Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 Keller Hall
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 15-011

Accounting for Language Changes over Time in Document Similarity Search

Sara Morsy, George Karypis

July 7, 2015

Accounting for Language Changes over Time in Document Similarity Search

Sara Morsy
University of Minnesota
morsy@cs.umn.edu

George Karypis
University of Minnesota
karypis@cs.umn.edu

ABSTRACT

Given a query document, ranking the documents in a collection based on how similar they are to the query is an essential task with extensive applications. For collections that contain documents whose creation dates span several decades, this task is further complicated by the fact that the language changes over time. For example, many terms add or lose one or more senses to meet people's evolving needs. To address this problem, we present methods that take advantage of two types of information in order to account for the language change. The first is the citation network that often exists within the collection, which can be used to link related documents with significantly different creation dates (and hence different language use). The second is the changes in the usage frequency of terms that occur over time, which can indicate changes in their senses and uses. These methods utilize the above information while estimating the representation of both documents and terms within the context of non-probabilistic static and dynamic topic models. Our experiments on two real-world datasets that span more than 40 years show that our proposed methods improve the retrieval performance of existing models and that these improvements are statistically significant.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Retrieval Models*; H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Linguistic processing*

General Terms

Experimentation, Performance

Keywords

Similarity search, Topic modeling, Longitudinal document collections, Language change, Citation network, Terms usage frequency changes, Regularization

1. INTRODUCTION

Searching for similar documents to a given query document is an important task with extensive applications. For example, during patent issuing and prosecution, examiners are responsible for retrieving all previous prior art that are most relevant to the application in order to determine the novelty claimed by it (this task is called **Prior Art Candidate Search**). Another example is in scientific research, where researchers are interested in retrieving articles that are related to some recently published articles. These types of collections (patents and scientific articles) span several decades and the documents that are similar or related to the queries may have been written a long time ago. We will refer to these types of collections that span a long time period as **longitudinal document collections**.

Many linguistic and computational studies have shown that the language changes over time [20, 26, 30, 25, 17]. These changes can take one of three forms. The first, called **word sense evolution**, is the form of change where an existing term adds (or removes) one or more senses, e.g., the term "mouse" gained a new sense when graphical user interfaces were introduced to indicate a computer input device. The second, called **term-to-term evolution** or (synonyms over time), is the form in which a new term is created that has the same meaning as an older term, e.g., "mp3 player" has become a recent synonym for "walkman". The new term can co-exist with the older one or eclipse it. The third form deals with **emergence of new terms**, in which new terms are introduced to describe newly created concepts. For example, "Google" and "Facebook" were first introduced after their launch on the Internet.

Due to these time-induced language changes, the task of finding similar documents in longitudinal document collections becomes harder. Approaches based on dimensionality reduction and topic modeling [6, 13, 5, 28] can indirectly account for some of these changes by estimating their model parameters using the entire collection. However, researchers have recognized that such approaches are suboptimal, and for this reason, methods that explicitly model the changes in the language have been investigated. One line of research focused on term-to-term evolution (or more specifically, named entity evolution, since the query used is a named entity) by reformulating the user's query given in today's language in order to translate it to the user-specified date's language so that the translated query is then used to retrieve old documents relevant to the query [16, 8, 9, 10, 7]. Another line of research developed *dynamic* topic modeling approaches to handle topic evolution over time [2, 4, 27, 29]. These approaches estimate a different term-topic matrix for each time span leading to time-specific distributions for the words over the latent topics. As a result, they can explicitly model language change. Even though these models were not directly evaluated in the context of document similarity search task, they have shown to be qualitatively better in modeling documents, words and topics for longitudinal document collections.

In this paper, we develop methods for document similarity search that explicitly account for language changes that occur in longitudinal document collections. Our methods improve upon the current non-probabilistic static and dynamic topic modeling approaches by leveraging two types of information: the citation network in the collection (when available) and changes in the terms' usage frequency over time. The citation network allows our methods to extract information from related document pairs (i.e., linked documents) whose language may have changed due to the passage of time. At the same time, the changes in the usage frequency of terms allow our methods to explicitly account for the changes in the words' uses and senses. We use these two types of information while estimating static and dynamic topic models. Specifically, the citation network is used to regularize the latent representation of documents in both static and dynamic topic models in order to increase the similarity between linked documents in the latent space. The changes in the usage frequency of the terms are used to compute *term-time-specific* transition regularization weights in dynamic topic models, which enable them to model non-smooth transitions that are indicative of major changes in the terms' usage.

Our main contributions are: (i) We study the effect of link regularization on retrieving documents similar to a query document when the query and set of searched documents have large differences in their publication dates; and (ii) We present a novel transition regularization technique for terms in dynamic topic models that captures the changes in their usage over time.

We evaluated our methods on the task of document similarity search, where the user is interested in retrieving similar documents in a specific time span for a query document from the most recent time period. We used a subset of the US utility patents as well as the Association of Computational Linguistics Anthology dataset that span more than 40 years. Our results show that: (i) adding link regularization improves the retrieval performance of both static and dynamic topic models, with larger improvements in early documents' retrieval; and (ii) having term-time-specific transition regularization weights is better than having the same weights in link regularized dynamic topic models.

2. RELATED WORK

2.1 Language Change over Time

There have been a lot of linguistic and computational studies that showed how and why the language changes over time [17, 20, 26, 30]. Language changes to meet people's evolving needs, which continuously change over time. These changes can occur to existing words by gaining and/or losing one or more senses. New words also tend to appear when creating new technology, medicine, or other concepts. Some computational studies were done to analyze how to capture these changes. Kulkarni *et al.* [17] and Tahmasebi *et al.* [26] showed that peaks in terms' frequencies could correspond to invention of new technology, events, or change of meaning. This signal is only reliable when the topic popularity over time does not change. For example, searching for the terms "Hurricane" and "Sandy" on Google Trends[1] shows a peak in their frequencies in October 2012, which occurred due to having a storm called "Hurricane Sandy". This frequency peak was a signal of changing the sense of "Sandy" only (no change occurred to the sense of "Hurricane").

For this reason, Kulkarni *et al.* [17] also provided two other ways to capture language change for existing words, which are based on syntactic and distributional changes. When a term gains a new sense, it could gain a new Part-Of-Speech (POS), e.g., the word "apple" used to have the POS "Noun" only until Apple technical company was established in 1971, where it gained a new POS ("Proper Noun"). When there is neither a change in the term frequency or POS distribution over time, distributional-based changes (based on the distributional hypothesis that states that words that appear in the same contexts tend to have similar meanings [12]) can also signal a change in the term's sense. These changes can be learned by mapping words to different semantic vector spaces over time, and then tracking changes in the words that appear close to the words of interest in these spaces.

2.2 Document Modeling

Early document modeling approaches include: Vector Space Models (VSM) [23], probabilistic models, e.g., Okapi BM25 [22], and language models, e.g., the query likelihood model [19]. These models are based on term matching, where they represent documents as bags-of-words, apply some term weighting function (as in the case of VSM and probabilistic models) or build a language model for each document (as in language models), and then apply a ranking function to rank documents based on their relevance to queries.

