Context-aware Citation Recommendation

Qi He  
Penn State University  
qhe@ist.psu.edu

Jian Pei  
Simon Fraser University  
jpei@cs.sfu.ca

Daniel Kifer  
Penn State University  
dan+www10@cse.psu.edu

Prasenjit Mitra  
Penn State University  
pmitra@ist.psu.edu

C. Lee Giles  
Penn State University  
giles@ist.psu.edu

ABSTRACT
When you write papers, how many times do you want to make some citations at a place but you are not sure which papers to cite? Do you wish to have a recommendation system which can recommend a small number of good candidates for every place that you want to make some citations? In this paper, we present our initiative of building a context-aware citation recommendation system. High quality citation recommendation is challenging: not only should the citations recommended be relevant to the paper under composition, but also should match the local contexts of the places citations are made. Moreover, it is far from trivial to model how the topic of the whole paper and the contexts of the citation places should affect the selection and ranking of citations. To tackle the problem, we develop a context-aware approach. The core idea is to design a novel non-parametric probabilistic model which can measure the context-based relevance between a citation context and a document. Our approach can recommend citations for a context effectively. Moreover, it can recommend a set of citations for a paper with high quality. We implement a prototype system in CiteSeerX. An extensive empirical evaluation in the CiteSeerX digital library against many baselines demonstrates the effectiveness and the scalability of our approach.

Categories and Subject Descriptors
H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval

General Terms
Algorithms, Design, Experimentation

Keywords
Bibliometrics, Context, Gleason’s Theorem, Recommender Systems

1. INTRODUCTION
When you write papers, how many times do you want to make some citations at a place but you are not sure which papers to cite? For example, the left part of Figure 1 shows a segment of a query manuscript containing some citation placeholders (placeholders for short) marked as ”[?]”, where citations should be added. In order to fill in those citation placeholders, one needs to search the relevant literature and find a small number of proper citations. Searching for proper citations is often a labor-intensive task in research paper composition, particularly for junior researchers who are not familiar with the very extensive literature. Moreover, the volume of research undertaken and information available make citation search hard even for senior researchers.

Do you wish to have a recommendation system which can recommend a small number of good candidates for every place that you want to make some citations? High quality citation recommendation is a challenging problem for many reasons.

For each citation placeholder, we can collect the words surrounding as the context of the placeholder. One may think we can use some keywords in the context of a placeholder to search a literature search engine like Google Scholar or CiteSeerX to obtain a list of documents as the candidates for citations. However, such a method, based on keyword matching, is often far from satisfactory. For example, using query “frequent itemset mining” one may want to search for the first paper proposing the concept of frequent itemset mining, e.g. [1]. However, Google Scholar returns a paper about frequent closed itemset mining published in 2000 as the first result, and a paper on privacy preserving frequent itemset mining as the second choice. [1] does not appear in the first page of the results. CiteSeerX also lists a paper on privacy preserving frequent itemset mining as the first result. CiteSeerX fails to return [1] on the first page, either.

One may wonder, as we can model citations as links from citing documents to cited ones, can we use graph-based link prediction techniques to recommend citations? Graph-based link prediction techniques often require a user to provide sample citations for each placeholder, and thus shifts much of the burden to the user. And, graph-based link prediction
In this paper, we present an effective solution to the problem of citation recommendations for placeholders in query manuscripts. Our approach is context-aware, where a context of citation recommendations for placeholders in a query manuscript is a challenging task. The recommended citations for placeholders should satisfy two requirements. First, a citation recommendation needs to be explainable. Our problem is different from generating a bibliography list for a paper where a recommendation should discuss some ideas related to the query manuscript. In our problem, a recommended citation for a placeholder in a query manuscript should be relevant and authoritative to the particular idea that is being discussed at that point in the query manuscript. Different placeholders in the same query manuscript may need different citations.

Second, the recommendations for a manuscript need to take into account the various ideas discussed in the manuscript. For example, suppose we are asked to recommend citations for a query manuscript in which one section discusses mixture models and another section discusses nonparametric distributions. Citations to nonparametric mixture models may be ranked high since they are related to both sections in the same manuscript.

