The Web of Topics: Discovering the Topology of Topic Evolution in a Corpus

Yookyung Jo
Department of Computer Science
Cornell University
Ithaca NY 14853
ykjo@cs.cornell.edu

John E. Hopcroft
Department of Computer Science
Cornell University
Ithaca NY 14853
jeh@cs.cornell.edu

Carl Lagoze
Computing and Information Science
Cornell University
Ithaca NY 14853
lagoze@cs.cornell.edu

ABSTRACT

In this paper we study how to discover the evolution of topics over time in a time-stamped document collection. Our approach is uniquely designed to capture the rich topology of topic evolution inherent in the corpus. Instead of characterizing the evolving topics at fixed time points, we conceptually define a topic as a quantized unit of evolutionary change in content and discover topics with the time of their appearance in the corpus. Discovered topics are then connected to form a topic evolution graph using a measure derived from the underlying document network. Our approach allows inhomogeneous distribution of topics over time and does not impose any topological restriction in topic evolution graphs. We evaluate our algorithm on the ACM corpus. The topic evolution graphs obtained from the ACM corpus provide an effective and concrete summary of the corpus with remarkably rich topology that are congruent to our background knowledge. In a finer resolution, the graphs reveal concrete information about the corpus that were previously unknown to us, suggesting the utility of our approach as a navigational tool for the corpus.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Abstracting methods*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.5.0 [**Information Interfaces and Presentation**]: General

General Terms

Algorithms

Keywords

topic evolution, topology, citation network, language model

1. INTRODUCTION

When navigating and seeking information in a digital document collection, the ability to identify topics with their time of appearance and see their evolution over time could be of significant help. Think of a scientific paper collection and a researcher who begins research in a specific area. She would want to quickly overview the area, determine how topics in the area have evolved, and locate important ideas and the papers that introduced them. Knowing a specific concept in a paper, she wants to find out whether there were previous papers that discussed the concept or the topic is new. As another example, a funding agency or people who administer a digital document collection might be interested in visualizing the landscape of topics in the collection to show the emergence and evolution of topics, bursts of topics, and the interaction among different topics that change over time.

These information seeking activities require the ability to identify topics with their time of appearance and to follow their evolution over time. In this paper, we describe our unique approach to providing the basic technologies to achieve such a goal. Our approach is applicable to a time-stamped document collection with an underlying document network. Such document collection format encompasses a wide range of digital text available over the Web recently. Examples are scientific paper collections, text collections with underlying social networks such as blogs and twitter, and in general the web documents with hyperlinks. In this paper, we demonstrate the utility of our approach by applying it to a scientific paper collection. We will use the word paper and document interchangeably.

Our approach emphasizes on discovering the topology of topic evolution inherent in a corpus. The topology inherent in the corpus carries surprisingly rich information about the evolution of topics as it is demonstrated in this paper (Figure 1, 2, 3, 4, and 5). We define a topic as a quantized unit of evolutionary change in content, and identify topics along with the time that they start to appear in the corpus. We do this by visiting each paper in the corpus chronologically and decide if the paper initiates a topic by requiring that it has a textual content that is not explained by previously discovered topics and that this textual content persists in a significant number of later papers. After obtaining topics by the chronological scan, we build graphs whose nodes are topics and whose edges reflect cross-citation relation between topics. Globally, this generates a map showing the landscape of topics over time as in Figure 1 obtained from the ACM corpus. The map shows a rich topology. For example, the population of topic nodes in network research grows fast in the later years without a significant body of ancestors before, while the compilers or graphics research areas exhibit steadier evolution over time. We can also find an individual topic evolution graph for a given seed topic as

WWW 2011, March 28–April 1, 2011, Hyderabad, India.
ACM 978-1-4503-0632-4/11/03.

shown in Figure 4 and 5. Such topic evolution graphs may contain multiple threads indicating that the seed topic has been influenced by multiple fields. The relationship between these threads may change over time as well.

The contribution of our paper is that *(1)* by defining a topic as a quantized unit of evolutionary change in content, we obtain a topic evolution graph without imposing topological restrictions on the graphs nor imposing restrictions on the time distribution of topic nodes. This is in contrast to a body of previous works [14] [18] [2] [1] [6] [5]. Such previous works either divide a corpus into time slots and find a fixed number of topics in each time slot or assume a predetermined topology for topic evolution such as a chain-like topology. *(2)* We obtain a large-scale topic evolution graph from ACM corpus. Also various local evolution graphs deduced from the topic relationships are explored. In general, previous works either report on topic evolution graphs with a chain-like topology or evolution graphs in small-scale. It seems that the topology of the evolution graph as complex and varied as ours was not reported previously. *(3)* Our approach uniquely incorporates the underlying document network such as the citation network into the topic evolution discovery.

In the rest of the paper, Section 2 and 3 explain our algorithm. Section 4 shows the experimental result with future work in Section 5. Section 6 discusses the related work and Section 7 concludes.

2. DETECTING TOPICS AS SIGNIFICANT CHANGES IN CONTENT EVOLUTION

We are interested in getting the summarized view of the corpus which shows the evolution of topics over time. A topic in a corpus is a semantically coherent content that is shared by a significant number of documents in the corpus. As time flows, a topic goes through evolutionary change. As the change accumulates, at some point in time, a document or a set of documents within the topic initiates a content that differs appreciably from the original content. Such content may die out or be shared by a significant number of later documents. In the latter case, we could quantize this significant change as a new topic. Yet, the new topic is born in the context of the original topic. By connecting the new topic with the previous topics that provided the context for the new topic, we can see the evolutionary process.

Taking this view, our approach captures the evolution of topics in a corpus by first identifying significant changes in the content evolution as topics and then connecting each topic with the previous topics that provided the context. Each topic is associated with the time the corresponding change is introduced in the corpus. As a result, we get a graph over topic nodes where nodes are associated with time.

Note that we usually need some regularizations to model the evolution of topics. For example, the common approaches taken by previous works are to quantize time into a number of time slots and connecting topics across time slots and/or to assume a chain-like topology for a topic evolution. In our case the regularization is that we quantize evolutionary change into topics whenever such change satisfies the requirement of novelty and significance. By novelty, we require that the new content differs appreciably from the original contents providing the context. By significance, we require that the new content is adopted by a sufficient number of later documents. The relative advantage of our modeling choice is that the topic nodes can be inhomogeneously placed in time and we do not impose restriction on the topology of the topic graphs, allowing the topology in the evolution inherent in the corpus to appear.

Technically, we use the mixture of word distributions [17] [7] [20] [14] to formulate the problem. A corpus is a collection of documents. A unigram vocabulary of a corpus is a set of all unigrams that appear in the corpus. Given a corpus, a word distribution θ is a multinomial distribution over the words in the unigram vocabulary V of the corpus. We denote the probability of producing a word w by the word distribution θ as $p(w|\theta)$. As θ is a multinomial distribution, the distribution satisfies the constraint $\sum_{w\in V} p(w|\theta) = 1$. The probability of a document d by a word distribution θ is defined as the probability of independently producing each occurrence of the unigrams in d by θ and is denoted as $p(d|\theta)$. In a document d, let $w_{d,i}$ denote the i^{th} occuring unigram in d and let N_d be the number of unigram occurrences in d. Then, $p(d|\theta) = \prod_{i=1}^{N_d} p(w_{d,i}|\theta)$. One particular word distribution we will repeatedly use is the one that maximizes the probability of the corpus. We call it the background model of a corpus and denote it by β. We can use Lagrangian multipliers to verify that $p(w|\beta) = \frac{c_w}{\sum_{w\in V} c_w}$ where c_w is the number of occurrences of a word w in the corpus and V is the unigram vocabulary of the corpus.