To learn the latent semantics in the collection, researchers have proposed different topic modeling approaches, which fall into non-probabilistic (matrix factorization) approaches, such as Latent Semantic Indexing (LSI) [6] and Regularized LSI (RLSI) [28], and probabilistic approaches, such as Probabilistic LSI (PLSI) [13] and Latent Dirichlet Allocation (LDA) [5]. These topic models represent documents and terms in low-dimensional spaces. In probabilistic topic models, a document is represented as a weighted mixture of latent topics, and a latent topic is represented as a weighted vector of terms. For example, LDA [5] generates a document by choosing a distribution over topics, then for each term in the document, a topic is chosen according to the topic distribution and a term is drawn according to the term distribution in that topic. In non-probabilistic topic models, each document and term is represented as a point in a low-dimensional latent space. For example, RLSI [28] formalizes the problem as a minimization of a quadratic loss function to factorize the original document-term matrix into low-dimensional document-topic and term-topic matrices, regularized by $L1$ and/or $L2$ norms. The authors showed that RLSI and LDA perform similarly in relevance ranking. One major limitation of these approaches is that they model documents and terms in static spaces, i.e., assuming no language change over time, that fail to represent the evolution of language use. This makes these approaches suboptimal for document similarity search for longitudinal document collections when the language of the query and its relevant documents are different due to changes in their language use.

2.3 Dealing with Language Change in IR

Recently, researchers have investigated some approaches to handle language change in Information Retrieval (IR). Some work has been done on term-to-term evolution to re-

[1] `http://www.google.com/trends`

formulate a named entity query (a named entity is a name of a person, place, or organization) given at some reference time R to translate it to the language used at some target time T [3, 15, 14, 24]. For instance, Berberich *et al.* [3] developed a probabilistic measure of across-time semantic similarity between term u at time R and term v at time T, $P(u@R|v@T)$, using co-occurrence statistics between each of u and v and the contexts in which they appear. They then used this similarity measure as the emission probability in a Hidden Markov Model, where the state space comprises all terms at time T and each state emits terms at time R with these emission probabilities. The evaluation was done on some selected queries, where they showed the top-k list of translated queries for each of them. Kalurachchi *et al.* [14] used association rule mining to extract semantically related named entities used at different times by associating each named entity to its contextual event (verb), such that two entities that share the same event multiple times at different times are extracted as frequent rules. Then, for each transaction consisting of all entities sharing the same event, the strength of the relationship between each pair of entities is measured using the Jaccard coefficient between the frequencies of other contextual words, such as objects and adjectives, that each entity has. Using the translated list of named entities for the query, they achieved higher precision and recall for retrieving relevant documents from a corpus containing the USA President's speeches from 1790 to 2006 than by using the original query. Our work is different from these methods in that we use the whole document as a query, not just a named entity, for which we would like to retrieve its most similar documents, and so these approaches cannot work efficiently for the problem addressed here, since they model words only, whereas we need to model documents for the task of document similarity search.

Language change has also been studied for Historic Document Retrieval (HDR), which is concerned about retrieving relevant documents to a query written in today's language, e.g., Modern English, from a pool of documents written in older languages, e.g., Middle English [16, 8, 9, 10, 7]. The approaches proposed to solve this problem mainly rely on two sources of information: available dictionaries and spelling variations over time. For instance, Efron [7] used the MorphAdorner dictionary that contains a list of (modern, archaic) pairs of words as his source of dictionary evidence, and the string edit distance as a measure of string similarity between two terms to account for spelling variations of the same term over time. In our work, we model the changes in the same language, i.e., Modern English, so these types of evidence are not appropriate for the problem addressed in this work.

Other researchers focused on modeling the dynamic evolution of topics over time using probabilistic topic models [2, 4, 11, 27, 29]. Dynamic Topic Model (DTM) [4] is an extension to LDA that captures the evolution of topics over time by learning different consecutive term-topic matrices that have smooth transitions over time. Ahmed and Xing [2] introduced infinite DTM (iDTM) that considers the change of topic popularity, topic word distribution and number of topics over time. Han *et al.* [11] presented a dynamic rank factor model (DRFM), which is capable of learning temporal changes in the importance of topics and the correlations among topics and words over time. Having time-specific term distributions over the latent topics in these models can be also used to explicitly model word sense evolution, since each term has multiple representations over the topics in different time spans. Most of these models were evaluated based on their temporal perplexity and were not evaluated on real tasks, such as document similarity search or classification. However, we believe that modeling changes in language use can improve the performance of document similarity search for queries done on longitudinal document collections.

3. NOTATIONS

Boldface uppercase letters will be used to represent matrices, boldface lowercase letters to represent vectors, and calligraphic letters will be used to represent sets. The ith row of matrix $\mathbf{X}$ is represented as $\mathbf{x}_i$, whereas the ith column of matrix $\mathbf{X}$ is represented as $\mathbf{x}_i^T$.

A matrix $\mathbf{D}$ represents the document-term matrix of size $N \times M$, where N denotes the number of documents and M denotes the number of words in the vocabulary. In the methods that will be developed, $\mathbf{D}$ is factored into a document-topic matrix $\mathbf{U}$ of size $N \times K$ and a term-topic matrix $\mathbf{V}$ of size $M \times K$, where each document and term is represented as a point in the K-dimensional latent topic space.

Each document in a collection has a publication year associated with it, which is assumed to be the time when the document was written. A *time span* s is a consecutive period of time measured in years. The documents in a collection can be divided into S disjoint subsets based on their publication years, each of which corresponds to a time span. Given a time span s and a document-term matrix $\mathbf{D}$, then $\mathbf{D}_s$ of size $N_s \times M$ is the span-specific document-term submatrix of $\mathbf{D}$ that contains only the rows of $\mathbf{D}$ corresponding to the documents that were published in s. Similarly with $\mathbf{D}$, the factored representation of $\mathbf{D}_s$ will be denoted by $\mathbf{U}_s\mathbf{V}_s^T$, where $\mathbf{U}_s$ and $\mathbf{V}_s$ are the $N_s \times K$ document-topic and $M \times K$ term-topic matrices of the documents in time span s, respectively.

Documents in the collections have links to each other, representing the citation network. The citation network will be denoted with matrix $\mathbf{W}$, where w_{ij} is the weight of the link between documents i and j. The weight can be binary, i.e., $w_{ij} = 1$ for linked document pairs, and 0 otherwise, or it can be a function of the publication times of the two linked documents. This is further discussed in Section 4.1.3. The set of pairs $\mathcal{P}$ is the set of linked documents, i.e., the document pair $(i, j) \in \mathcal{P}$ if and only if document i cites document j.

4. METHODS

To improve the retrieval performance of queries done on longitudinal document collections, we address the problem of language change over time by leveraging two types of information: the citation network that often exists in these collections and the changes in the terms' usage frequency over time. We first explain how to incorporate link information in both static and dynamic topic models by adding link regularization to them, then we present a novel transition regularization technique in dynamic topic models based on the changes that occur in the terms' usage frequency over time.

