9 * 10^5$ edges and the size of $|Future| \leq 1.9 * 10^6$ edges. These numbers are much smaller in practice (see Section 5) and current big data frameworks can easily manage graphs of this size. We plan to study possible optimizations in the future.

Metrics computation. The liveliness, relative evolution degree, pivot evolution degree, split degree and convergence degrees of all pivot evolution graphs can directly be computed by a standard SQL aggregation query. For example, the following query computes these metrics for all future pivot evolution graphs:

```
create view PivotFuture as
 select p,i max(l) as liveliness,
        1-avg(rs) as revol,
        1-avg(ps) as pevol,
        count(*)/count(distinct x) as split,
        count(*)/count(distinct y) as conv
 from Future
 group by p,i
```

Observe that for evaluating the pivot query filters presented in Section 3 without the *Path*-operator, it is sufficient to store *Graph* (for visualization) and *PivotFuture*, *PivotPast* and *PivotAll* (for filtering). An efficient implementation of *Path* queries, for example by using graph-labeling schemes for checking node reachability in acyclic graphs, is part of our future work.

[6]Since G_{β_i} is a multistage graph, all paths between two nodes are of the same length.

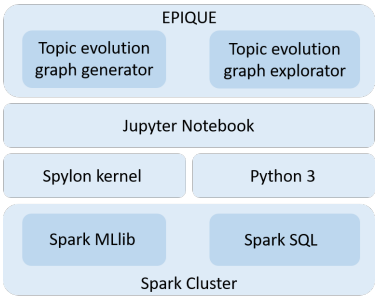


Figure 3: EPIQUE web application architecture

Architecture. Figure 3 gives an overview of the architecture of our web application implemented on top of Apache Spark and Jupyter Notebook. The entire process to study science evolution over a corpus is splitted into two steps for building the pivot evolution graphs and for interactively exploring these graphs. Each step corresponds to a separate user interface. The evolution graph generation is implemented in Scala and exectued through the Spylon[7] kernel. Evolution graph exploration uses a standard Python kernel to take advantage of advanced Python 3 graphical user interface libraries for facilitating user interaction.

5 EVALUATION

Experiment Setting. We conducted our experimentations on three real-world data sets of different scales by using the titles and the abstracts of each document. The smallest dataset, called ISTEX, contains 14 851 papers in the domain of ecological economics and environmental economics. The second one is arXiv, a repository of electronic preprints approved for publication after moderation, which consists of scientific papers in the fields of mathematics, physics, astronomy, electrical engineering, computer science, quantitative biology, statistics, and quantitative finance, etc. This data set contains about 1.1 million documents. The third dataset is from the Wiley online library which contains 1 million documents including additional domains such as agriculture, art, humanities, etc. The statistics over these three data sets are summarized in Table 1, where $\#D$ is the total number of documents, $\#T$ is the number of topics per period, $\#\mathcal{G}$ is the total number of pivot graphs, $\#E$ is the number of edges (total and per pivot graph).

We run our EPIQUE workflow on an Apache Spark cluster in standalone mode with Spark version 2.4, Scala version 2.11 and Java version 8. The cluster consists of 11 machines: one driver has 20GB memory, and 10 worker nodes, each one having 24 CPU cores with hyperthreading and 50 GB memory. For all of our experiments, we used the documents extracted from a 20-year period and split each corpus into 10 slices by using the time window spanning 3 years with 1-year overlap. Thus, we have 10 LDA models for each corpus.

Column *LDA* in Table 1 shows the average execution time for computing the *LDA* model per period. This computation is only done once and mainly depends on the number of extracted topics per period. In Table 1, we can see that the Wiley corpus is about 70 times larger than ISTEX but has a similar LDA execution time for the same number of topics. On the other hand, the LDA execution time is more important for arXiv, where we extract a higher number of topics.