The probability of a document d by a mixture of word distributions $\theta_1, \theta_2, ..., \theta_k$ with the corresponding mixture coefficients $\pi_1, \pi_2, ..., \pi_k$ is defined as $\prod_{i=1}^{N_d}\left(\sum_{j=1}^{k} \pi_j p\left(w_{d,i}|\theta_j\right)\right)$ with the constraints $\sum_{j=1}^{k} \pi_j = 1$ and $\pi_j \geq 0$ for $j = 1, ..., k$. Each word in the document is produced by a probability that is a linear combination of the word distributions. In this paper, we will repeatedly use the type of mixture where one of the word distributions is a background model β with a fixed mixture coefficient b, while the mixture coefficients for the remaining word distributions $\theta_1, \theta_2, ..., \theta_k$ are determined by maximizing the probability of the document by the mixture. We denote the probability of a document d by this type of mixture as $p(d|\theta_1, ..., \theta_k; \beta, b)$. By the above definition, it is given as

$$p(d|\theta_1, ..., \theta_k; \beta, b) =$$
$$\prod_{i=1}^{N_d}\left(\left((1-b)\sum_{j=1}^{k} \pi_j p\left(w_{d,i}|\theta_j\right)\right) + bp\left(w_{d,i}|\beta\right)\right) \qquad (1)$$
$$\text{where} \quad \pi_1, ..., \pi_k =$$
$$\underset{\pi_1,...,\pi_k}{\text{argmax}} \prod_{i=1}^{N_d}\left(\left((1-b)\sum_{j=1}^{k} \pi_j p\left(w_{d,i}|\theta_j\right)\right) + bp\left(w_{d,i}|\beta\right)\right)$$

for $k \geq 1$. We also define the trivial case of $k = 0$ as $p(d|; \beta, b) = p(d|\beta)$.

In addition to the documents of a corpus, our approach requires the publication date of each document and a network over the documents where an edge between two documents indicates that they are semantically related with probability much higher than random chance. In this paper, our evalution is on a scientific paper collection, and we use the citation network over the papers. We index the documents in the corpus in chronological order as $d_1, d_2, ...,$ where if $i < j$ then the publication date of d_i is earlier than or equal to that of d_j.

To detect topics we visit the documents in the corpus chronologically. At each document, we test whether the document initiates a content that differs enough from the previously identified topics and is shared significantly by later documents. If so, we generate a new topic.

We use a tuple (d_s, θ_s, F) to characterize a topic τ in our topic discovery. Here $\tau.d_s$ is the paper that initiates the topic τ. We call it the start paper of τ. $\tau.\theta_s$ is a word distribution that represents the content of the start paper $\tau.d_s$ and $\tau.F$ is a set of papers that appreciably carry the content of the start paper. The exact definition of these terms will be given in the following topic definition.

In order to define a topic in terms of the previously defined topics, assume that we have chronologically scanned the documents from d_1 up to d_{t-1} and found k topics $\tau_0, ..., \tau_{k-1}$. We then examine the document d_t to decide whether it initiates a new topic.

Let the word distribution θ_t represent the content of d_t. We define θ_t by requiring that the mixture of θ_t and the background model β maximizes the probability of d_t.

$$\begin{aligned}\theta_t &= \underset{\theta_t}{\operatorname{argmax}}\, p(d_t|\theta_t;\beta,b) \\ &= \underset{\theta_t}{\operatorname{argmax}} \prod_{i=1}^{N_{d_t}} \left((1-b)\,p(w_{d_t,i}|\theta_t) + bp(w_{d_t,i}|\beta)\right) \quad (2)\end{aligned}$$

The background model β in the mixture absorbs the words that appear in d_t by random chance so that θ_t gets high support for the words that differentiate d_t from the rest of the corpus. The mixture coefficient b for β is fixed and is a parameter we set.

We then find the documents that carry the new content introduced by d_t. In order to measure how much a document f carries the content of d_t that differs from the previously identified topics $\tau_0, ..., \tau_{k-1}$, we use the document probability gain $g\left(f, \theta_t, \{\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s\}\right)$ defined as

$$\begin{aligned}&g\left(f, \theta_t, \{\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s\}\right) \\ &\qquad = \log \frac{p(f|\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s, \theta_t;\beta,b)}{p(f|\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s;\beta,b)}. \quad (3)\end{aligned}$$

The denominator $p(f|\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s;\beta,b)$ computes the probability of the document f using the mixture of θ_s's from the previous topics and the background model β while the numerator $p(f|\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s, \theta_t;\beta,b)$ additionally uses θ_t from d_t in its mixture to compute the probability of f. The document probability gain is non-negative as long as $\{\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s\}$ is not empty ($k > 0$) because the optimization domain of $p(f|\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s;\beta,b)$ is a subspace of that of $p(f|\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s, \theta_t;\beta,b)$. Note that in order for the document probability gain to be high, *(1)* θ_t should be different enough from θ_s's of the previous topics, otherwise $p(f|\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s;\beta,b)$ approaches $p(f|\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s, \theta_t;\beta,b)$, *(2)* the document f should contain the content that can be produced with good probability by θ_t but not by the θ_s's from the previous topics.

In finding the documents that appreciably carry θ_t, we use the documents that cite d_t as a candidate pool. We let F be the set of q documents whose document probability gain is the top q largest among the documents that cite d_t. Here q is a parameter to be set. We call the documents in F as top followers of d_t. In order to test whether d_t initiates a content that differs enough from the previously identified topics and is shared significantly by later documents, we check the conditions Eq.4 - 6. Here C_a, C_f and m are parameters.

$$\sum_{f \in F \cup \{d_t\}} g\left(f, \theta_t, \{\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s\}\right) \geq C_a \quad (4)$$

$$\sum_{f \in F} g\left(f, \theta_t, \{\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s\}\right) \geq C_f \quad (5)$$

$$\forall f \in F, g\left(f, \theta_t, \{\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s\}\right) \geq m \quad (6)$$

Eq.4 requires that the improvement in the log probability of d_t and the top followers due to θ_t over the θ_s's of the previous topics is lower-bounded by C_a. We separately require that such improvement in the log probability restricted to the top followers is lower-bounded by C_f in Eq.5, because the document probability gain $g\left(f, \theta_t, \{\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s\}\right)$ for a top follower $f \in F$ is a more reliable indication of whether θ_t initiates a new topic than the document probability gain for d_t, $g\left(d_t, \theta_t, \{\tau_0.\theta_s, ..., \tau_{k-1}.\theta_s\}\right)$. The reason is that the high probability words in θ_t computed by Eq.2 are either the words related to the topic of d_t or the noisy words unrelated to the topic whose background probability is low. While the document probability gain for d_t gets the contribution from both groups of words, the document probability gain for a top follower gets the contribution mostly from the topic-related words, hence, more reliable, because the noisy words in d_t is not likely to repeat in another document connected to d_t by a citation network. Eq.6 ensures that each top follower contributes in carrying the new content of d_t by setting a lower-bound on the document probability gain. If these conditions are met, we generate a new topic τ_k with $\tau_k.d_s = d_t$, $\tau_k.\theta_s = \theta_t$, $\tau_k.F = F$. Note that we use the "lookaheads" that are the later documents of d_t in determining whether to initiate a new topic at d_t so that we do not introduce non-significant or noisy topics. Before the algorithm scans the first document d_1, we initialize τ_0 with the background model β so that the algorithm starts with non-empty previous topics.

