Finding Topic Trends in Digital Libraries

Levent Bolelli[1], Seyda Ertekin[2], Ding Zhou[3], C. Lee Giles[2]

[1]Google Inc.
76 Nineth Ave
New York, NY 10011
levent@google.com

[2]The Pennsylvania State University
University Park, PA 16802
sertekin@cse.psu.edu
giles@ist.psu.edu

[3]Facebook Inc.
285 Hamilton Avenue
Palo Alto, CA 94301
dzhou@psu.edu

ABSTRACT

We propose a generative model based on latent Dirichlet allocation for mining distinct topics in document collections by integrating the temporal ordering of documents into the generative process. The document collection is divided into time segments where the discovered topics in each segment is propagated to influence the topic discovery in the subsequent time segments. We conduct experiments on the collection of academic papers from CiteSeer repository. We augment the text corpus with the addition of user queries and tags and integrate the citation graph to boost the weight of the topical terms. The experiment results show that segmented topic model can effectively detect distinct topics and their evolution over time.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*data mining*

General Terms

Algorithms, Design, Experimentation

1. INTRODUCTION

Automatic identification of semantic content of documents has become increasingly important due to its effectiveness in many tasks, including information retrieval, information filtering and organization of documents collections in digital libraries. The identification of the topic(s) that a document addresses increases our understanding of that document, the characteristics of the collection as a whole and the interplay between distinct topics. In collections where the temporal ordering of documents is not of importance, studying a snapshot of the collection at any given time is sufficient to deduct as much information as possible about the various topics of interest in the collection. On the other hand, many document collections exhibit temporal relationships that is often times utilized to aid the topic discovery process. Scientific literature is one of the fields that exhibits strong temporal relationship between the documents where the popularity of the research topics that are addressed in the papers change over time. Figure 1 shows the number of times that three Computer Science terms have been mentioned in the abstracts of the articles published by ACM from 1990 until 2004. Among those terms, the popularity of *expert systems* has been on a steep decline, whereas the number of papers that mention *Hidden Markov Model* and *Support Vector Machine* has been increasing. Another interesting fact to note is that *Support Vector Machine* is virtually non-existent in the collection until 1997, according to ACM repository. It is clear that popularity of topics vary over time, new topics emerge and some topics cease to exist. Thus, capturing such topic dynamics through the integration of the temporal order of documents into the topic discovery process can potentially yield more accurate topic analysis of document collections.

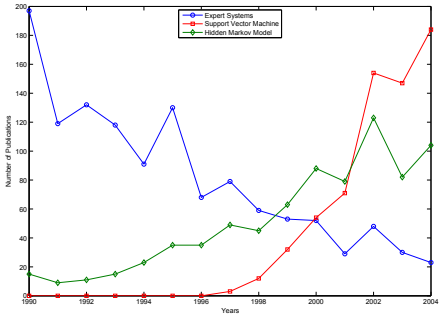
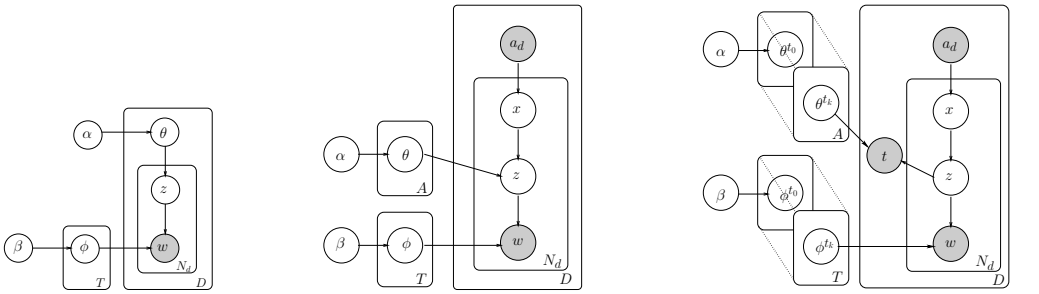


Figure 1: The popularity of sample key phrases mentioned in papers over 15 years.

We propose a generative model of documents, namely Segmented Author-Topic Model (S-ATM), that utilizes the temporal ordering of documents to assist the process of topic discovery. S-ATM is based on the Author-Topic Model [3] and extends it to integrate the temporal characteristics of the document collection into the generative process. Furthermore, we augment the text corpus of the articles with user queries from CiteSeer and user assigned tags of papers from CiteULike, and we utilize the citation relationship between papers to discover the *topicality* of words and boost the weight of those words to improve the quality of the discovered topics.