We use the non-probabilistic static topic model, RLSI [28], and Dynamic Smooth RLSI (a combination of RLSI and DTM) as the baseline static and dynamic models, respec-

tively, since RLSI was shown to have the best retrieval performance compared to other models. We use the static version of RLSI as a baseline to compare it with its dynamic version, since dynamic topic models were not used in document similarity search before.

4.1 Incorporating Link Information in Topic Models

A link between a pair of documents provides an indication that the two documents are related. Incorporating citation information with content information was found to be useful for the task of document classification, where citation information complements content information to place similar documents close to each other in the documents latent space. This type of constraint in matrix factorization is called "link regularization" [18]. Citation information can be useful for document similarity search as well, especially if the two linked documents were written at different time periods, since their language use has most probably changed. For this reason, we add link regularization to both static and dynamic topic models to bring closer together pairs of linked documents while estimating their representations in the latent space.

4.1.1 Static Topic Models

Given a document-term matrix $\mathbf{D}$, RLSI [28] formalizes the problem of learning the document-topic matrix $\mathbf{U}$ and the term-topic matrix $\mathbf{V}$ as a regularized matrix factorization approach by solving the following optimization problem:

$$\min_{\mathbf{U},\mathbf{V}} \frac{1}{2}||\mathbf{D}-\mathbf{U}\mathbf{V}^T||_F^2+\frac{\alpha}{2}||\mathbf{U}||_F^2+\frac{\beta}{2}||\mathbf{V}||_F^2, \quad (1)$$

where α and β are the parameters controlling the regularization on $\mathbf{U}$ and $\mathbf{V}$, respectively[2]. We will refer to this method as *Static* RLSI, or **SRLSI**.

We leverage citation information that exists in the document collection by adding link regularization to the objective function in Eq. (1) as:

$$\min_{\mathbf{U},\mathbf{V}} \left(\frac{1}{2}||\mathbf{D}-\mathbf{U}\mathbf{V}^T||_F^2+\frac{\alpha}{2}||\mathbf{U}||_F^2+\frac{\beta}{2}||\mathbf{V}||_F^2 + \frac{\theta}{2}\sum_{(i,j)\in\mathcal{P}} w_{ij}||\mathbf{u}_i-\mathbf{u}_j||_2^2\right), \quad (2)$$

where θ is the link regularization controlling parameter and w_{ij} is the weight of the link for the linked document pair (i,j). This weight can be binary, as proposed in [18], or it can be computed based on the documents publication times, which we will discuss in more detail in Section 4.1.3. We will refer to the Static RLSI model with link regularization as **SRLSI+link**.

The objective function in Eq. (1) is not jointly convex with respect to the two variables $\mathbf{U}$ and $\mathbf{V}$. However, when one of them is fixed, it becomes convex with respect to the other one. Hence, by alternately minimizing it with respect to $\mathbf{U}$ then with respect to $\mathbf{V}$, it is guaranteed to converge to a local minimum. When one of the matrices is fixed, updating the other matrix becomes an $L2$-regularized least squares problem, which has an exact solution. Solving for $\mathbf{V}$ in Eq. (2) is the same as in Eq. (1). To find $\mathbf{U}$ in Eq. (2), we use coordinate descent to update each entry u_{nk} of $\mathbf{U}$, while keeping all other entries fixed. Let $\mathbf{V}_{\setminus k}$ the matrix of $\mathbf{V}$ with the k^{th} column removed, $\mathbf{v}_k^T$ be the k^{th} column vector of $\mathbf{V}$, and $\mathbf{u}_{n\setminus k}$ the vector of $\mathbf{u}_n$ with the k^{th} entry removed. We can rewrite the objective function in Eq. (2) as a function with respect to u_{nk} as:

$$\begin{aligned}
\mathcal{L}(u_{nk}) &= \Bigg(\frac{1}{2}||\mathbf{d}_n-\mathbf{V}_{\setminus k}\mathbf{u}_{n\setminus k}-u_{nk}\mathbf{v}_k||_2^2+\frac{\alpha}{2}u_{nk}^2 \\
&\quad+\frac{\alpha}{2}||\mathbf{u}_{n\setminus k}||_2^2+\frac{\theta}{2}\sum_{(n,i)\in\mathcal{P}} w_{ni}(u_{nk}-u_{ik})^2 \\
&\quad+\frac{\theta}{2}\sum_{(n,i)\in\mathcal{P}} w_{ni}||\mathbf{u}_{n\setminus k}-\mathbf{u}_{i\setminus k}||_2^2\Bigg) \\
&= \frac{1}{2}||\mathbf{v}_k||_2^2u_{nk}^2-\Big(\mathbf{d}_n-\mathbf{V}_{\setminus k}\mathbf{u}_{\setminus k}\Big)^T\mathbf{v}_k u_{nk} \\
&\quad+\frac{\alpha}{2}u_{nk}^2+\frac{\theta}{2}\sum_{(n,i)\in\mathcal{P}} w_{ni}(u_{nk}-u_{ik})^2+const \\
&= \frac{1}{2}s_{kk}^2u_{nk}^2-\Big(r_{nk}-\sum_{l\neq k}s_{kl}u_{nl}\Big)u_{nk}+\frac{\alpha}{2}u_{nk}^2 \\
&\quad+\frac{\theta}{2}\sum_{(n,i)\in\mathcal{P}} w_{ni}(u_{nk}-u_{ik})^2+const, \quad (3)
\end{aligned}$$

where s_{ij} and r_{ij} are the entries of the $K\times K$ matrix $\mathbf{S}=\mathbf{V}^T\mathbf{V}$ and the $N\times K$ matrix $\mathbf{R}=\mathbf{D}\mathbf{V}$, respectively, and *const* is a constant with respect to u_{nk}. We can then solve for u_{nk} as:

$$u_{nk}=\frac{\theta\sum_{(n,i)\in\mathcal{P}} w_{ni}u_{ik}+r_{nk}-\sum_{l\neq k}s_{kl}u_{nl}}{s_{kk}+\alpha+\theta\sum_{(n,i)\in\mathcal{P}} w_{ni}}. \quad (4)$$

4.1.2 Dynamic Topic Models

In order to better handle language change over time, we need to have a term-topic matrix for each time span, since each term can have a different distribution over the latent topics in each time span, depending on the change that occurred to it (if any). We develop Dynamic Smooth RLSI (**DSRLSI**), which is a dynamic version of RLSI that combines RLSI with the smooth transition regulation used in DTM [4]. DSRLSI enforces the coupling between consecutive term-topic matrices by having *smooth transition* regularization (that was used in DTM), which is a regularization term in the objective function that penalizes the Frobenius norm of the difference between each two consecutive term-topic matrices. The objective function for DSRLSI is:

$$\min_{\substack{\mathbf{U}_s,\mathbf{V}_s\\ s=1..S}} \left(\frac{1}{2}\sum_{s=1}^{S}\Big(||\mathbf{D}_s-\mathbf{U}_s\mathbf{V}_s^T||_F^2+\alpha||\mathbf{U}_s||_F^2+\beta||\mathbf{V}_s||_F^2\Big) + \frac{1}{2}\sum_{s=2}^{S}||\mathbf{V}_s-\mathbf{V}_{s-1}||_F^2\right), \quad (5)$$

where $\mathbf{D}_s$ is the document-term matrix for the documents that appear in time span s, and $\mathbf{U}_s$ and $\mathbf{V}_s$ are the learned document- and term-topic matrices in s.