Optimization: The optimization problems of Eq.2 and Eq.1 with $k \geq 2$ are solved using the logarithm of the original optimization functions. Eq.2 is solved by Lagrangian multipliers. Inequality constraints ($\pi_j \geq 0$) are considered in applying Lagrangian multipliers, because unlike the maximum likelihood estimate of the background model β, the solution in Eq.2 only with the equality constraint may lie outside the inequality constraint boundaries. Eq.1 resembles the optimization problem arising in PLSI, but it is an easier problem because the word distributions are fixed. In particular, Eq.1 is a convex optimization for which efficient algorithms exist [4]. Our implementation iteratively moves the estimation point for $(\pi_1, ..., \pi_k)$ in the direction of the gradient projected onto the constrained domain. The next estimation is made by finding the point along the chosen direction with maximum function value using Newton's method. The running time is very reasonable for a large-scale corpus as seen in Section 4.

3. FINDING RELATIONSHIPS BETWEEN TOPICS

After discovering topics, we discover the relationships between topics in order to track the topic evolution. We discover the relationships between topics by first obtaining the member documents of each topic and then for each pair of topics examining the relation between their member documents.

For a topic τ, we include its start paper $\tau.d_s$ and the papers that cite $\tau.d_s$ as its member papers. In addition, the papers that are textually close to τ are included as its member papers. In order to textually represent the topic τ, $\tau.\theta_F$ is defined as the word distribution whose mixture with the background model maximizes the probability of the top followers $\tau.F$ and the start paper $\tau.d_s$ and is given by

$$\tau.\theta_F = \underset{\theta}{\operatorname{argmax}} \prod_{f \in \tau.F \cup \{\tau.d_s\}} p(f|\theta; \beta, b).$$

We use $\tau.\theta_F$ instead of $\tau.\theta_s$ because $\tau.\theta_F$ is less prone to noise as it is the word distribution based on the aggregation of papers. Also, Inequality 5 and 6 ensure that the papers in $\tau.F$ faithfully carry the content of $\tau.d_s$. To determine whether a paper d is textually close enough to qualify as a member paper of τ, we use the document probability gain $g\left(d, \tau.\theta_F, \{\}\right)$ (Eq.3) normalized by the number of words in the document d.

$$g\left(d, \tau.\theta_F, \{\}\right)/N_d = \frac{1}{N_d} \log \frac{p(d|\tau.\theta_F; \beta, b)}{p(d|; \beta, b)}$$

Here $g\left(d, \tau.\theta_F, \{\}\right)$ is negative for a paper d not related to the topic τ, while for a paper d related to τ, $g\left(d, \tau.\theta_F, \{\}\right)$ is positive. We include a paper d as a member paper of τ if $g\left(d, \tau.\theta_F, \{\}\right)/N_d \geq \gamma$ where γ is a positive parameter. Thus, if we denote the set of member papers of τ as $\tau.M$, $\tau.M$ is given as

$$\tau.M = \{\tau.d_s\} \cup \{d|d \text{ cites } \tau.d_s\} \cup \{d|g\left(d, \tau.\theta_F, \{\}\right)/N_d \geq \gamma\}.$$

Once we obtain the member papers for each topic, we find the relationships between topics. For a pair of topics, we use their cross citation count as their relationship data. The cross citation count between τ_i and τ_j is defined as $|\{(d_a, d_b)|d_a \text{ cites } d_b \text{ or } d_b \text{ cites } d_a, d_a \in \tau_i.M, d_b \in \tau_j.M\}|$. Using the cross citation count we derive a metric that represents the strength of the relationship between the pair of topics. By applying a threshold to the metric, we generate a graph of topics.

Let n_1 and n_2 be the number of member papers in topics τ_1 and τ_2 and let c be their cross citation count. Then there are $n_1 n_2$ pairs of papers d_i and d_j where $d_i \in \tau_1.M$ and $d_j \in \tau_2.M$. We say that there are $n_1 n_2$ cross pairs between τ_1 and τ_2. The metric we use to represent the strength of the relationship between the topics τ_1 and τ_2 is based on the following log likelihood ratio.

$$\log \frac{p(c \text{ cross citation count}|\tau_1 \text{ and } \tau_2 \text{ are related})}{p(c \text{ cross citation count}|\tau_1 \text{ and } \tau_2 \text{ are random})} \tag{7}$$

The numerator of the log is the probability to generate c cross citation count between τ_1 and τ_2 when the two topics are related, while the denominator is the corresponding probability when the two topics are randomly selected with respect to each other. In each case, we assume that the cross citation edges are generated by a binomial process. That is, there are $n_1 n_2$ cross pairs as trials and in each trial a citation edge is independently generated with a fixed Bernoulli probability. When the two topics are related, we use p_1 as the Bernoulli probability of the binomial process. When the two topics are random, we use p_0 as the Bernoulli probability.

The probabilities $p1$ and $p0$ are estimated as follows. The corpus has N papers and E citation links. A citation link is treated as an undirected edge. Let d be the average degree of a paper. By definition, $d = \frac{2E}{N}$. When the two topics τ_1 and τ_2 are randomly selected with respect to each other, it is reasonable to assume that if we pick a paper d_i from $\tau_1.M$ and d_j from $\tau_2.M$, the probability that the pair d_i and d_j has a citation edge is the probability that a random pair of papers in the corpus has a citation edge, which is given as $\frac{2E}{N(N-1)} = \frac{d}{N-1}$. We set $p_0 = \frac{d}{N-1}$. To estimate p_1 we make the following argument. A paper d_i has a number of neighbor papers in the citation network. However, the neighbor papers are not the exhaustive set of papers that are related to d_i. There are other papers related to d_i in the sense that d_i and another paper discuss the similar subject or an idea is transferred between them. We let R_i be the number of papers that are related to d_i. We also let R be the average of R_i's. By definition of R, the number of related pairs of papers in the corpus is $\frac{N \cdot R}{2}$. We assume that all E citation edges are contained within the related pairs of papers. The probability of a related pair of papers having a citation edge is then given as $\frac{2E}{N \cdot R} = \frac{d}{R}$. When the two topics τ_1 and τ_2 are related, we could assume that a cross pair of papers d_i and d_j from the two topics are related. Thus, we set $p_1 = \frac{d}{R}$. We use R as a parameter as we don't know its value. It is a parameter which we have an intuitive interpretation for.