JCDL'09, June 15–19, 2009, Austin, Texas, USA.



9(a) Topic Model 9(b) Author-Topic Model 9(c) Segmented Author-Topic Model

Figure 2: Topic Model, Author-Topic model and Segmented Author-Topic Model in plate notation.

2. SEGMENTED AUTHOR-TOPIC MODEL

The Segmented Author-Topic Model (S-ATM) eliminates the exchageability assumption of the traditional Author-Topic Model (ATM) by sequential modeling of documents. The model segments the document collection into time slices $\mathfrak{t}_k = \mathfrak{t}_0 + k\Delta\mathfrak{t}$ where $\mathfrak{t}_0$ denotes the earliest timestamp, $\Delta\mathfrak{t}$ is the size of time slice and $k = [0 \cdots n]$ is a particular time segment. In S-ATM, the generation of a document starts with a group of authors $\mathbf{a_d}$ deciding on writing a document d. Each topic has a multinomial distribution over words and each author has a multinomial distribution over topics. A document with multiple authors has a distribution over topics that is a mixture of the topic distributions of authors. For each word w in document d, an author of d is chosen uniformly from the set of authors a_d of the document, and a word is generated through sampling a topic from the multinomial distribution of the chosen author over all topics. In the model, author-topic distributions θ have a symmetric Dirichlet prior with a hyperparameter α and word distributions of topics ϕ have a symmetric Dirichlet prior with a hyperparameter β. In broad terms, S-ATM extends ATM by performing the generative process of the collection in the temporal order of time segments, and utilize the learned past distributions as prior knowledge in subsequent iterations of S-ATM.

2.1 Gibbs Sampling for the Estimation of Model Parameters

For each word w_i, the topic z_i and the author x_i responsible for this word are assigned based on the posterior probability $P(z_i, x_i | w_i, \mathbf{z_{-i}}, \mathbf{x_{-i}}, \mathbf{w_{-i}}, \mathbf{a_d})$ that is conditioned on all other variables: z_i and x_i denote the topic and author assigned to w_i, and $\mathbf{z_{-i}}$ and $\mathbf{x_{-i}}$ are all other assignments of topic and author, excluding current instance. $\mathbf{w_{-i}}$ represents other observed words in the document set and $\mathbf{a_d}$ is the observed author set for the document. Gibbs sampling estimates the probability for T topics and V words as follows:

$$P(z_i = j, x_i = k | w_i = m, \mathbf{z_{-i}}, \mathbf{x_{-i}}, \mathbf{w_{-i}}, \mathbf{a_d}) \propto \quad (1)$$

$$P(w_i = m | x_i = k) P(x_i = k | z_i = j) \propto \quad (2)$$

$$\frac{C^{WT}_{mj} + \beta}{\sum_{m'} C^{WT}_{m'j} + V\beta} \frac{C^{AT}_{kj} + \alpha}{\sum_{j'} C^{AT}_{kj'} + T\alpha} \quad (3)$$

where $m' \neq m$ and $j' \neq j$, α and β are prior parameters for topic and word Dirichlets, C^{WT}_{mj} represents the number of times that word $w_i = m$ is assigned to topic $z_i = j$, C^{AT}_{kj} represents the number of times that author $x_i = k$ is assigned to topic j. The variables $\mathbf{z_{-i}}$, $\mathbf{x_{-i}}$, $\mathbf{w_{-i}}$ are dropped in the transformation from Eq. 1 to Eq. 2 due to the independence assumption of the words. We can then estimate $P(w_i = m | z_i = r)$ and $P(z_i = r | x_i = q)$ from the topic-word distribution ϕ and author-topic distribution θ, respectively:

$$P(w_i = m | z_i = r) \propto \frac{C^{WT}_{mr} + \beta}{\sum_{m'} C^{WT}_{m'r} + V\beta} \quad (4)$$

$$P(z_i = r | x_i = q) \propto \frac{C^{AT}_{rq} + \alpha}{\sum_{r'} C^{AT}_{r'q} + T\alpha} \quad (5)$$