Adding link regularization to DSRLSI, which we will refer

[2]Note that RLSI can have different regularization norms on both $\mathbf{U}$ and $\mathbf{V}$. In our experiments (not reported here) we found that $L2$ regularization achieved the best results and for this reason our methods will only use $L2$ regularization.

to as **DSRLSI+link**, the objective function becomes:

$$\min_{\substack{\mathbf{U}_s,\mathbf{V}_s \\ s=1..S}} \left(\frac{1}{2} \sum_{s=1}^{S} \left(||\mathbf{D}_s - \mathbf{U}_s \mathbf{V}_s^T||_F^2 + \alpha ||\mathbf{U}_s||_F^2 + \beta ||\mathbf{V}_s||_F^2 \right) + \frac{\theta}{2} \sum_{(i,j)\in\mathcal{P}} w_{ij} ||\mathbf{u}_i - \mathbf{u}_j||_2^2 + \frac{1}{2} \sum_{s=2}^{S} ||\mathbf{V}_s - \mathbf{V}_{s-1}||_F^2 \right). \quad (6)$$

Finding the optimal $\mathbf{U}_s$ in Eqs. (5) and (6) is the same as finding $\mathbf{U}$ as done in Eqs. (3) and (4), with replacing $\mathbf{d}_n$ with $\mathbf{d}_{s,n}$, $\mathbf{u}_n$ with $\mathbf{u}_{s,n}$ and $\mathbf{V}$ with $\mathbf{V}_s$, where $\mathbf{d}_{s,n}$ is the n^{th} original document vector in s and $\mathbf{u}_{s,n}$ is the n^{th} document latent representation in s. To learn $\mathbf{V}_s$ in Eqs (5) and (6), we can rewrite the objective function with respect to $\mathbf{v}_{s,m}$ as:

$$\mathcal{L}(\mathbf{v}_{s,m}) = \frac{1}{2} ||\mathbf{d}_{s,m}^T - \mathbf{U}_s \mathbf{v}_{s,m}||_2^2 + \frac{\beta}{2} ||\mathbf{v}_{s,m}||_2^2 + \frac{1}{2} ||\mathbf{v}_{s,m} - \mathbf{v}_{s-1,m}||_2^2 + \frac{1}{2} ||\mathbf{v}_{s+1,m} - \mathbf{v}_{s,m}||_2^2, \quad (7)$$

where $\mathbf{d}_{s,m}^T$ is the m^{th} column of $\mathbf{D}_s$ and $\mathbf{v}_{s,m}$, $\mathbf{v}_{s-1,m}$ and $\mathbf{v}_{s+1,m}$ are the m^{th} term latent representations in time spans s, $s-1$ and $s+1$, respectively. Hence, the optimal solution for $\mathbf{v}_{s,m}$ is:

$$\mathbf{v}_{s,m} = \left(\mathbf{U}_s^T \mathbf{U}_s + (\beta+2)\mathbf{I} \right)^{-1} \left(\mathbf{U}_s \mathbf{d}_{s,m}^T + \mathbf{v}_{s+1,m} + \mathbf{v}_{s-1,m} \right). \quad (8)$$

The matrix inversion in Eq. (8) can be done easily, since the matrix is of dimension K, where K is usually less than 100, and it is be computed once for updating the whole matrix $\mathbf{V}_s$.

4.1.3 Computation of Link Weights

Recall from Sections 4.1.1 and 4.1.2 that our models associate a weight with each link, which is used to control the degree of importance of the various links. We used the documents' publication times to determine these weights based on the difference between the two linked documents timestamps. Our intuition behind this idea is that when two linked documents have large differences in their publication dates, their language use is most probably more dissimilar than linked documents that belong to closer time spans. Therefore, we assign larger weights to links that have a larger publication time difference than those that have a smaller time difference. This will force similar documents with large differences in their dates and have more dissimilar bag-of-words representations (more than similar documents with small differences in their dates) to come closer together in the latent space.

We experimented with four link weighting functions: binary (*bin*), logarithmic (*log*), linear (*lin*), and quadratic (*quad*). The *bin* function assigns a weight of 1 to all links in the dataset, whereas the other functions assign different weights based on the publication time difference of the linked documents. Specifically, given a link from document i to document j whose publication dates are t_i and t_j, respecitively, then *log* assigns a weight of $1 + \log_2(t_i - t_j)$; *lin* assigns a weight of $1 + (t_i - t_j)$, and *quad* assigns a weight of $1 + (t_i - t_j)^2$. Thus, these three functions progressively assign higher weights to the links based on the difference in the publication dates of the documents involved.

4.2 Incorporating Changes in Terms' Usage Frequency in Dynamic Topic Models

Recall from Section 4.1.2 that DTM learns different term-topic matrices by assuming that these matrices evolve smoothly (with a fixed rate) over time. However, changes in each term's distribution over the latent topics occur at different times and with different rates over time. As was shown in many linguistic and computational studies, changes in the usage frequency of terms over time are usually a good signal of changes in the terms' senses and uses (Section 2.1). Therefore, we make use of this type of information to improve the latent representation of words in dynamic topic models. Instead of having smooth transition regularization on consecutive $\mathbf{V}_s$ as in DSRLSI, we assign a specific weight to each term for each pair of consecutive time spans. The idea behind having these *term-time-specific* weights is that when there is a huge difference in the frequency of a term m in some time span s than in its previous span $s-1$, there should not be a smooth transition between $\mathbf{v}_{s,m}$ and $\mathbf{v}_{s-1,m}$, since m might have been used in different contexts (e.g., different topics) in s where it was not used before in $s-1$. On the other hand, when the frequencies of m in s and $s-1$ are similar, the two term distributions over the latent topics in s and $s-1$ should be similar, as there is no evidence that there has been a change in the term's meaning in these time spans.