In summary, citation recommendation for placeholders (and for the overall bibliography) is a challenging task. The recommendations need to consider many factors: the query manuscript in whole, the contexts of the citation placeholders individually and collectively, and the literature articles individually. We need to construct an elegant mathematical framework for relevance and develop an efficient and scalable implementation.

In this paper, we present an effective solution to the problem of citation recommendations for placeholders in query manuscripts. Our approach is context-aware, where a context is a snippet of the text around a citation or a placeholder. The core idea is to design a novel non-parametric probabilistic model which can measure the context-based relevance between a citation context and a document. Our approach can recommend citations for a context effectively, which is the innovative part of the paper. Moreover, it can recommend a set of citations for a paper with high quality.

In addition to the theoretical contribution, we also implement a prototype system in CiteSeerX. Figure 2 shows a real example in our demo system, where a user submits a citation context with a placeholder and the title/abstract. Among the top 6 results, the 1st, 4th, 5th and 6th ones are proper recommendations relevant and new to the manuscript; the 2nd one is the original citation; the 3rd one is a similar but irrelevant recommendation.

We also present an extensive empirical evaluation in the CiteSeerX digital library against many baselines. Our empirical study demonstrates the effectiveness and the scalability of our approach.

The rest of this paper is organized as follows. We discuss related work in Section 2. We formalize the problem in Section 3. We discuss candidate retrieval methods in Section 4. We present our context-aware relevance model for ranking in Section 5. We report our empirical evaluation in Section 6. Section 7 concludes the paper.

2. RELATED WORK

In this section, we discuss the related work on document recommendation, topic-based link prediction, and analysis of citation contexts.

2.1 Document Recommendation Systems

There are some previous efforts on recommending a bibliography list for a manuscript, or recommending papers to reviewers. The existing techniques generally rely on a user profile or a partial list of citations.

Basu et al. [4] focused on recommending conference paper submissions to reviewers based on paper abstracts and reviewer profiles. Reviewer profiles are extracted from the Web. This is a specific step in a more general problem known as the Reviewer Assignment Problem, surveyed by Wang et al. [26]. Chandrasekaran et al. [6] presented a technique to recommend technical papers to readers whose profile information is stored in CiteSeer. A user’s publication records are used to model her profile. User profiles and documents are presented as hierarchical concept trees with predefined concepts from the ACM Computing Classification System. The similarity between a user profile and a document is measured by the weighted tree edit distance. Our work can also be seen as a profile-based system, where a query manuscript is a profile. However, our system uses richer information than just predefined concepts or paper abstracts.

Shaparenko and Joachims [22] proposed a technique based on language modeling and convex optimization to recommend documents. For a large corpus, the k-most similar documents based on cosine similarity are retrieved. However, similarity based on full-text is too slow for large digital libraries. Furthermore, according to our experiments, similarity based on document abstract results in poor recall (cf. Table 1).

Some previous studies recommend citations for a manuscript already containing a partial list of citations. Specifically, given a document \( d \) and its partial citation list \( r' \), those...
studies try to recover the complete citation list denoted by $r \supset r'$. Collaborative filtering techniques have been widely applied. For example, McNee et al. \cite{23} built various rating matrices including an author-citation matrix, a paper-citation matrix, and a co-citation matrix. Papers which are co-cited often with citations in $r'$ are potential candidates. Zhou et al. \cite{27} propagated the positive labels (i.e., the existing citations in $r'$) in multiple graphs such as the paper-paper citation graph, the author-paper bipartite graph, and the paper-venue bipartite graph, and learned the labels of the rest documents for a given testing document in a semi-supervised manner. Torres et al. \cite{25} used a combination of context-based and collaborative filtering algorithms to build a recommendation system, and reported that the hybrid algorithms performed better than individual ones. Strohman et al. \cite{23} experimented with a citation recommendation system where the relevance between two documents is measured by a linear combination of text features and citation graph features. They concluded that similarity between bibliographies and Katz distance \cite{15} are the most important features. Tang and Zhang \cite{24} explored recommending citations for placeholders in a very limited extent. In particular, a user must provide a bibliography with papers relevant to each citation context, as this information is used to compute features in the hidden layer of a Restricted Boltzmann Machine before predictions can be made. We feel this requirement negates the need for requesting recommendations.