The iteration at time $\mathfrak{t}_0$ starts with random initialization of author-topic assignments C^{AT} and topic-word assignments C^{WT} which, at the end of the training, yields us the author-topic distributions $\theta^{\mathfrak{t}_0}$ and and topic-word distributions $\phi^{\mathfrak{t}_0}$. Each subsequent iteration then utilizes the distributions obtained in the previous iterations to initialize the assignments for the current time segment as follows:

$$C^{AT}_{rq,\mathfrak{t}_k} = \lambda \mathfrak{R}(C^{AT}) + (1 - \lambda) \sum_{i=0}^{k-1} (\frac{1}{2})^{k-i} \theta^{t_i}_{rq} \quad (6)$$

and the initialization of the topic-word assignments becomes

$$C^{WT}_{mr,\mathfrak{t}_k} = \lambda \mathfrak{R}(C^{WT}) + (1 - \lambda) \sum_{i=0}^{k-1} (\frac{1}{2})^{k-i} \phi^{t_i}_{mr} \quad (7)$$

9

where $\mathfrak{R}(\cdot)$ adds random noise to the initialization by assigning topics to authors in Eq. 6 and words to topics in Eq. 7 independent of the prior knowledge obtained from $(\theta^0, \theta^1, \cdots, \theta^{k-1})$ and $(\phi^0, \phi^1, \cdots, \phi^{k-1})$, respectively. The initialization places higher emphasis on recent distributions than earlier ones through a decay component. This enables the learner to integrate all prior knowledge it has gathered so far with varying levels of confidence. Since we train the model on each time segment while propagating knowledge from previous segments, the distributions $\theta^{\mathfrak{t}_k}$ and $\phi^{\mathfrak{t}_k}$ only contain the topic probabilities of authors and words seen until t_k. Hence, at the start of the initialization of a new segment $\mathfrak{t}_{k+1}$, the model may find a new author a', or a new

Table 1: Sample CiteSeer queries and CiteULike tags of five papers in the dataset for our experiments.

Paper Title	CiteSeer Queries	CiteULike Tags
Partitioning-Based Clustering for Web Document Categorization	[*document categorization*] [*soft clustering of web pages*]	[*clustering*] [*partitioning*] [*web*]
A Call-By-Need Lambda Calculus	[*online lambda reduction*] [*lambda syntax*]	[*algorithm*] [*lisp*] [*lambda calculus*]
Item-based Collaborative Filtering Recommendation Algorithms	[*item recommendation*] [*recommender algorithms*]	[*collaborative filtering*] [*recommender systems*] [*social networks*]
Using Web Structure for Classifying and Describing Web Pages	[*view web structure*] [*classifying web pages*]	[*automatic classification*] [*information organization*]
A Min-max Cut Algorithm for Graph Partitioning and Data Clustering	[*min max clustering*] [*data clustering bisection*]	[*clustering*] [*spectral*] [*mincut*]

word w', in which case the distributions $\theta^{\mathbf{t}_i}_{a'm}$ and $\phi^{\mathbf{t}_i}_{mw'}$, $i = [0, \cdots, k]$ $m = [1, \cdots, T]$ will be zero, denoting that we don't have prior knowledge for that particular observation. The parameter λ determines the amount of prior knowledge that we want to propagate to subsequent segments. In our experimental settings, we do not estimate the hyperparameters α and β and set fixed smoothing parameters at $50/T$ and 0.01, respectively, where T is the number of topics.

3. EXPERIMENTS

3.1 Synthetic Dataset

The synthetic dataset consists of five author communities with each having a unique distribution over ten topics. In the collection, each topic is modeled as a distribution over 200 words. For each author, we randomly sample a community and the author generates words that follow the topic distribution of that community. In order to model topic dynamics, each community is modeled over three consecutive time segments, where the community's topic distributions vary over those segments. At each time segment, the authors generate words that follow the topic distributions of his community for that particular time segment. The dataset consists of 1000 authors that we have observations over those three time segments. We ran the author-topic model on each time segment to achieve a baseline comparison for S-ATM. ATM randomly initializes the author-topic and topic-word distributions at the beginning of each time segment where the S-ATM partially integrates the prior knowledge, softened through random noise. In order to reduce the effect of random start at the first time segment, the results of both algorithms are averaged over 100 runs with 1000 iterations of Gibbs Sampling and with $\lambda = 0.5$. We compare precision-at-1 accuracies to assess whether the authors have been correctly classified to their respective topics.