To model this type of regularization, which we call *term-time-specific* transition regularization, we weight the transition regularization of each term in each pair of consecutive time spans as follows. Terms that have similar normalized frequencies in two consecutive time spans will have a higher penalty (weight) on their transition regularization, whereas terms whose normalized frequencies in two consecutive time spans are different will have less penalty on their transition regularization. This allows the same term to have different distributions over the latent topics in two consecutive time spans when its meaning or sense changes. We will refer to this model as Dynamic Term-time-specific RLSI, or **DTRLSI**. The objective function for DTRLSI is:

$$\min_{\substack{\mathbf{U}_s,\mathbf{V}_s \\ s=1..S}} \left(\frac{1}{2} \sum_{s=1}^{S} \left(||\mathbf{D}_s - \mathbf{U}_s \mathbf{V}_s^T||_F^2 + \alpha ||\mathbf{U}_s||_F^2 + \beta ||\mathbf{V}_s||_F^2 \right) + \frac{1}{2} \sum_{s=2}^{S} \sum_{m=1}^{M} \lambda_{s,m} ||\mathbf{v}_{s,m} - \mathbf{v}_{s-1,m}||_2^2 \right), \quad (9)$$

where $\lambda_{s,m}$ is the parameter controlling the transition regularization for term m in time spans s and $s-1$. We assign $\lambda_{s,m}$ as:

$$\lambda_{s,m} = \frac{0.1}{10^{\gamma} * |\operatorname{ndf}_s(m) - \operatorname{ndf}_{s-1}(m)| + 0.1},$$

where $\operatorname{ndf}_s(m)$ is the normalized document frequency, i.e., the percentage of documents where term m appears in time span s. The parameter γ is an integer (usually 0 or 1) that controls the range of transition regularization. This function makes the maximum value for $\lambda_{s,m} = 1$. Link regularization can be added to Eq. (9) to have DTRLSI with link regularization, which we will refer to as **DTRLSI+link**.

Since the weights are different for every $\mathbf{v}_{s,m}$, we cannot use Eq. (8) for finding $\mathbf{v}_{s,m}$, as the matrix inversion needs

Table 1: Time complexities for all models per iteration.

Model	Time Complexity for Updating $\mathbf{U}$	Time Complexity for Updating $\mathbf{V}$
SRLSI	$\max\{MK^2, NK \times \text{AvgRL}, NK^2\}$	$\max\{MK^2, MK \times \text{AvgCL}, NK^2\}$
SRLSI+link	$\max\{MK^2, NK \times \text{AvgRL}, NK^2T_i \times \text{AvgNL}\}$	same as SRLSI
DSRLSI	same as SRLSI	$\max\{SMK^2, SMK \times \text{AvgCL}, NK^2\}$
DSRLSI+link	$\max\{SMK^2, NK \times \text{AvgRL}, NK^2T_i \times \text{AvgNL}\}$	same as DSRLSI
DTRLSI	same as SRLSI	$\max\{SMK^2, SMKT_i \times \text{AvgCL}, NK^2\}$
DTRLSI+link	same as DSRLSI+link	same as DTRLSI

AvgRL denotes the average row length in $\mathbf{D}$, i.e., the average number of terms per document. Similarly, AvgCL denotes the average column length in $\mathbf{D}$. AvgNL denotes the average number of links per document. T_i denotes the number of inner iterations when using coordinate descent.

Table 2: Datasets statistics.

Dataset	# Documents	Vocabulary Size	nnz	#Links	#Test Documents	#Test links (early)	#Test links (recent)
USPTO	40,000	38,868	16,578,274	49,421	707	813	3,609
ACL2013	18,897	27,059	9,673,340	101,878	1,966	390	16,867

The column "nnz" shows the number of non-zero entries in the document-term matrix. "#Links" shows the total number of links in the whole dataset. "#Test Documents" shows the number of documents that were used in the test set. The last two columns show the number of documents linked to the test documents in the early and recent halves of time spans, respectively.

to be done for every term m, which is very expensive since M is usually large. Therefore, we use coordinate descent to update every entry $v_{s,mk}$ of every vector $\mathbf{v}_{s,m}$. The loss function in Eq. (9) can be written with respect to $v_{s,mk}$ as:

$$\begin{aligned}\mathcal{L}(v_{s,mk}) &= \frac{1}{2}||\mathbf{d}_{s,m}^T - \mathbf{U}_{s,\backslash k}\mathbf{v}_{s,m\backslash k}||_2^2 + \frac{\beta}{2}(v_{s,mk})^2 \\ &+ \frac{\beta}{2}||\mathbf{v}_{s,m\backslash k}||_2^2 + \frac{\lambda_{s,m}}{2}(v_{s,mk} - v_{s-1,mk})^2 \\ &+ \frac{\lambda_{s,m}}{2}||\mathbf{v}_{s,m\backslash k} - \mathbf{v}_{s-1,m\backslash k}||_2^2 \\ &+ \frac{\lambda_{s+1,m}}{2}(v_{s+1,mk} - v_{s,mk})^2 \\ &+ \frac{\lambda_{s+1,m}}{2}||\mathbf{v}_{s+1,m\backslash k} - \mathbf{v}^s_{m\backslash k}||_2^2 \\ &= \frac{1}{2}s_{kk}v_{s,mk}^2 - \Big(r_{mk} - \sum_{l\neq k} s_{kl}v_{s,ml}\Big)v_{s,mk} \\ &+ \frac{\beta}{2}v_{s,mk}^2 + \frac{\lambda_{s,m}}{2}(v_{s,mk} - v_{s-1,mk}) \\ &+ \frac{\lambda_{s+1,m}}{2}(v_{s+1,mk} - v_{s,m}) + const, \qquad (10)\end{aligned}$$

where s_{ij} and r_{ij} are the entries of the $K \times K$ matrix $\mathbf{S} = \mathbf{U}_s^T\mathbf{U}_s$ and the $M \times K$ matrix $\mathbf{R} = \mathbf{D}_s^T\mathbf{U}_s$, respectively, and *const* is a constant with respect to $v_{s,mk}$. We can then update $v_{s,mk}$ as:

$$v_{s,mk} = \frac{r_{mk} - \sum_{l\neq k} s_{kl}v_{s,ml} + \lambda_{s,m}v_{s-1,m} + \lambda_{s+1,m}v_{s+1,m}}{s_{kk} + \beta + \lambda_{s,m} + \lambda_{s+1,m}}. \qquad (11)$$

4.3 Computational Requirements

Table 1 shows the time complexity for updating each of the $\mathbf{U}$ and $\mathbf{V}$ matrices in each model per iteration. Adding link regularization to SRLSI increases the time complexity of updating $\mathbf{U}$ by $T_i \times \text{AvgNL}$, where T_i is the number of iterations needed for performing coordinate descent and AvgNL denotes the average number of links per document. Comparing SRLSI with DSRLSI, we can see that the time complexity of $\mathbf{V}$ increases with a factor of S, since S different term-topic matrices need to be learned, whereas by comparing SRLSI with DTRLSI, the increase is by a factor of ST_i. By comparing SRLSI+link and DSRLSI+link, we see that updating $\mathbf{U}$ in the latter model may take more time than the former only if the first term dominates the third one. Finally, we see that the time complexity of updating $\mathbf{V}$ in DTRLSI is larger than that of DSRLSI by a factor of T_i.