We now compute the log likehood ratio (Eq.7). Among $n_1 n_2$ cross pairs of papers between the two topics τ_1 and τ_2, c trials generate a citation link while $n_1 n_2 - c$ trials do not. Translating this into a binomial process with $p1$ and $p0$ respectively,

$$\begin{aligned} &\log \frac{p(c \text{ cross citation count}|\tau_1 \text{ and } \tau_2 \text{ are related})}{p(c \text{ cross citation count}|\tau_1 \text{ and } \tau_2 \text{ are random})} \\ &= \log \frac{\binom{n_1 n_2}{c} p_1^c (1-p1)^{n_1 n_2 - c}}{\binom{n_1 n_2}{c} p_0^c (1-p0)^{n_1 n_2 - c}} \\ &= \left(\log \frac{p_1}{p_0} + \log \frac{1-p_0}{1-p_1}\right) \left(c - n_1 \cdot n_2 \cdot \frac{\log \frac{1-p_0}{1-p_1}}{\log \frac{p_1}{p_0} + \log \frac{1-p_0}{1-p_1}}\right) \end{aligned}$$

Removing $\left(\log \frac{p_1}{p_0} + \log \frac{1-p_0}{1-p_1}\right)$ as it is a constant over pairs of topics, yields

$$r\left(\tau_1, \tau_2; R\right) = c - n_1 \cdot n_2 \cdot \frac{\log \frac{1-p_0}{1-p_1}}{\log \frac{p_1}{p_0} + \log \frac{1-p_0}{1-p_1}} \tag{8}$$

which we call the relationship strength metric. The value $\frac{\log \frac{1-p_0}{1-p_1}}{\log \frac{p_1}{p_0} + \log \frac{1-p_0}{1-p_1}}$ is mathematically inbetween the probabilities p_1 and p_0. Thus, $r\left(\tau_1, \tau_2; R\right)$ can be interepreted as the cross citation count c discounted by the expected cross citation count $n_1 \cdot n_2 \frac{\log \frac{1-p_0}{1-p_1}}{\log \frac{p_1}{p_0} + \log \frac{1-p_0}{1-p_1}}$ when cross citation links are generated by a probability inbetween p_0 and p_1. Note that such discount grows proportional to the topic pair size $n_1 n_2$.

We generate a link between two topics τ_1 and τ_2 by imposing a threshold to the relationship strength metric

$$r\left(\tau_1, \tau_2; R\right) \geq \kappa$$

where κ is a parameter.

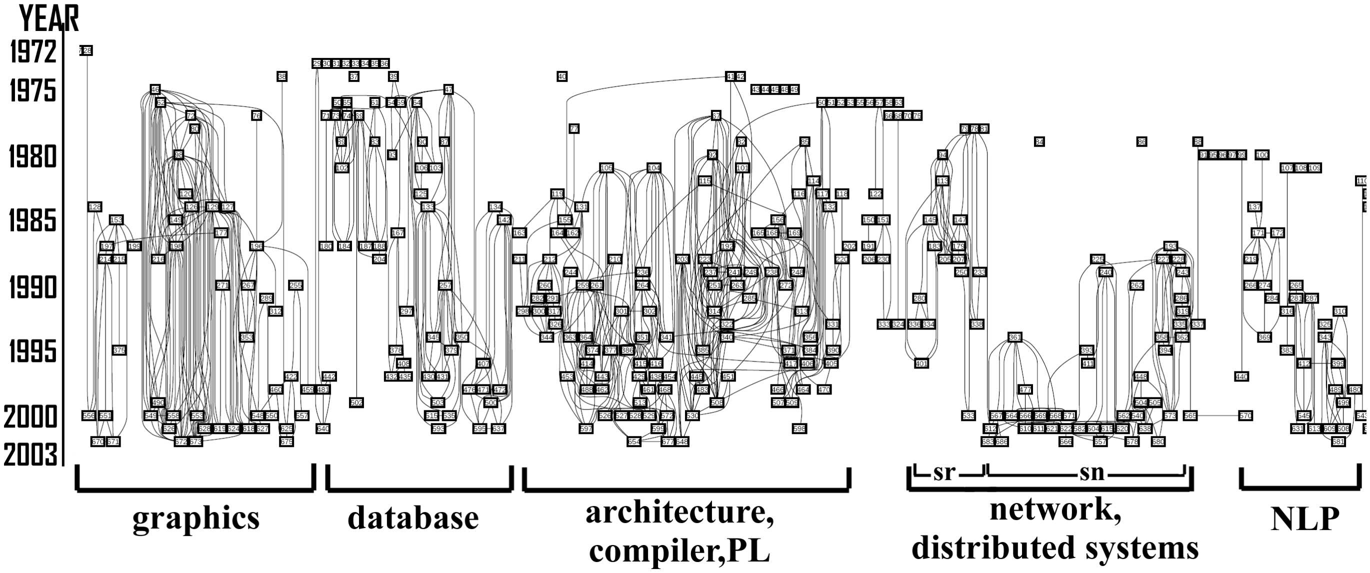

Figure 1: Snapshot of the global topic evolution map of the ACM corpus showing the five largest connected components

4. EVALUATION

4.1 Experimental Set up

We applied our algorithm to the collection of papers in the ACM corpus from the year 1952 to the year 2007. There are 129,544 papers in the collection with 291,122 citation links, for an average degree of 4.49. For each paper, we use its title and abstract as its text. The text is stemmed by Porter Stemmer [9]. The algorithm to detect topics was run with the parameters $\beta = 0.8$, $q = 10$, $C_a = 360.0$, $C_f = 210.0$ and $m = 7.0$. 743 topics were discovered. The running time was 1 hour and 17 seconds on a common desktop with intel E2160 processor.

The parameters β, q, C_a, C_f and m are used to determine the granularity of evolutionary change in detecting topics. We empirically determined their values by checking that the first few examples of the topic conditions Eq.4 - 6 perform the intended function. The cost of such adjustment was very small compared to the corpus size. The remaining parameters γ, R and κ are used to generate topic evolution graphs from the raw topic relationship data. Because the topic relationship data with cross citation counts have multitudinal information that are not entirely captured by a single graph representation, we vary the parameters γ, R and κ as control knobs to explore the topic evolutionary graphs.

4.2 Global Topic Evolution Map

After finding the topics, we obtained their member papers with the textual threshold parameter for member papers $\gamma = 0.7$. We now have the relationship between topics represented by the cross citation counts. To turn this relationship into a graph of topic nodes, we computed the relationship strength metric $r(\tau_i, \tau_j; R)$ (Eq.8) with $R = 200$ for each pair of topics τ_i and τ_j. The topic evolution graph in Figure 1 [1] was obtained by applying a threshold value $\kappa = 75$ to $r(\tau_1, \tau_2; R)$.