In our study, we do not require a partial list of citations, since creating such a list for each placeholder shifts most of the burden on the user. Thus, we can recommend citations both globally (i.e., for the bibliography) and also locally (i.e., for each placeholder, the innovative part of this paper).

### 2.2 Topic-based Citation Link Prediction

Topic models are unsupervised techniques that analyze the text of a large document corpus and provide a low-dimensional representation of documents in terms of automatically discovered and comprehensible "topics". Topic models have been extended to handle citations as well as text, and thus can be used to predict citations for bibliographies. The aforementioned work of Tang and Zhang \cite{24} fits in this framework. Nallapati et al. \cite{18} introduced a model called Pairwise-Link-LDA which models the presence or absence of a link between every pair of documents and thus does not scale to large digital libraries. Nallapati et al. \cite{18} also introduced a simpler model that is similar to the work of Cohn and Hofmann \cite{7} and Erosheva et al. \cite{9}. Here citations are modeled as a sample from a probability distribution associated with a topic. Thus, a paper can be associated with topics when it is viewed as a citation. It can also be associated with topics from the analysis of its text. However, there is no mechanism to enforce consistency between the topics assigned in those two ways.

In general, topic models require a long training process because they are typically trained using iterative techniques such as Gibbs Sampling or variational inference. In addition to this, they must be retrained as new documents are added.

### 2.3 Citation Context Analysis

The prior work on analyzing citation contexts mainly belongs to two groups. The first group tries to understand the motivation functions of an existing citation. For example, Aya et al. \cite{2} built a machine learning algorithm to automatically learn the motivation functions (e.g., compare, contrast, use of the citation, etc.) of citations by using citation context based features. The second group tries to enhance topical similarity between citations. For example, Huang et al. \cite{11} observed that citation contexts can effectively help to avoid "topic drifting" in clustering citations into topics. Ritchie \cite{21} extensively examined the impact of various citation context extraction methods on the performance of information retrieval.

The previous studies clearly indicate that citation contexts are a good summary of the contributions of a cited paper and clearly reflect the information needs (i.e., motivations) of citations. Our work moves one step ahead to recommend citations according to contexts.

### 3. PROBLEM DEFINITION AND ARCHITECTURE

In this section, we formulate the problem of context-based citation recommendation, and discuss the preprocessing step.

#### 3.1 Problem Definition

Let $d$ be a document, and $D$ be a document corpus.

**Definition 3.1.** In a document $d$, a context $c$ is a bag of words. The global context is the title and abstract of $d$. The local context is the text surrounding a citation or placeholder. If document $d_1$ cites document $d_2$, the local context of this citation is called an out-link context with respect to $d_1$ and an in-link context with respect to $d_2$.

A user can submit either a manuscript (i.e., a global context and a set of out-link local contexts) or a few sentences (i.e., an out-link local context) as the query to our system. There are two types of citation recommendation tasks, which happen in different application scenarios.

**Definition 3.2** (Global Recommendation). Given a query manuscript $d$ without a bibliography, a global recommendation is a ranked list of citations in a corpus $D$ that are recommended as candidates for the bibliography of $d$.

Note that different citation contexts in $d$ may express different information needs. The bibliography candidates provided by a global recommendation should collectively satisfy the citation information needs of all out-link local contexts in the query manuscript $d$.

**Definition 3.3** (Local Recommendation). Given an out-link local context $c$, with respect to $d$, a local recommendation is a ranked list of citations in a corpus $D$ that are recommended as candidates for the placeholder associated with $c$.

For local recommendations, the query manuscript $d$ is an optional input and it is not required to already contain a representative bibliography.

To the best of our knowledge, global recommendations are only tackled by document-citation graph methods (e.g., \cite{23}) and topic models (e.g., \cite{24,18}). However, the context-aware approaches have not been considered for global or local recommendations (except in a limited case where a bibliography with papers relevant to each citation context is required as the input).
3.2 Preprocessing

Our proposed context-based citation recommendation system can take two types of inputs, a query manuscript $d_1$ or just a single out-link local context $c_*$. We preprocess the query manuscript $d_1$ by extracting its global context (i.e. the title and abstract) and all of its out-link local contexts. Extracting the local contexts from $d_1$ is not a trivial task. Ritchie [21] conducted extensive experiments to study the impact of different lengths of local citation contexts on information retrieval performance, and concluded that fixed window contexts (e.g. size of 100 words) are simple and reasonably effective. Thus, before removing all stop words, for each placeholder we extract the citation context by taking 50 words before and 50 words after the placeholder. This preprocessing is efficient. For a PDF document of 10 pages and 20 citations, preprocessing takes on average less than 0.1 seconds.