[7]https://github.com/Valassis-Digital-Media/spylon-kernel

Table 1: Dataset statistics

Datasets	#D total	Period total	#Periods total	#T / period	#$\mathcal{G}$ total	#E total	#E per pivot	*LDA* sec / period	$\mathcal{G}$ sec (total)	$\mathcal{G}$ sec / pivot
ISTEX	14 851	1991 – 2010	10	20	1806	211 850	117	28	490	0.27
arXiv	1 156 300	1998 – 2017	10	50	3364	272 944	81	40	641	0.19
Wiley	1 023 515	1996 – 2015	10	20	1360	198 179	145	30	505	0.37

The last two columns $\mathcal{G}$ give the total and average execution time for the pivot graph computation step. We computed the pivot graphs for 9 β-threshold values spanning from 0.1 to 0.9. The total execution time obviously depends on the β-thresholds. The number of the thresholds increases the number of pivot graphs to be computed and the values of the thresholds defines the size of the computed pivot graphs. The average execution time to construct a pivot graph (last column) can be obtained by dividing the total time to build all pivot graphs by the number of pivot graphs. For example, for ISTEX corpus, we obtained 1806 pivot graphs in 490 seconds which gives the average value of 0.27 seconds. The average pivot graph computation time mainly depends on the pivot graph size. This can be seen for the arXiv dataset, which has the smallest average number of pivot graph edges ($\#E/pivot = 81$) and the lowest average pivot graph computation time (0.19 seconds/graph).

Pivot Topic Analysis. The metrics defined in Section 3 can be used for the structural and quantitative analysis of the evolution of topics. The objective of this section is to explore the impact of the main parameters,*i.e.*, the β threshold and the topic number $\#T$, on the structure and the semantics of the generated pivot evolution graphs.

Figure 4 shows the distribution of *future* pivot evolution graphs in arXiv *wrt.* three groups of metrics, the *relative evolution degree* vs. the *pivot evolution degree* , the *split degree* vs. *convergence degree* and the *liveliness* vs. the *split degree*. The figure is organized into 3 lines of 3 sub-graphs where each line corresponds to identical fixed parameters β and $\#T$ and each sub-graph corresponds to a group of metrics. On the first line, we set $\beta = 0.2$ and $\#T = 50$. On the second line $\#T$ remains the same ($\#T = 50$) whereas β is increased to $\beta = 0.5$. On the third line, β remains the same as in the 2nd line ($\beta = 0.5$) whereas $\#T$ is increased to $\#T = 150$. Each Figure only shows pivot topic graphs with at least two nodes and the number of isolated topics is reported in the figure captions.

When comparing Figure 4a with Figure 4d, we can see that for the lower threshold $\beta = 0.2$, pivot topics evolve more than for the higher value $\beta = 0.5$. Lower β values also allow pivot topics to connect with more topics than higher β values which only connect similar topics. This is shown in Figure 4b which represents a large number of complex pivot topic graphs with higher split and convergence degrees than the pivot topic graphs in Figure 4e. The previous observation is also confirmed in Figure 4c and Figure 4f which compare topic *liveliness* vs. split degree: the lower threshold $\beta = 0.2$ generates pivot graphs which are more complex than pivot graphs with the same liveliness scores generated by $\beta = 0.5$. Therefore, for a fixed $\#T$, varying β allows for revealing interesting evolution patterns at different levels of detail where the evolution of some topic might be too complex for low β values and become more intelligible for higher β values.

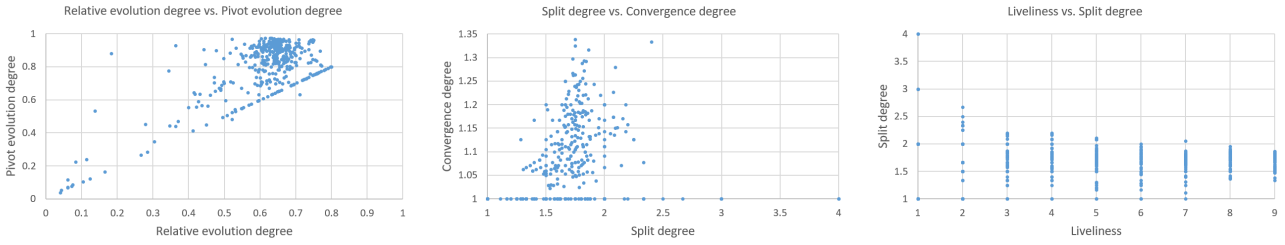
When the topic number per period increases ($\#T = 150$ in Figures (g), (h) and (i)), the workflow generates more pivot graphs, among which some become very complex. For example, in Figure 4g, pivot topics tend to evolve a lot even for a low relative evolution degree. The pivot graphs in Figure 4h are much more complex than the graphs generated by the same β-threshold with $\#T = 50$ topics (Figure 4e and Figure 4f). As we can see, the *split degree* attains a value of 19 compared with maximal *split degree* 1.5 in Figure 4e. The increase of $\#T$ reduces the proportion of isolated topics, 30% for $\#T = 150$ compared with 60% for $\#T = 50$. As we see in the next section, this is also due to the existence of many similar topics in each period, which also increases the probability that two topics can be aligned.