The small rectangles in the graph with numbers in them represent topics. The numbers are the topic ids and run chronologically from 1 to 743. An edge exists between a topic node pair if $r(\tau_1, \tau_2; R) \geq \kappa$. The Y axis in the left of the graph represents time. Though the corpus is from the year 1952 to 2007, the majority of topics are obtained in the time span from year 1972 to 2003. This is because in the early years of the ACM corpus the population of papers is sparse. Also we didn't discover many topics in the latest years of the ACM corpus because our method requires a significant number of follower papers that cite the start paper of a topic. We think that this problem can be remedied either by lowering the values of the parameters q, C_a, C_f and m, or by replacing the citation network with a document network that does not require as much time in generating edges. The graph is shown with time span from the year 1972 to 2003. All topic graphs shown in this paper are drawn by first requiring that each topic node should be placed at the publishing year of its start paper, and then letting the "dot" application in the graphviz tool [8] draw the graph. The "dot" draws an acyclic graph by minimizing edge crossing in a 2D layout [8]. [2]

In general, a lower threshold value κ brings in more edges to the graph for more relationship structures. But, such dense edges obscure the core structures of the graph in the 2D layout. Thus, in order to show the core structures in topic relationship, we used a high value of threshold $\kappa = 75$ in Figure 1 to have fewer edges. As a result, many other

[1] Figure 1 is a low-resolution snapshot of a large graph. To get a better view, you may want to magnify the pdf file.

[2] The topic nodes that are nearby in the 2D layout drawn by "dot" are usually well connected and thus closely related to each other. However, the nodes that are next to each other without any connection are not related. They are placed next to each other simply because of the default behavior of "dot".

meaningful structures are missing in Figure 1. We will explore some of those structures later in the paper.

In the graph, 235 nodes out of the 743 nodes are isolated. Figure 1 is a partial snapshot of the graph focused on the region with the large connected components. By inspecting the abstract of the start paper and the tf.idf summary of the top follower papers of each topic, we manually labeled the five largest connected components as shown at the bottom of the graph. These connected components are (1) graphics, (2) database, (3) architecture, compiler and programming lanauge, (4) network and distributed system, and (5) natural language processing.

Note the difference and variety in the topology of these connected components. For example, the connected component for network and distributed systems has two weakly connected subgraphs. The left subgraph labeled sr contains topic nodes for "reliable secure protocols at the presence of faulty processes" and "cryptography" in distributed systems. The right subgraph labeled sn contains topic nodes for network research. Subgraph sn starts with the topic nodes around mid 80s and grows into a very popolated area with many topic nodes later. There are some earlier topic nodes in network research that do not yet appear in the subgraph sn due to the high threshold κ for edges, but there is an overall trend of increasing population over time. On the other hand, the connected components for graphics, database, and architecture and PL exhibit steadier distribution of topic nodes over time.

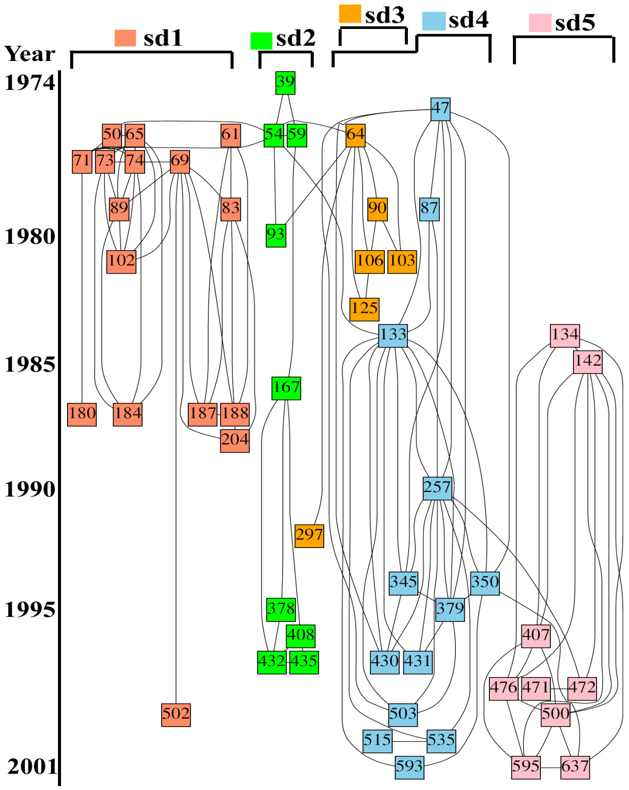

Figure 2: The topic evolution graph for Database

Table 1: Textual information of selected topics in Database

id	title of the start paper
50	The entity-relationship model—toward a unified view of data
65	Synthesizing third normal form relations from functional dependencies
74	Multivalued dependencies and a new normal form for relational databases
102	Can we use the universal instance assumption without using nulls?
184	A new normal form for nested relations
61	The Semantics of Predicate Logic as a Programming Language
188	Decidability and expressiveness aspects of logic queries
502	Query rewriting for semistructured data
39	The UNIX time-sharing system
54	System R. relational approach to database management
59	Decomposition—a strategy for query processing
93	Introduction to a system for distributed databases (SDD-1)
167	Efficiently updating materialized views
432	Maintenance of data cubes and summary tables in a warehouse
64	The notions of consistency and predicate locks in a database system
90	Weighted voting for replicated data
103	Nonblocking commit protocols
125	Multilevel atomicity—a new correctness criterion for database concurrency control
297	ARIES. a transaction recovery method supporting fine-granularity locking and partial rollbacks using write-ahead logging
47	Multidimensional binary search trees used for associative searching
133	R-trees. a dynamic index structure for spatial searching
257	The R*-tree: an efficient and robust access method for points and rectangles
350	Fast subsequence matching in time-series databases
379	Nearest neighbor queries
535	Indexing moving points
593	Locally adaptive dimensionality reduction for indexing large time series databases
134	Accurate estimation of the number of tuples satisfying a condition
407	Improved histograms for selectivity estimation of range predicates
476	Wavelet-based histograms for selectivity estimation
595	Space-efficient online computation of quantile summaries
637	Continuous queries over data streams

We now zoom in to one of the large connected components and see how the topology in a finer resolution reflects the topic evolution trend in the area. Figure 2 shows the connected component for database in detail. Also, Table 1 shows the titles of the start paper of selected topics in Figure 2 to give a more concrete idea on the textual content of the topics. We explain the graph in Figure 2 by dividing it into 5 subgraphs as suggested by the visualized connectivity. In reading the description below, refer to Table 1 for more detail.

Subgraph sd1 on the left is on the theoretical foundations of database systems such as data models and relational algebra. It includes "entity-relationship model in 1976"(topic id 50), "discussion on third normal form"(topic id 65), and "4th normal form"(topic id 74). The discussion on data dependency continues through topics 89, 102, and 184, with topic id 184 on "a new normal form for nested relation". The thread of topics 61, 83, 187, 188, and 204 in Subgraph sd1 is relatively separated from the topics covered above. This thread shows that logic programs were actively discussed in database design. For example, topic id 83 discusses "logic program based queries over relational database". Note that most of the topic nodes in subgraph sd1 reside in 1970's and 80's. But there is a topic node 502 in 1999. The topic node 502 is about "query rewriting for semistructured data", which seems to reflect the evolution of topics. Subgraph sd2 is on building database systems. Topic nodes 54 and 59 in 1976 are on "System R" and "INGRES" respectively. Much later in time, topic nodes 378, 408, 432, and 435 in 1995 to 1997 are on "data warehouse" discussing view maintenance and data cubes, OLAP etc.