5. EXPERIMENTAL EVALUATION

5.1 Datasets

We evaluate the performance of our methods on two different datasets. The first is derived from the US utility patents [1]. We used the specification section of each patent as its content. Each patent is assigned by the US patent office to one primary node within the International Patent Classification (IPC classification) hierarchical classification system based on the field of its invention. We extracted a subset from this dataset by selecting a set of 35 IPC classes that have a large number of documents and then randomly retrieving a subset of documents from each IPC class that were granted between 1970 and 2009. The set of 35 IPC classes were selected so that to they contain related patents (i.e., they are siblings within IPC's hierarchical classification system). This is done in order to create a dataset for which the task of document similarity searcg will be harder, as it will contain a fairly thematically homogeneous collection of documents. We will refer to this dataset as **USPTO**[3]. The second dataset is the 2013 release of the Association of Computational Linguistics (ACL) Anthology dataset (**ACL2013**)[21], which contains all scientific papers published in many ACL venues between 1965 and 2013. The datasets' statistics are summarized in Table 2.

We performed lemmatization and removed stop words, words of length less than three and words that occurred less than 30 times in each dataset. We computed the weight of each term in each document as its TF-IDF value, normalized

[3]The dataset used can be found here: `http://goo.gl/B1k2Yj`

Table 3: Effect of link regularization on SRLSI.

	USPTO			ACL2013		
Method	NDCG(all)	NDCG(early)	NDCG(recent)	NDCG(all)	NDCG(early)	NDCG(recent)
SRLSI	0.1645	0.1339	0.1740	0.1757	0.1233	0.1782
SRLSI+link(bin)	**0.1684**†	0.1417†	0.1767†	0.1843†	0.1455†	0.1861†
SRLSI+link(log)	0.1682†	0.1404†	**0.1768**†	0.1847 † ‡	0.1477 † ‡	0.1864†
SRLSI+link(lin)	0.1683†	**0.1424** † ‡	0.1764†	**0.1853** † ‡	0.1501 † ‡	**0.1870** † ‡
SRLSI+link(quad)	0.1674†	0.1403†	0.1758†	0.1845†	**0.1535** † ‡	0.1859†

These results are based on 60-topic models. † indicates statistical significance over SRLSI, whereas ‡ indicates statistical significance over SRLSI+link(bin). Bold-faced entries represent the best performance obtained for each metric.

Table 4: Effect of link regularization on DSRLSI.

	USPTO			ACL2013		
Method	NDCG(all)	NDCG(early)	NDCG(recent)	NDCG(all)	NDCG(early)	NDCG(recent)
DSRLSI	0.1632	0.1370	0.1714	0.1680	0.1262	0.1700
DSRLSI+link(bin)	0.1671†	0.1425	0.1748†	0.1841†	0.1920†	0.1838†
DSRLSI+link(log)	**0.1679**†	0.1426	**0.1758**†	0.1848†	0.1906†	0.1845†
DSRLSI+link(lin)	**0.1679**†	0.1425	**0.1758**†	**0.1867** † ‡	0.1991†	**0.1861** † ‡
DSRLSI+link(quad)	0.1646	**0.1482** † ‡	0.1698	0.1841†	**0.2048** † ‡	0.1832†

These results are based on 60-topic models. † indicates statistical significance over DSRLSI, whereas ‡ indicates statistical significance over DSRLSI+link(bin). Bold-faced entries represent the best performance obtained for each metric.

by the document length.

5.2 Evaluation Methodology and Metrics

We evaluated the performance of our methods against the baselines on the task of document similarity search (for patents, this task is called prior art candidate search, which is defined as finding patent documents that may constitute prior art for a given patent application), where the user is interested in retrieving similar documents published in a specific time span for query documents that were published recently. Since we do not have relevance information for these datasets, we followed the approach of CLEF-IP 2011 competition for prior art candidate search[4] to create the ground truth relevance scores, so we considered linked pairs of documents to be similar. We assigned a relevance score of one to the documents linked to the query document and a score of zero to all other documents. We divided the documents in each dataset according to their publication times into different 5-year time spans. For each dataset, we split its corresponding document-term matrix $\mathbf{D}$ into three matrices: $\mathbf{D}_{train}$, $\mathbf{D}_{val}$ and $\mathbf{D}_{test}$. We removed from $\mathbf{D}$ all documents that were published in the most recent decade and have five or more links with other documents in the whole dataset to construct $\mathbf{D}_{train}$, then we randomly and evenly divided the removed set of documents to construct $\mathbf{D}_{val}$ and $\mathbf{D}_{test}$ (so the links used for validation and test were not included during learning the models). The matrix $\mathbf{D}_{train}$ was used to estimate each of the models. Then, each document in $\mathbf{D}_{test}$ was used as a query, where we computed the cosine similarity between its vector and the vectors of all the documents in $\mathbf{D}_{train}$ and ranked them according to their similarity values in non-increasing order to get the ranked list of documents returned by the model.

To assess the performance of each model, we computed the Normalized Discounted Cumulative Gain (NDCG) score. For each query, we ranked the documents that exist in each time span to get an NDCG score for each of these spans. We report the average NDCG scores over all time spans (**NDCG (all)**) as well as the average NDCG score for the early half of the time spans (**NDCG(early)**) and the recent half of the time spans (**NDCG(recent)**). This allows us to assess the performance of each method for retrieving similar documents that are far away in time from the query documents. We also measured the statistical significance of our methods against the baselines as well as our methods against each other using one-sided t-test with a p-value of less than 0.05.

[4] `http://www.ir-facility.org/prior-art-search1`

5.3 Model Selection

We did an extensive search over the parameter space for the various methods. The regularization parameters α and β on $\mathbf{U}$ and $\mathbf{V}$, respectively, were chosen from the set of values: {0.01, 0.05, 0.1, 0.5, 1}. The link regularization parameter θ was set to one of the values: {0.1, 1, 10, 20, 30, 40, 50}, and {0.001, 0.01, 0.1, 1, 10, 20, 30} for USPTO and ACL2013, respectively. We experimented with two values for the parameter γ in DTRLSI: $\gamma = 1$, where λ_m^s is in the range: $[0.0198, 1]$ and $[0.0202, 1]$, and $\gamma = 0$, where λ_m^s is in the range: $[0.1681, 1]$ and $[0.1709, 1]$ for USPTO and ACL2013, respectively. Finally, we set the latent space dimensionality in the ranges [20, 100] with a step of 20.

The matrix $\mathbf{D}_{val}$ was used to select the best performing parameters (α and β for non-link regularized models and α, β and θ for link regularized models) in terms of the NDCG(all) score. The matrix $\mathbf{D}_{test}$ was then used with these best performing parameters to get the relevance ranking scores for each model.

6. EXPERIMENTAL RESULTS

We structure the presentation of our results into five parts. The first studies the effect of incorporating link information with static and dynamic RLSI. The second compares the retrieval performance of dynamic RLSI with smooth and term-time-specific transition regularization techniques. The third examines the effect of the different link weighting func-

Table 5: Effect of link regularization on DTRLSI.