Diversity-based Topic Number Selection. Figure 5 shows a future arXiv pivot graph generated for $\#T = 150$ and $\beta = 0.5$, which corresponds to a data point in Figure 4h where $split(\mathcal{G}^{\beta}(t)) = 8.3$ and $conv(\mathcal{G}^{\beta}(t)) = 6.4$. The graph connects topics with similarity higher than $\beta = 0.5$ and has nevertheless a high split and convergence degree. When looking in more detail, we can observe that the topics in each period are also very similar which explains why the single root pivot topic is connected to more than 20 topics in the second period.

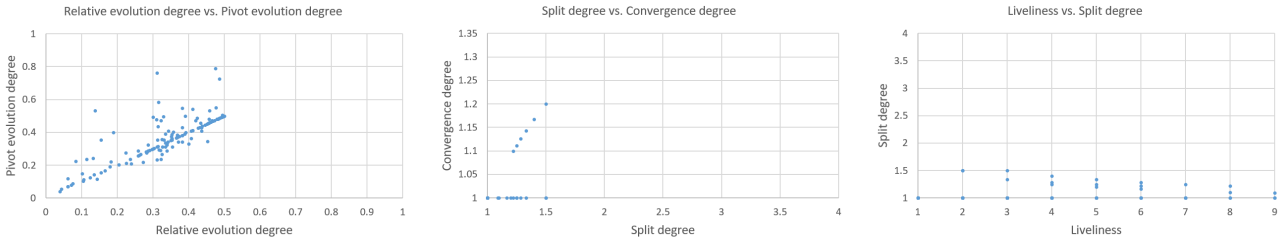
In order to build pivot graphs over more representative topic sets, we use *topic diversity* for estimating the quality of a topic set. The topic diversity inside a period can be estimated by observing the dissimilarity distribution over all topic pairs inside the period. For example, Figure 6a and Figure 6b shows the topic diversity obtained for different LDA models applied to 1164 documents published in arXiv during 1998 to 2000 and 16 072 documents published in arXiv during 2008 to 2010 respectively. Each LDA model corresponds to a different number of topics $\#T$ ranging from 10 to 150. For example, we can see in Figure 6a that for $\#T$ ranging between 40 and 60, less than 5 percent (blue line) of all topic pairs have a similarity value higher than 0.1 (dissimilarity value lower than 0.9), but this diversity value rapidly drops for $\#T > 60$. For example, for $\#T = 100$, the same similarity bound of 0.1 only holds for the half of the topic pairs (green median line). Figure 6b shows that for the larger corpus we can achieve the same diversity for much more topics (the topic diversity for $\#T = 140$ topics in period $2008 - 2010$ is similar to the topic diversity for $\#T = 60$ topics in period $1998 - 2000$. This kind of grid analysis allows experts to choose an optimal number of topics for their analysis.

Pivot Topic Exploration. Our query language allows users to select topics in specific regions of the sub-figures in Figure 4. For example, the following query $Q1$ chooses all topics which appear in the upper right window of Figures 4a and on the right part of Figure 4b on the line corresponding to the liveliness value 5 in Figure 4c.