The comparative results for both algorithms are given in Figure 3. S-ATM is theoretically the same as ATM for the first time segment since there is no prior knowledge that it can utilize. Hence, both algorithms are initialized randomly and yield similar results. In time segments 2 and 3 where the communities transition from one topic to another, ATM tends to lose its accuracy as it starts to predict incorrect topics for authors, since they do not have distinctly pronounced topic memberships as is the case for time segment 1. S-ATM, on the other hand, utilizes the knowledge about the past memberships of authors to communities and hence, classifies authors to their respective communities with higher accuracy.

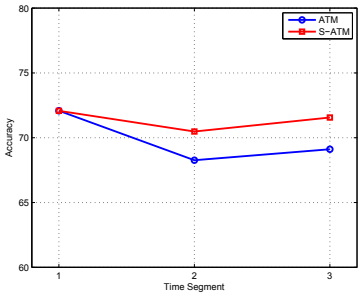


Figure 3: Comparison of ATM and S-ATM on the synthetic dataset.

3.2 Experiments on the CiteSeer Collection

Our collection of CiteSeer documents consists of a set of scientific articles published over 15 years between 1990 and 2004. In total, there are 41,540 documents published by 35,314 authors. We used the title, abstract and keywords fields from the documents and preprocessed the text by removing all punctuation and stop words, yielding a vocabulary size of 25,101 distinct words.

We also augment the text corpus with additional information. Once a scientific paper is published, it becomes an immutable document in the sense that the content of the document does not change over time. However, the environments that the documents reside in, such as search engines and digital libraries, continue to gather additional semantic information for the document in terms of *queries* and *tags*, which we utilize to aid the topic discovery process.

User Queries and Tags: From the access logs of CiteSeer, we selected the user queries to the documents in our collection and kept the queries with less 50 characters in length. Our observations from the analysis of logs indicate that longer queries are navigational in nature and tend to seek exact title match of the papers, whereas shorter queries have more informational characteristics and comprise of only a couple of concise key terms that highlight the topical nature of the clicked documents. After preprocessing CiteSeer's user logs, we identified 246,902 queries for the documents in our sample collection.

The CiteULike tag dataset contains 6527 unique tags for all of the documents in CiteSeer's repository. In the sample dataset for our experiments, we identified 2919 tags with 1012 unique words which we integrated into the corpus.

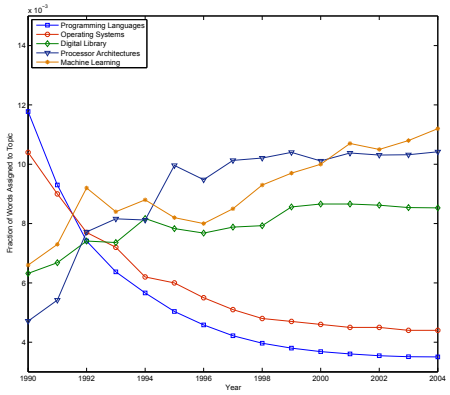


Figure 4: Topic trends for five research topics in Computer Science discovered in CiteSeer collection.

Boosting Topical Terms: Document collections with explicit link graphs, such as web pages with hyperlinks and scientific papers with citation graphs can be better analyzed when both the textual contents and the links are used in a unified way. Those two heterogeneous sources have been utilized in various application domains, including design of focused crawlers [1], web page clustering [4] and scientific paper clustering [2]. We adopt the citation-based topical term identification from [2] that ranks the most important topic-bearing terms based on the existence and absence of citations between papers using Expected Entropy Loss measure. From the entropy-ranked list of terms, we selected the top %10 of the vocabulary as the cut-off for term boosting. Overall, we assigned twice the weight to queries, tags and top %10 topical words than regular words in the content of the documents.

3.3 Experimental Results for CiteSeer

We report four sample topics and the most influential authors for those topics discovered by S-ATM in CiteSeer repository. The topics are extracted from a single sample at the 1000th iteration of the Gibbs sampler with a model distribution propagation parameter $\lambda = 0.5$.