	USPTO			ACL2013		
Method	NDCG(all)	NDCG(early)	NDCG(recent)	NDCG(all)	NDCG(early)	NDCG(recent)
DTRLSI	0.1629	0.1375	0.1708	0.1662	0.1270	0.1680
DTRLSI+link(bin)	0.1655†	0.1465†	0.1715	0.1848†	0.1909†	0.1845†
DTRLSI+link(log)	0.1670†	0.1486†	0.1727	0.1843†	0.1912†	0.1840†
DTRLSI+link(lin)	**0.1678**†	0.1488†	0.1737†	0.1819†	0.1887†	0.1815†
DTRLSI+link(quad)	0.1653†	**0.1503**†	0.1700	0.1860†	**0.2126** † ‡	0.1848†

These results are based on 60-topic models. † in DTRLSI+link models indicates statistical significance over DTRLSI, whereas ‡ indicates statistical significance over DTRLSI+link(bin). Bold-faced entries represent the best performance obtained for each metric.

Table 6: Summary of best results achieved by static and dynamic RLSI without and with link regularization.

	USPTO			ACL2013		
Method	NDCG(all)	NDCG(early)	NDCG(recent)	NDCG(all)	NDCG(early)	NDCG(recent)
SRLSI	0.1645	0.1339	0.1740	0.1757	0.1233	0.1782
DSRLSI	0.1632	0.1370	0.1714	0.1680	0.1262	0.1700
DTRLSI	0.1629	0.1375	0.1708	0.1662	0.1270	0.1680
SRLSI+link(lin)	**0.1683**	0.1424	**0.1764**∗	0.1853	0.1501	**0.1870**∗
DSRLSI+link(lin)	0.1679	0.1425	0.1758	**0.1867**	0.1991†	0.1861
DTRLSI+link(quad)	0.1653	**0.1503** † ‡	0.1700	0.1860	**0.2126** † ‡	0.1848

These results are based on 60-topic models. † indicates statistical significance over SRLSI+link, ‡ indicates statistical significance over DSRLSI+link, and * indicates statistical significance over DTRLSI+link. Bold-faced entries represent the best performance obtained for each metric.

tions used in link regularized models. We present the results in these three sections using 60 dimensions, which we considered to be a good representative for all other dimensions. The fourth presents some qualitative analysis done on the retrieval performance of the proposed models as compared to the baselines. Finally, we present the timing performance for all models.

6.1 Effect of Adding Link Information to Static and Dynamic RLSI

Tables 3 and 4 show the retrieval performance achieved by Static and Dynamic Smooth RLSI with and without link regularization. These results show that incorporating link information improves the performance of both models, especially for being able to rank high the relevant documents that were published in earlier times. This confirms our initial hypothesis that the citation network provides important information that can be used to rank high relevant documents whose language may have changed due to the passage of time.

Moreover, we can see that link regularization has improved the performance for both models in ACL2013 much more than in USPTO. We believe that this is due to having a much larger number of links with respect to the number of documents in ACL2013 than in USPTO (as shown in Table 2). This allowed ACL2013 documents to be much more connected than USPTO, which as a consequence helped link regularization in properly estimating the latent documents representation in ACL2013.

6.2 Effect of Term-time-specific Transition Regularization on Dynamic RLSI

Table 5 shows the retrieval performance for dynamic RLSI with term-time-specific transition regularization with and without link regularization. Comparing DSRLSI and DTRLSI without link regularization (as shown in Table 6), we can see that there is no significant difference in their retrieval performance. However, after adding link regularization, having term-time-specific transition regularization achieves better NDCG(early) scores, whereas NDCG(all) and NDCG(recent) scores do not have significant differences among the two schemes. This confirms our hypothesis that different terms should have different transition regularization weights, since terms change their senses with different rates over time, and one way to capture this difference is based on terms' usage frequency changes over time, as we explained earlier in Section 4.2.

Note that in relation to the gains achieved due to the use of the citation network, these results are consistent with the results presented in Section 6.1.

6.3 Effect of Different Link Weighting Functions

From Tables 3-5, we can see that by adding larger weights to links that belong to document pairs with larger differences in their publication dates, the methods achieve better retrieval performance than having the same weights on all links. This confirms our intuition behind using time-aware link weighting functions (Section 4.1.3) that the larger the difference between the publication dates of two related documents, the more dissimilar their language use is.

By comparing the performance of these link weighting functions, the *quad* function seems to outperform all other weighting functions in terms of NDCG(early), except in SRLSI for USPTO, whereas there is some variation in the performance of all four functions for NDCG(recent). We believe that this is because the *quad* function assigns much larger weights to links that belong to document pairs with larger differences in their dates, and hence it was more capable of placing these pairs closer together in the latent space than all other functions.

6.4 Qualitative Analysis

We also performed qualitative analysis on the retrieval performance of SRLSI and DSRLSI with and without link regularization, in order to study the effect of link regular-

Table 7: Titles of documents added (removed) by adding link regularization to SRLSI.

Query Title: Simultaneous Ranking and Clustering of Sentences: A Reinforcement Approach to Multi-Document Summarization
− Recognition of Linear Context-free Rewriting Systems
− Polynomial Learnability and Locality of Formal Grammars
− An Algorithm for Determining Talker Location using a Linear Microphone Array and Optimal Hyperbolic Fit
− Evaluating Discourse Processing Algorithms
− THALES: A Software Package for Plane Geometry Constructions with a Natural Language Interface
− Inherently Reversible Grammars, Logic Programming and Computability
+ Parsing with Flexibility - Dynamic Strategies and Idioms in Mind
+ Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text
+ Statistical Parsing of Messages
+ Chart Parsing of Robust Grammars
+ Text on Tap: the ACL/DCI
+ Word Association Norms, Mutual Information, and Lexicography

This list contains the documents published during the time span 1990-1995. The plus (minus) sign denotes the documents that were added to (removed from) the top-10 retrieved documents by SRLSI+link as compared to SRLSI.

Table 8: Titles of documents added and removed by adding link regularization to DSRLSI.

Query Title: Simultaneous Ranking and Clustering of Sentences: A Reinforcement Approach to Multi-Document Summarization
− Semantic-Head-Driven Generation
− Comparing Two Grammar-based Generation - A Case Study
− A Uniform Architecture for Parsing, Generation and Transfer
− Generation and Translation - Towards A Formalism-Independent Characterization
− Generating from a Deep Structure
− Reversible Unification Based Machine Translation
− Handling Pragmatic Information With A Reversible Architecture
− Optimization Algorithms of Deciphering as the Elements of a Linguistic Theory
− An Augmented Context Free Grammar for Discourse
+ Word Association Norms, Mutual Information, and Lexicography
+ Noun Classification from Predicate Argument Structures
+ Automatic Learning for Semantic Collocation
+ A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text
+ Automatic Acquisition of Hyponyms on Large Text Corpora
+ Surface Grammatical Analysis for the Extraction of Terminological Noun Phrases
+ Class-Based n-gram Models of Natural Language
+ A Fast Algorithm for the Generation of Referring Expressions
+ Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text

This list contains the documents published during the time span 1990-1995. The plus (minus) sign denotes the documents that were added to (removed from) the top-10 retrieved documents by DSRLSI+link as compared to DSRLSI.

ization on placing related documents with large differences in their publication dates in the latent space. We randomly selected some query documents and analyzed their top-10 lists of documents retrieved by each model from the pool of documents published during the period 1990-1995, by looking at their titles. Tables 7 and 8 show the titles of the list of documents that were removed from the top-10 retrieved documents list after adding link regularization to SRLSI and DSRLSI, respectively, for a sample query document, as well as the list of documents that substituted them. Both SRLSI +link and DSRLSI+link were able to rank higher documents related to text analysis (compared to SRLSI and DSRLSI, respectively) that used different language from the query document.