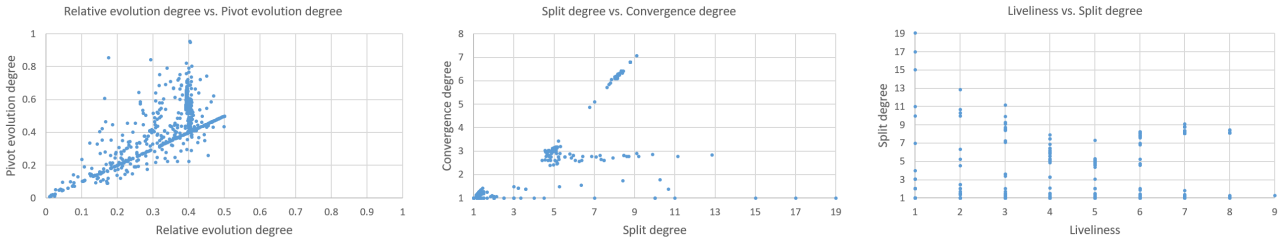
```
Q1:=DB.Future.Revol(0.5,0).Pevol(0.6,0)
      .Split(2,0).Live(5)
```



(a) $\beta = 0.2$, $\#T = 50$, #Pivot = 477, #Isolated = 23 (b) $\beta = 0.2$, $\#T = 50$, #Pivot = 477, #Isolated = 23 (c) $\beta = 0.2$, $\#T = 50$, #Pivot = 477, #Isolated = 23



(d) $\beta = 0.5$, $\#T = 50$, #Pivot = 198, #Isolated = 302 (e) $\beta = 0.5$, $\#T = 50$, #Pivot = 198, #Isolated = 302 (f) $\beta = 0.5$, $\#T = 50$, #Pivot = 198, #Isolated = 302



(g) $\beta = 0.5$, $\#T = 150$, #Pivot = 1014, #Isolated = 486 (h) $\beta = 0.5$, $\#T = 150$, #Pivot = 1014, #Isolated = 486 (i) $\beta = 0.5$, $\#T = 150$, #Pivot = 1014, #Isolated = 486

Figure 4: Distribution of future pivot evolution graphs in arXiv *wrt.* their metrics.

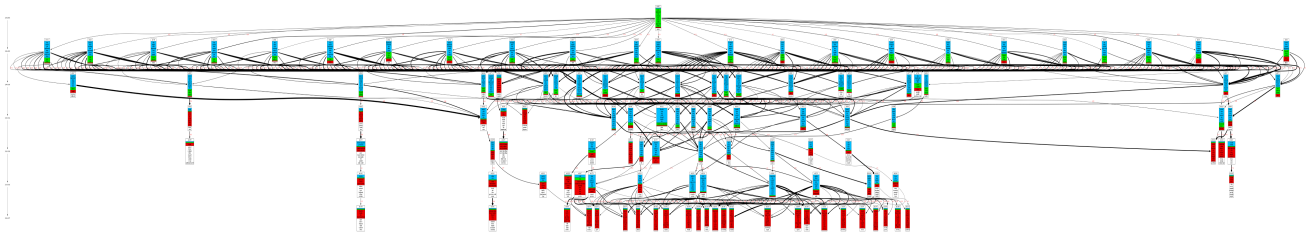
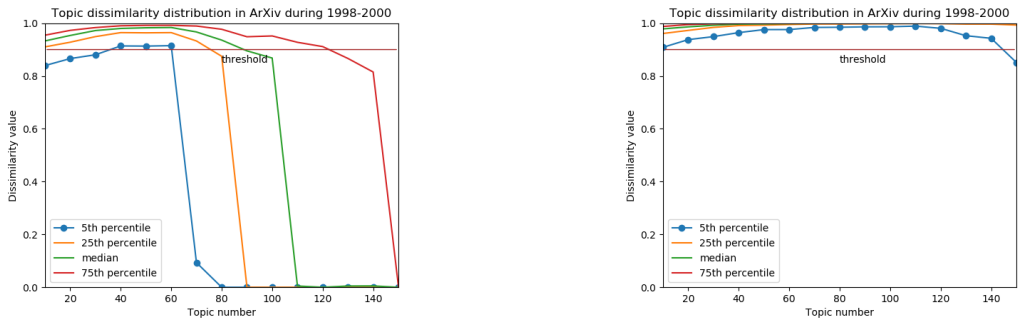
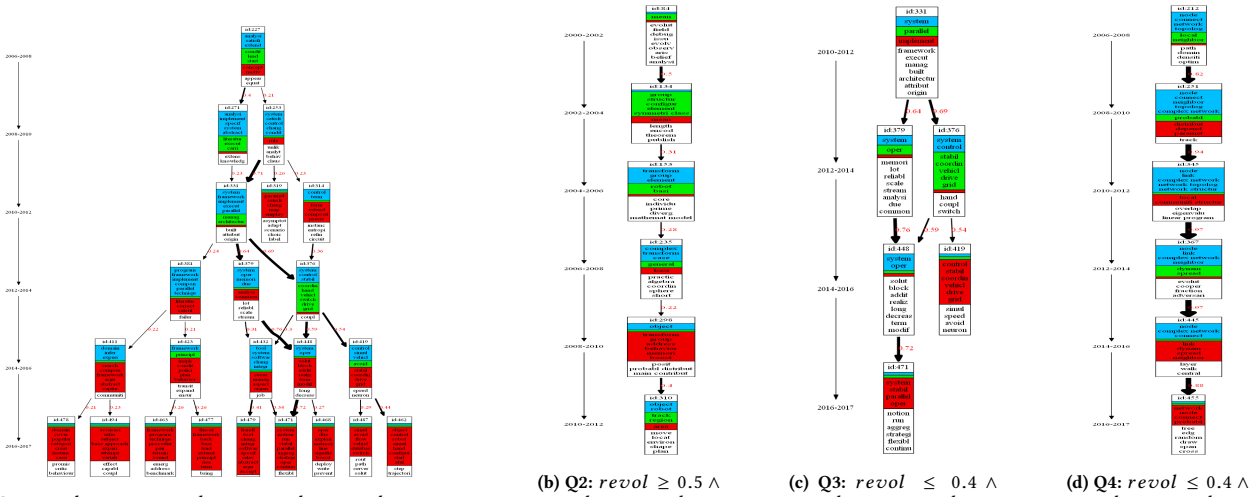