We show the popularity trends of sample topics discovered by S-ATM in Figure 4. The popularity of topics are calculated by the fraction of words assigned to each topic for a year for all topics and for each year from 1990 to 2004. It can be seen that the popularity of machine learning topic has been steadily increasing over those years, due to widespread interest of its applications in many research areas. This can also be evidenced from the evolution of Topic 92 in Table 2. The supervised learning term *classifiers* emerges as one of the top words for the image processing topic. We observe the stabilization of the popularity of Digital Library and Processor Architectures topics.

4. CONCLUSIONS

This paper presents a generative model of documents that iteratively learns author-topic and topic-word distributions for scientific publications while integrating the temporal order of the documents into the generative process. The model parameters are estimated using Gibbs sampling and the distributions learned for each year are used as priors for the probabilities in the subsequent years. In addition to the textual content of the papers, we utilize user queries, tags and employ citation-based topic boosting to improve the topic discovery process. Our quantitative evaluation on a synthetic dataset as well as the application of S-ATM to a sample dataset from CiteSeer repository indicates that we can effectively discover scientific topics and most influential authors for the topics, as well as the evolution of topics over time.

Topic 8					
1990		**1994**		**2004**	
memory	.11255	memory	.11907	dynamic	.08096
random	.07198	dynamic	.07533	memory	.07993
disk	.06544	storage	.05643	access	.06774
access	.06369	access	.04582	random	.04634
consistency	.05017	shared	.04079	low	.03792
Author	**Prob.**	**Author**	**Prob.**	**Author**	**Prob.**
Patterson_D	.04036	Larus_J	.03709	Kandemir_M	.02275
Chen_P	.03814	Grunwald_D	.03683	Dubois_M	.01885
Soffa_M	.02478	Ball_T	.02689	Jouppi_N	.01817
Topic 23					
1990		**1994**		**2004**	
graph	.14944	graph	.16844	networks	.12200
process	.09876	routing	.08812	search	.08946
routing	.06919	process	.07256	graph	.08486
architecture	.06688	architecture	.06580	routing	.07542
computation	.04859	networks	.06108	process	.06778
Author	**Prob.**	**Author**	**Prob.**	**Author**	**Prob.**
Kaiser_G	.03190	Ranka_S	.07624	Wang_J	.04217
Perry_D	.02717	Mehtora_K	.06770	Sen_S	.04186
Gupta_R	.01883	Lilja_D	.05937	Morris_R	.02637
Topic 48					
1990		**1992**		**2004**	
databases	.25395	retrieval	.39651	mining	.42257
transactions	.15872	databases	.24174	users	.12466
dbms	.06802	users	.08319	retrieval	.06109
users	.04534	dbms	.03241	databases	.05730
heterogeneous	.03174	transactions	.03115	heterogeneous	.04007
Author	**Prob.**	**Author**	**Prob.**	**Author**	**Prob.**
Özsu_T	.09058	Fuhr_N	.09886	Sanderson_M	.06653
Chung_C	.04816	Croft_B	.06304	Younas_M	.05426
Perry_D	.04081	Li_J	.05036	Allan_J	.05129
Topic 92					
1990		**1994**		**2004**	
voronoi	.19580	segmentation	.14452	web	.39106
segmentation	.11918	diffuse	.08563	segmentation	.03875
texture	.11918	relaxation	.04478	regions	.02783
lighting	.05107	voronoi	.04170	classifiers	.02276
textures	.04256	interior	.03342	texture	.02137
Author	**Prob.**	**Author**	**Prob.**	**Author**	**Prob.**
Shields_M	.06129	Max_N	.03917	Antonacopoulos_A	.01654
Hanrahan_P	.01900	Nayar_S	.03006	Silva_A	.00865
Ware_C	.01702	Oren_M	.02931	Soatto_S	.00711

Table 2: Evolution of Sample Topics

5. REFERENCES

[1] G. Almpanidis, C. Kotropoulos, and I. Pitas. Combining text and link analysis for focused crawling-an application for vertical search engines. *Information Systems*, 32(6):886–908, 2007.

[2] L. Bolelli, S. Ertekin, and C. L. Giles. Clustering scientific literature using sparse citation graph analysis. In *PKDD'06*, pages 30–41, 2006.

[3] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths. Probabilistic author-topic models for information discovery. In *KDD '04*, pages 306–315, 2004.

[4] C. D. X. He, H. Zha and H. Simon. Web document clustering using hyperlink structures. *Computational Statistics and Data Analysis*, 41:19–45, 2002.