For example, the retrieved paper entitled "Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text" used the term "lexical cohesion" to define a set of words that represent the same topic. The same definition is used in the query document by using the term "topic themes", which shows an example of term-to-term evolution that was captured by link regularization.

6.5 Timing Performance of the Models

Table 9 shows the timing performance for learning each model. As shown in the table, adding link regularization results in a slight increase in the time taken to learn each model. Comparing SRLSI with DSRLSI and DSRLSI with DTRLSI (without and with link regularization), we can see that the time is approximately the double, e.g., DSRLSI takes double the time that SRLSI takes.

7. CONCLUSION AND FUTURE WORK

In this paper, we presented methods for document similarity search that use two types of information to improve the retrieval performance in longitudinal document collections: citation information and changes in terms' usage frequency over time. We used these two types of information to regularize the latent representation of documents and terms while estimating them using static and dynamic topic models. We added link regularization to both static and dynamic topic models in order to bring closer together related documents that might have different content. Moreover, we used term-time-specific transition regularization in dynamic topic models to better regularize the transitions between the latent representation of terms in consecutive time spans according to their usage frequency changes instead of having smooth transition regularization for all terms in all time spans.

We compared the retrieval performance of our proposed models against the existing baselines on the task of document similarity search, where the user is interested in searching the collection for documents that are similar to a re-

Table 9: Timing Performance for All Models in Minutes.

	SRLSI	SRLSI+link	DSRLSI	DSRLSI+link	DTRLSI	DTRLSI+link
USPTO	84	107	175	217	383	416
ACL2013	30	42	111	133	265	283

These times are averaged over all the runs when using 60 dimensions and all combinations of the other parameters (see Section 5.3) for each model.

cent query document. Our results (summarized in Table 6) showed that incorporating link information with both static and dynamic RLSI is useful for similarity search, especially for retrieving relevant documents that were written at far away dates from the queries. In addition, we showed that link regularized dynamic topic models with term-time-specific transition regularization is better than with having smooth transition regularization for early document's retrieval, since term-time-specific transition regularization allows the terms to have transitions with different rates over time based on their frequency changes.

In the future, we plan to extend this work by leveraging other signals of language change, such as changes in the POS distributions of each term over time, which were studied in [17]. We believe that these signals will help further improve the retrieval performance of dynamic RLSI along with the changes in the terms' usage frequency, as was shown in the qualitative analysis done in [17].

8. REFERENCES

[1] The united states patent and trademark office. `http://www.uspto.gov/`.

[2] A. Ahmed and E. P. Xing. Timeline: A dynamic hierarchical dirichlet process model for recovering birth/death and evolution of topics in text stream. *arXiv preprint arXiv:1203.3463*, 2012.

[3] K. Berberich, S. J. Bedathur, M. Sozio, and G. Weikum. Bridging the terminology gap in web archive search. In *WebDB*, 2009.

[4] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *ICML*, 2006.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 2003.

[6] S. Dumais, G. Furnas, T. Landauer, S. Deerwester, S. Deerwester, et al. Latent semantic indexing. In *TREC*, 1995.

[7] M. Efron. Query representation for cross-temporal information retrieval. In *SIGIR*, 2013.

[8] A. Ernst-Gerlach and N. Fuhr. Retrieval in text collections with historic spelling using linguistic and spelling variants. In *JCDL*, 2007.

[9] A. Gotscharek, A. Neumann, U. Reffle, C. Ringlstetter, and K. U. Schulz. Enabling information retrieval on historical document collections: the role of matching procedures and special lexica. In *AND*, 2009.

[10] A. Gotscharek, U. Reffle, C. Ringlstetter, K. U. Schulz, and A. Neumann. Towards information retrieval on historical document collections: the role of matching procedures and special lexica. *IJDAR*, 2011.

[11] S. Han, L. Du, E. Salazar, and L. Carin. Dynamic rank factor model for text streams. In *NIPS*, 2014.

[12] Z. S. Harris. Distributional structure. *Word*, 1954.

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

[14] A. C. Kaluarachchi, A. S. Varde, S. Bedathur, G. Weikum, J. Peng, and A. Feldman. Incorporating terminology evolution for query translation in text retrieval with association rules. In *CIKM*, 2010.

[15] N. Kanhabua and K. Nørvåg. Exploiting time-based synonyms in searching document archives. In *JCDL*, 2010.

[16] M. Koolen, F. Adriaans, J. Kamps, and M. De Rijke. *A cross-language approach to historic document retrieval*. Springer, 2006.

[17] V. Kulkarni, R. Al-Rfou, B. Perozzi, and S. Skiena. Statistically significant detection of linguistic change. *arXiv preprint arXiv:1411.3315*, 2014.

[18] W.-J. Li and D.-Y. Yeung. Relation regularized matrix factorization. In *IJCAI*, 2009.

[19] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*. Cambridge university press Cambridge, 2008.

[20] A. M. McMahon. *Understanding language change*. Cambridge University Press, 1994.

[21] D. Radev, P. Muthukrishnan, V. Qazvinian, and A. Abu-Jbara. The acl anthology network corpus. *Language Resources and Evaluation*, 2013.

[22] S. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, and M. Gatford. Okapi at trec-3. In *TREC*, 1994.

[23] G. Salton, A. Wong, and C.-S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 1975.

[24] N. Tahmasebi, G. Gossen, N. Kanhabua, H. Holzmann, and T. Risse. Neer: An unsupervised method for named entity evolution recognition. In *COLING*, 2012.

[25] N. Tahmasebi and T. Risse. The role of language evolution in digital archives. In *SDA*, 2013.

[26] N. Tahmasebi, T. Risse, and S. Dietze. Towards automatic language evolution tracking, a study on word sense tracking. In *EvoDyn Workshop*, 2011.

[27] C. Wang, D. Blei, and D. Heckerman. Continuous time dynamic topic models. *arXiv preprint arXiv:1206.3298*, 2012.

[28] Q. Wang, J. Xu, H. Li, and N. Craswell. Regularized latent semantic indexing: A new approach to large-scale topic modeling. *TOIS*, 2013.

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

[30] D. T. Wijaya and R. Yeniterzi. Understanding semantic change of words over centuries. In *DETECT workshop*, 2011.