Figure 5: $\mathcal{G}^{0.5}(495)$ with $\#T = 150$



(a) 1998-2000 (b) 2008-2010

Figure 6: Dissimilarity distribution by topic number



(a) Q1: $revol \geq 0.5 \wedge pevol \geq 0.6 \wedge split \geq 2 \wedge live = 5$

(b) Q2: $revol \geq 0.5 \wedge pevol \geq 0.6 \wedge split \leq 1.2 \wedge live = 5$

(c) Q3: $revol \leq 0.4 \wedge pevol \leq 0.5 \wedge split \geq 1.5 \wedge live = 3$

(d) Q4: $revol \leq 0.4 \wedge pevol \leq 0.5 \wedge split \leq 1.2 \wedge live = 5$

Figure 7: Examples of query results which filter pivot topics by using aforementioned metrics

Observe that the user does not specify the β-threshold. A result example of query $Q1$ and three other queries is shown in Figure 7. Although Figure 7b and Figure 7d have the same structure, they have different evolution pace (corresponding to different β values). The pivot graph in Figure 7b has more emerging terms (green part) whereas the pivot graph in Figure 7d has more stable terms (blue part) which correspond to our queries to select high-evolution and low-evolution pivot topics respectively.

Apart from these metric-based filters, our query language also allows users to define other multi-dimensional filtering criteria including topic labels and temporal conditions for the selection of pivot topics.

6 CONCLUSION AND FUTURE WORK

We have presented a new framework for the visualisation and exploration of topic evolution networks representing the progress and evolution of research in scientific document archives. This framework has been implemented on top of Apache Spark using LDA and cosine similarity for topic extraction and topic alignment. The user can express complex evolution pattern queries to obtain the relevant pivot topic graphs. A first prototype [14] is currently used to extract complex evolution patterns for different scientific domains as part of the EPIQUE project and in collaboration with philosophers of science. As future work we intend to optimize the computation of pivot topic evolution graphs and exploit the LDA document-topic matrix for enriching the analysis. Additionally, we plan to integrate other topic extraction methods than LDA.

REFERENCES

[1] Victor Andrei and Ognjen Arandjelović. 2016. Complex temporal topic evolution modelling using the Kullback-Leibler divergence and the Bhattacharyya distance. *EURASIP Journal on Bioinformatics and Systems Biology* 2016, 1 (2016), 16.

[2] A. Bhattacharyya. 1943. On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society* 35 (1943), 99–109.

[3] David M. Blei and John D. Lafferty. 2006. Dynamic Topic Models. In *Proceedings of the 23rd International Conference on Machine Learning (ICML '06)*. ACM, New York, NY, USA, 113–120.