Subgraph sd3 is on concurrency control in database systems. For example, it contains “consistency and locks” (topic id 64), “nonblocking commit protocols” (topic id 103), “multilevel atomicity” (topic id 125), and “transaction recovery with fine granularity locking and partial rollbacks” (topic id 297). Note that the topic node 93 that connects subgraphs sd2 and sd3 is on a distributed database system(SDD-1), combining the system building aspect of Subgraph sd2 and the distributed system nature of Subgraph sd3. Subgraph sd4 is on data structure of data storage for efficient search. The subgraph demonstrates content evolution over time with “multidimensional binary search tree” (topic id 47) in 1975, “R-tree” in 1984 (topic id 133), “R*-tree” in 1990 (topic id 257) for spatial search, and more recently, “nearest neighbor queries” (topic id 379) and “distance browsing” (topic id 503) in spatial databases, discussion in time-series databases (topic id 350, 593), and discussion in moving object databases (topic id 515 and 535). Subgraph sd5 is on efficient query processing with tuple size estimation, histograms, etc (topic id 134, 142, 407, and 476). More recent topics in Subgraph sd5 are on query estimation for online datastream (topic id 595 and 637).

Overall, the observation on the connected component for databases shows that
(1) the content cohrerence of evolving topics is reflected in the connectivity pattern of the graph,
(2) dynamic change in topic node population along each subgraph over time seems to reflect reality,
(3) The content evolution along the topic thread is visible.

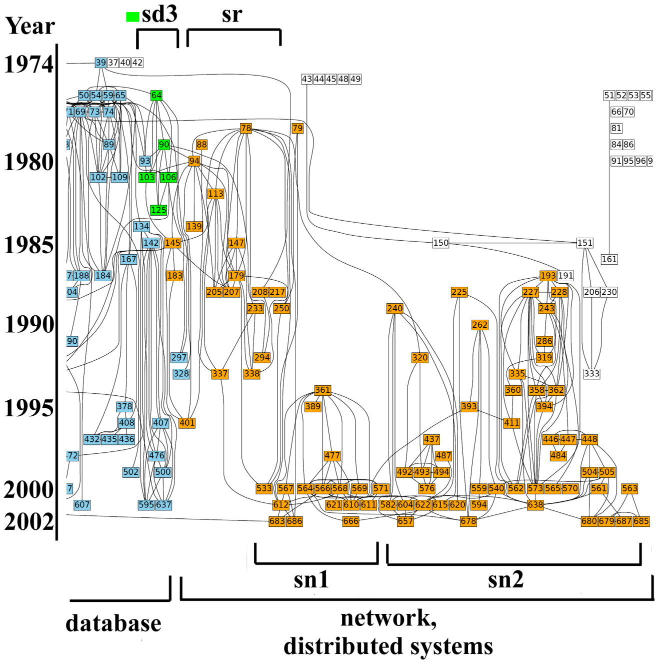

Figure 3: Partial snapshot of the global topic evolution graph with more edges showing the merge of connected components

As we bring in more edges to the global topic evolution map (Figure 1) by lowering the threshold κ, the existing connected components absorb more isolated nodes. Also, the connected components get connected to each other. For example, when we lower κ to 40 to bring in more edges, we observe that the connected component of databases becomes connected to the connected component of distributed systems and network. Figure 3 shows the snapshot of such a merge at $\kappa = 40$. Figure 3 contains network, distributed systems, and the subset of topic nodes from databases, from right to left. Comparing with Figure 1, one can see that the subgraph representing network research is getting richer in Figure 3. Its subgraph sn2, on the right starting in year 1987, exhibits the evolution of topics in networks from TCP layers to the more recent BGP routings. Its subgraph sn1 on the left starting more recently in year 1994 is mostly on mobile, ad-hoc networks. The connection between databases and distributed systems is mainly made by the nodes in Subgraph sd3 in databases (Figure 2) being connected to the nodes in distributed systems, which makes sense because Subgraph sd3 in databases is about concurrency control in database systems. In fact, at $\kappa = 40$, the connected components for architecture and PL and for databases and for network and distributed systems are all connected, while the connected component for graphics is still isolated.

Figure 1 mainly shows the large connected components of the topic graph with $\kappa = 75$. The nodes not shown in Figure 1 are isolated nodes and nodes forming small structures. Examples of the small structures not shown in Figure 1 are the thread of topics in association rule mining, frequent item mining, and the thread for topics in web search. When we bring in more edges, these small structures grow to show richer topic evolution patterns. We will see an example in the next subsection. Also, when we bring in more edges, the isolated nodes either are absorbed into the existing structures or they form new connected components. Examples of such new connected components we have seen are the topic thread for computer science education and the topic thread for CAD.

4.3 Topic Evolution Graphs for Individual Topics

We now investigate how to find the topic evolution graphs for individual topics. All topic evolution graphs in this subsection are obtained by starting from a single topic node as a seed and discovering the earlier topic nodes from which the seed node has possibly evolved. There are nontrivial technical challenges involved. A single set of parameters (γ, R, κ) for determining topic edges is not universally adequate to reveal the evolution structure for individual topics. For example, the parameters (γ, R, κ) used in Figure 1 is effective in revealing the evolution structure of dense areas but it does not discover the structures for sparse area. On the otherhand, the parameter values adequate to discover the structures of sparse areas may leave the dense area too dense to decipher the structure. In this subsection, rather than focusing on finding the single best parameter values, we explore the parameter space and present multiple examples of graphs obtained with varying parameter values. A simple breadth-first search is quite effective in discovering the topic evolution graphs for a seed topic (Figure 4 and Figure 5(a)). But we also present a case that needs smarter graph expansion strategy (Figure 5(b1)-(b3)). Solving these technical challenges and finding a unified and automated way to discover the individual evolution graphs is left for future work. Nonetheless, the examples covered here demonstrate that the topic graphs obtained by the relatively simple methods show informative evolutionary structure, carrying concrete information about the corpus that are sometimes previously unknown to us.

To discover a topic evolution graph from a seed topic, we apply a breadth-first search starting from the seed node but only following the edges that lead to topic nodes earlier in time. In order to follow the edges in one direction in time, we treat the edges between topic nodes as directed edges. For an edge that connects topics i and j, if the time of topic i is earlier than that of topic j, the edge is a directed edge from topic j to the earlier topic i. In this subsection, we used a lower value for the textual similarity threshold parameter $\gamma = 0.2$ than the value used in Figure 1, in order to have more member papers in topics so that we do not suffer from sparseness of cross citations. The textual information for topics is obtained by manual inspection of the start paper and tf.idf summary of the top follower papers.

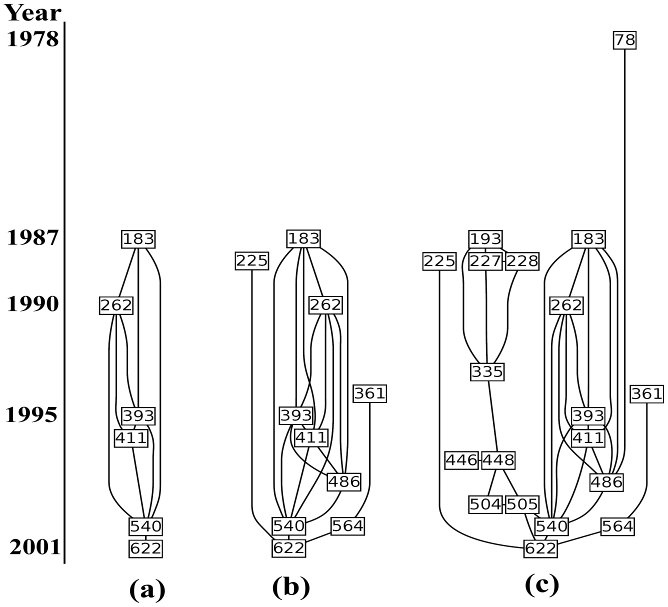

Figure 4: The topic evolution graphs for Topic 622

Figure 4(a) shows the topic evolution graph for topic 622 obtained by breadth-first search with topic edge parameters $R = 500, \kappa = 500$. Figure 4(b) and (c) are the graphs similarly obtained but with lower values of κ to bring more topic nodes into the graph. The values 200 and 140 are used for κ in Figure 4(b) and (c) respectively. Topic 622 is about "peer to peer system with distributed hash table". Its start paper is the paper that introduces Chord. Chord is a peer-to-peer system with distributed hash table that scales logarithmically.