The critical steps of the system are: (1) quickly retrieving a large candidate set which has good coverage over the possible citations, and (2) for each placeholder associated with an out-link local context and for the bibliography, ranking the citations by relevance and returning the top $K$. The next two sections focus on these critical steps.

4. THE CANDIDATE SET

Citation recommendation systems, including our system and especially those that compute intricate graph-based features [23, 27], first quickly retrieve a large candidate set of papers. Then, features are computed for papers in this set, generated by some other method (e.g., GN or Author). We call this method CitHop.

4.1 Context-oblivious Methods

The context-oblivious methods do not consider local contexts in a query manuscript. In a query manuscript $d_1$, for each context $c_*$, we can retrieve:

1. The top $N$ documents with abstract and title most similar to $d_1$. We call this method GN (e.g., $G_{100}$, $G_{1000}$).
2. The documents that share authors with $d_1$. We call this method Author.
3. The papers cited by documents already in a candidate set, generated by some other method (e.g., GN or Author). We call this method AuthHop.
4. The documents written by the authors whose papers are already in a candidate set generated by some other method. We call this method AuthHop.

Note that retrieval based on the similarity between the entire text content of documents would be too slow. Thus, these methods appear to provide a reasonable alternative. However, the body of an academic paper covers many ideas that would not fit in its abstract and so many relevant documents will not be retrieved. Retrieval by author similarity will add many irrelevant papers, especially if the authors have a broad range of research interests and if CitHop or AuthHop is used to further expand the candidate set. Context-aware methods can avoid such problems.

4.2 Context-aware Methods

The context-aware methods help improve coverage for recommendations by considering local contexts in the query manuscript. In a query manuscript $d_1$, for each context $c_*$, we can retrieve:

5. The top $N$ papers whose in-link contexts are most similar to $c_*$. We call this method LN (e.g., $L_{100}$).
6. The papers containing the top-$N$ out-link contexts most similar to $c_*$ (these papers cite papers retrieved by LN). When used in conjunction with LN, we call this method LCN (e.g., $LC_{100}$).

We found that method LCN is needed because frequently a document from a digital library may have an out-link context that describes how it differs from related work (e.g., “prior work required full bibliographies but we do not”). Thus, while an out-link context usually describes a cited paper, sometimes it may also describe the citing paper, and this description may be what best matches an out-link context $c_*$ from a query manuscript.

The above 6 methods can be combined in some obvious ways. For example, $(L_{100}+CitHop)+G_{1000}$ is the candidate set formed by the following process: for each context $c_*$ in a query manuscript $d_1$, add the top 100 documents with in-link local context most similar to $c_*$ (i.e. $L_{100}$) and all of their citations (i.e. CitHop) and then add the top
1000 documents with abstract and title most similar to \( d_1 \) (i.e. G1000).

5. MODELLING CONTEXT-BASED CITATION RELEVANCE

In this section, we propose a non-parametric probabilistic model to measure context-based (and overall) relevance between a manuscript and a candidate citation, for ranking retrieved candidates. To improve scalability, we use an approximate inference technique which yields a closed form solution. Our model is general and simple so that it can be used to efficiently and effectively measure the similarity between any two documents with respect to certain contexts or concepts in information retrieval.

5.1 Context-Based Relevance Model

Recall that a query manuscript \( d_1 \) can have a global context \( c_1 \) (e.g., a title and abstract, which describes the problem to be addressed) and the out-link local contexts \( c_2, \ldots, c_k \) (e.g., text surrounding citations) which compactly express ideas that may be present in related work. A document \( d_2 \) in an existing corpus \( D \) has a global context \( b_1 \) (e.g., title and abstract), and local in-link contexts \( b_1, \ldots, b_k \) (e.g., text that is used by papers citing \( d_2 \)) which compactly express ideas covered in \( d_2 \).