[4] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.

[5] David Chavalarias and Jean-Philippe Philippe Cointet. 2013. Phylomemetic patterns in science evolution—the rise and fall of scientific fields. *PloS one* 8, 2 (2013), e54847.

[6] Baitong Chen, Satoshi Tsutsui, Ying Ding, and Feicheng Ma. 2017. Understanding the topic evolution in a scientific domain: An exploratory study for the field of information retrieval. *Journal of Informetrics* 11, 4 (2017), 1175–1189.

[7] Uriel Cohen Priva and Joseph L. Austerweil. 2015. Analyzing the history of Cognition using Topic Models. *Cognition* 135 (Feb. 2015), 4–9.

[8] Eugene Garfield. 1955. Citation Indexes for Science: A New Dimension in Documentation through Association of Ideas. *Science* 122, 3159 (July 1955), 108–111.

[9] Qi He, Bi Chen, Jian Pei, Baojun Qiu, Prasenjit Mitra, and Lee Giles. 2009. Detecting Topic Evolution in Scientific Literature: How Can Citations Help?. In *ACM Conf. on Information and Knowledge Management (CIKM '09)*. New York, NY, USA, 957–966.

[10] Beibei Hu, Xianlei Dong, Chenwei Zhang, Timothy D. Bowman, Ying Ding, Staša Milojević, Chaoqun Ni, Erjia Yan, and Vincent Larivière. 2015. A Lead-lag Analysis of the Topic Evolution Patterns for Preprints and Publications. *J. Assoc. Inf. Sci. Technol.* 66, 12 (Dec. 2015), 2643–2656.

[11] Paul Jaccard. 1912. The Distribution of the Flora in the Alpine Zone.1. *New Phytologist* 11, 2 (1912), 37–50.

[12] April Kontostathis, Leon M. Galitsky, William M. Pottenger, Soma Roy, and Daniel J. Phelps. 2004. A survey of emerging trend detection in textual data mining. In *Survey of text mining*. Springer, 185–224.

[13] Thomas S. Kuhn, Otto Neurath, and Thomas Samuel Kuhn. 1994. *The Structure of scientific revolutions* (2nd ed., enlarged ed.). Number ed.-in-chief: Otto Neurath ; Vol. 2 No. 2 in International encyclopedia of unified science Foundations of the unity of science. Chicago Univ. Press, Chicago, Ill. OCLC: 258260085.

[14] Ke Li, Hubert Naacke, and Bernd Amann. 2020. EPIQUE: Extracting Meaningful Science Evolution Patterns from Large Document Archives (Demonstration). In *Int'l Conf. on Extending Database Technology (EDBT)*. Copenhagen, Denmark.

[15] Angelo A. Salatino, Francesco Osborne, and Enrico Motta. 2018. AUGUR: Forecasting the Emergence of New Research Topics. In *ACM/IEEE on Joint Conference on Digital Libraries (JCDL '18)*. ACM, New York, NY, USA, 303–312.

[16] Dafna Shahaf, Carlos Guestrin, Eric Horvitz, and Jure Leskovec. 2015. Information Cartography. *Commun. ACM* 58, 11 (2015), 62–73.

[17] Xiaoling Sun, Jasleen Kaur, Staša Milojević, Alessandro Flammini, and Filippo Menczer. 2013. Social Dynamics of Science. *Scientific Reports* 3 (Jan. 2013), 1069. https://doi.org/10.1038/srep01069

[18] Yee W Teh, Michael I Jordan, Matthew J Beal, and David M Blei. 2005. Sharing clusters among related groups: Hierarchical Dirichlet processes. In *Advances in neural information processing systems*. 1385–1392.

[19] Chong Wang, David Blei, and David Heckerman. 2008. Continuous Time Dynamic Topic Models. In *Conference on Uncertainty in Artificial Intelligence (UAI'08)*. AUAI Press, Arlington, Virginia, United States, 579–586.

[20] Xuerui Wang and Andrew McCallum. 2006. Topics over Time: A non-Markov Continuous-time Model of Topical Trends. In *Int'l Conf. on Knowledge Discovery and Data Mining (KDD '06)*. ACM, New York, NY, USA, 424–433.