With high link threshold, Figure 4(a) shows that we discovered 5 earlier nodes for topic 622. These earlier nodes are well connected to each other forming a single thread. As we lower the link threshold κ gradually, we bring in more nodes forming new threads (Figure 4(b) and (c)). We first take a look at the thread consisting of topic nodes 183, 262, 393, 411, 486, and 540 in Figure 4. This is the thread that survived in Figure 4(a). The thread is about multicast, which makes sense because "peer-to-peer system" can be thought of as a decentralized multicast protocol in the application layer.

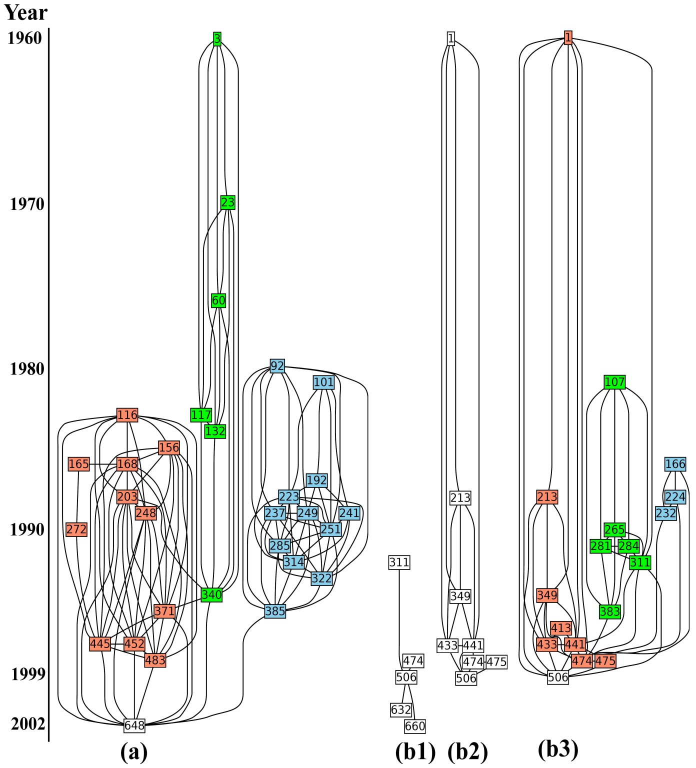

Figure 5: (a) The topic evolution graph for Topic 648, (b1)-(b3) The topic evolution graphs for Topic 506

The thread has a directed change with the evolutionary flavor in that as time goes by, multicast is studied from lower network layer to higher layers, from extended LAN to IP layer to application layer. Chronologically, Topic 183 discusses "reliable multicast in the presence of failures, message ordering, and scalability". The start paper of Topic 262 discusses "multicast routing in datagram internetworks and extended LAN". Topic 393 covers "IP multicast protocol (SRM)". Topic 411 talks about "receiver-driven multicast". Topic 486 is about "using key graphs for secure group communication scalable(logarithmic) for dynamic group change". Topic 540 introduces "End system multicast" arguing for "the need to provide multicast not in the IP layer but in the higher layer of the network". Some of the topics covered in other weaker threads in Figure 4(c) are "domain name system"(Topic 225), "local service protocol in ad-hoc networks"(Topic 564 and 225), "network topology and the power-law internet topology"(Topic 505).

Figure 5(a) shows the topic evolution graph for Topic 648 with parameters $R = 300, \kappa = 400$. The start paper of Topic 648 is about "making C programs type-safe by pointer analysis guaranteeing memory safety". The three threads in Figure 5(a) are, from left to right, the thread on "type", the thread on "storage reclamation and garbage collection", and the thread on "pointer analysis". Note that the middle thread reaches far earlier years of the ACM corpus compared to the other threads. In such early years, garbage collection is discussed in the context of the list structure of LISP (Topic 3, 23, and 60).

Figure 5(b1)-(b3) shows the various topic evolution graphs for Topic 506. Topic 506 is about link-based web search with its start paper title being "Authoritative sources in a hyperlinked environment". Topic 506 is located in the region where edges are relatively sparse. Figure 5(b1) shows the snapshot of the small connected component that contains topic 506, with the same parameters used for the global topic evolution map in Figure 1. In order to obtain the topic evolution graph for 506 such as in Figure 5(b2) or (b3), we applied a threshold to the bare cross citation count c instead of to the relationship strength metric $r(\tau_1, \tau_2; R) = c - n_1 \cdot n_2 \cdot \frac{\log \frac{1-p_0}{1-p_1}}{\log \frac{p_1}{p_0} + \log \frac{1-p_0}{1-p_1}}$ to determine whether a pair of topics has an edge between them. The reason is that the metric requires that the cross citation count c be greater than the second term in order for the metric to be positive. While the second term is effective in discounting the unfair advantage of a topic pair with large size having more cross citation count, the requirement is too stringent for Topic 506 and its neighbors with sparse edges. As a result, Figure 5(b2) shows the topic evolution graph for 506 discovered by drawing more member papers to each topic ($\gamma = 0.2$) and applying a threshold 400 to cross citation counts. Compared to Figure 5(b1), more earlier topic nodes are discovered in Figure 5(b2). These nodes are well connected to each other. Topic nodes 1, 213, and 349 are information retrieval topics and topic nodes 433, 441, 474, and 475 are topics in web search.

When we lowered the cross citation count threshold from 400 to 200 in order to enrich the existing thread and to find other relevant threads, we encountered a problem. With the lower link threshold, some topic nodes bring in a lot of less relevant nodes into the thread. For example, the breadth-first search found a topic node on web-caching which in turn brings a lot of new nodes in caching and memory. To prevent this problem, we employed a simple branch pruning strategy that prunes a branch consisting of the nodes expanded from a single node if this branch has very little connection to the rest of the graph. The details of the pruning strategy are omitted.

Figure 5(b3) shows the result of the experiment after the second step of the breadth-first search. The existing thread has the additional topic node 413 which is about compression of inverted index for fast information retrieval. The middle thread consists of topic nodes close to NLP that are somewhat relevant to information retrieval such as document clustering (Topic 311), lexical segmentation (Topic 284), machine translation (Topic 281). The rightmost thread contains the discussion in hypertext system in the late 80's such as hypertext system implementation (Topic 166 and 224) and formal defintion of hypertext system using petri-net (Topic 232).

5. FUTURE WORK

The technical challenges we would like to address in the future are mining the relationships between topics, topic evolution thread discovery and textual mining on evolution threads. Another promising direction for future work is to build a navigational application based on our algorithm. When we navigated the topic evolution graphs obtained, we often discovered from the graphs concrete information that was either unknown to us or only vaguely known. This experience suggests the utility of our topic evolution graphs as a navigational aid as well as an effective global summary for the corpus.

6. RELATED WORK

Recently, various approaches [15] [14] [18] [2] [19] [1] [6] [5] have been proposed to model topic evolution in a time-stamped corpus. These approaches use variants [19] [2] [1] [6] of LDA [3] or variants [14] [5] of PLSI [7] or clustering on feature space of tf idf to model text. The works on topic evolution also differ in their modeling choice of how to accomodate the transition of topic content over time and the topological change in topic evolution. [2] divides the documents in a corpus into a number of time slots and applies LDA in each time slot while letting the hyperparameters change over time through Gaussian noise. [19] extends LDA by using per-topic Beta distribution to generate the time-stamp of each document. The discovered topics are more narrowly distributed in time showing the dynamic change in their population over time. [18] generates clusters at discrete time points with aging that discounts the contribution of old data points. It then uses the overlap of clusters from different time points to determine the transition or emergence or disppearance of clusters over time. [14] divides the documents by time slot and applies a variant of PLSI to extract the topics. It then uses KL-divergence based similarity between topic models to derive the topic evolution graphs. [1] processes a batch of documents at a time and applies LDA while using the topic models learned from previous time slices as a prior for the current model. [5] successively applies PLSI to a batch of documents at a time. It folds in new words using Bayesian inversion of probability as well as using traditional folding in of new documents to evolve the topic model. [6] applies LDA-based model to the documents in each time slot. In finding topics for a time slot, their model considers the previous documents cited by the documents in the time slot as well with a mechanism to give more weight to a relevant document.

Our work differs from the previous work in that our approach is designed with more emphasis on revealing the topology of topic evolution inherent in the corpus and it leverages the network underlying the corpus in a unique way. We discover topics without dividing the corpus into time slots by conceptually defining a topic as a quantized unit of significant change in topic evolution. This allows topics to be discovered with non-homogeneous distribution over time that are inherent in the corpus as shown in Figure 1. Topics are then connected by the relationship derived from the citation network to form a topic evolution graph. In contrast, previous works [6] [2] [18] [14] [5] [1] divide the corpus into time slots to discover topics, and [2] [5] [1] restrict the topology of a topic evolution graph by letting each topic thread form a chain. There are previous works that discover topics without imposing time restriction [19] [15], or that do not impose much topological restriction in connecting topics [6] [18] [14]. However, these works still do not demonstrate the rich topology of topic evolution as shown in the figures in our evaluation.

Our work is built on the premise that the words relevant to a topic are distributed over documents such that the distribution is correlated with the underlying document network such as a citation network. Specifically, in our topic discovery methodology, in order to test if a multinomial word distribution derived from a document constitutes a new topic, the following heuristic is used. We check that the distribution is exclusively correlated to the document network by requiring it to be significantly present in other documents that are network neighbors of the given document while suppressing the nondiscriminative words using the background model. Such correlation is previously used in [10] to discover topic terms. The strategy of using the background model in the mixture model to absorb the nondiscriminitive words is employed in a number of previous works [20] [14]. In order to measure the contribution of a word distribution on a document over the existing word distributions, we used a log odd ratio test(Eq.3). We inherit such form of log odd ratio test from [17].

Many available text data have a network associated with them. Examples are citation networks or various social networks. The importance of utilizing the network associated with text data is recently recognized in topic detection [16] [12] [13], in topic evolution detection [6], and in social network mining [11] etc. [16] incorporates the citation link generation into the generative process of LDA. [13] uses a potential that encourages the neighboring documents to be similar in their topic distribution. [12] uses citation statistics to derive various relationship measures among topics such as topical diffusion, diversity and tranfer. [11] solves the problem of tracking the evolution of a single event using a model that utilizes the similarity propagated through the network and time.

7. CONCLUSION

In this paper, we propose a new approach to discover the evolution of topics over time in a time-stamped document collection. Our approach emphasizes on capturing the topology of topic evolution that is inherent in the corpus. The evaluation of our algorithm on the ACM corpus demonstrates that the topology of the topic evolution discovered by our algorithm is very rich and carries concrete information on how the corpus has evolved over time. Our result suggests a wealth of interesting future work including the technical challenges we faced during the evaluation and the possibility of building an application for summarizing and navigating a research paper collection.

8. ACKNOWLEDGMENTS

The work is supported by NSF Grant DUE-0840744 *NSDL Technical Network Services: A Cyberinfrastructure Platform for STEM Education* and Air Force Grant F56-8350. We thank ACM for providing the metadata of the ACM corpus. We thank anonymous reviewers for their comments.

9. REFERENCES

[1] L. AlSumait, D. Barbara, and C. Domeniconi. On-line LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *ICDM*, 2008.

[2] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *23rd International Conference on Machine Learning*, 2006.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, (3):993–1022, 2003.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

[5] A. Gohr and A. Hinneburg. Topic evolution in a stream of documents. In *SDM*, 2009.

[6] Q. He, B. Chen, J. Pei, B. Qiu, P. Mitra, and C. L. Giles. Detecting topic evolution in scientific literature: How can citations help? In *CIKM*, 2009.

[7] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, 1999.

[8] http://graphviz.org.

[9] http://tartarus.org/∼martin/PorterStemmer/.

[10] Y. Jo, C. Lagoze, and C. L. Giles. Detecting research topics via the correlation between graphs and texts. In *SIGKDD*, 2007.

[11] C. X. Lin, B. Zhao, Q. Mei, and J. Han. Pet: A statistical model for popular events tracking in social communities. In *SIGKDD*, 2010.

[12] G. S. Mann, D. Mimno, and A. McCallum. Bibliometric impact measures leveraging topic analysis. In *JCDL*, 2006.

[13] Q. Mei, D. Cai, D. Zhang, and C. Zhai. Topic modeling with network regularization. In *WWW*, 2008.

[14] Q. Mei and C. Zhai. Discovering evolutionary theme patterns from text - an exploration of temporal text mining. In *SIGKDD*, 2005.

[15] S. Morinaga and K. Yamanishi. Tracking dynamics of topic trends using a finite mixture model. In *SIGKDD*, 2004.

[16] R. Nallapati, A. Ahmed, E. P. Xing, and W. W. Cohen. Joint latent topic models for text and citations. In *SIGKDD*, 2008.

[17] B. Shaparenko and T. Joachims. Information genealogy: Uncovering the flow of ideas in non-hyperlinked document databases. In *SIGKDD*, July 2007.

[18] M. Spiloipoulou, I. Ntoutsi, Y. Theodoridis, and R. Schult. Monic - modeling and monitoring cluster transitions. In *SIGKDD*, 2006.

[19] X. Wang and A. McCallum. Topics over time: A non-Markov continuous-time model of topical trends. In *SIGKDD*, 2006.

[20] C. Zhai, A. Velivelli, and B. Yu. A cross-collection mixture model for comparative text mining. In *SIGKDD*, 2004.