In this section, we describe a principled and consistent way of measuring \( \text{sim}(d_1, d_2) \), defined as the overall relevance of \( d_2 \) to \( d_1 \) and \( \text{sim}(d_1, d_2; c_i) \), defined as the relevance of \( d_2 \) to \( d_1 \) with respect to a specific context \( c_i \) (in particular, \( c_1 \) could be any of the out-link contexts \( c_1 \)). Our techniques are based on Gleason’s Theorem specialized to finite dimensional real vector spaces.

**Theorem 5.1 (Gleason [10]).** For an \( n \)-dimensional real vector space \( V \) (with \( n \geq 3 \)), let \( p \) be a function that assigns a number in \( [0,1] \) to each subspace of \( V \) such that \( p(v_1 \oplus v_2) = p(v_1) + p(v_2) \) whenever \( v_1 \) and \( v_2 \) are orthogonal subspaces \(^1 \) and \( p(V) = 1 \). Then \( p(v) = \text{Trace}(P_v) \) where \( P_v \) is the projection matrix for subspace \( v \) (e.g. \( P_{v\oplus w} \) is the projection of \( v \) onto the subspace \( v \)), and \( T \) is a density matrix – a symmetric positive semidefinite matrix whose trace is 1.

Note that van Rijsbergen [20] proposed using Gleason’s Theorem as a model for information retrieval, and this was also extensively studied by Melucci [17]. However, our framework is substantially different from their proposals since it relies on comparisons between density matrices.

Let \( W \) be the set of all words in our corpus and let \( |W| \) be the number of words. The vector space \( V \) is \( |W| \)-dimensional with one dimension for each word.

In this framework, atomic concepts will be represented as one-dimensional vector spaces and we will treat each context (global, in-link, out-link) as an atomic concept. Each atomic concept \( c \) will be associated a unit column vector which we shall also denote by \( c \) (one such representation can be derived from normalizing tf-idf scores into a unit vector). The projection matrix for \( c \) is then \( cc^T \). Our goal is to measure the probability that \( c \) is relevant to a document \( d \) and so, by Gleason’s Theorem, each document \( d \) is associated with a density matrix \( T_d \). The probability that \( c \) is relevant to \( d \) is then \( p_d(c) = \text{Trace}(T_dcc^T) = c^T T_d c \). Note that similar atomic concepts (as measured by the dot product) will have similar relevance scores.

Now, the probability distribution characterized by Gleason’s theorem is not a generative distribution – it cannot be used to sample concepts. Instead, it is a distribution over yes/no answers (e.g. is concept \( c \) relevant or not?). Thus to estimate \( T_d \) for a document \( d \) we will need some (unknown) generative distribution \( p_{gen} \) over unit vectors (concepts). Our evidence about the properties of \( T_d \) come from the following process. \( p_{gen} \) independently generates \( k \) concepts \( c_1, \ldots, c_k \) and these concepts are then independently judged to be relevant to \( d \). This happens with probability \( \prod_{i=1}^{k} p_{gen}(c_i) p_d(c_i) \). We seek to find a density matrix \( T_d \) that maximizes the likelihood. The log likelihood is:

\[
\sum_{i=1}^{k} \ln p_{gen}(c_i) + \sum_{i=1}^{k} \ln \left( c_i^T T_d c_i \right).
\]

Now, if there is only one concept (i.e. \( k = 1 \)), then the maximum likelihood estimator is easy to compute: it is \( T_d = c_1c_1^T \). In the general case, however, numerical methods are needed [3]. The computational cost of estimating \( T_d \) can be seen from the following proposition:

**Proposition 5.2.** The density matrix \( T_d \) can be represented as \( \sum_{i=1}^{r} v_i t_i^T \) where the \( t_i \) are a set of at most \( r \) orthogonal column vectors with \( \sum (t_i \cdot t_i) = 1 \), and \( r \) is the dimension of the space spanned by the \( c_i \) (the number of linearly independent contexts). There are \( O(r^2) \) parameters to determine and numerical (iterative) techniques will scale as a polynomial in \( r^2 \).

The detailed proof is in Appendix A.

Furthermore, the addition of new documents to the corpus will cause the addition of new in-link contexts, requiring a recomputation of all the \( T_d \). Thus the likelihood approach will not scale for our system.

5.2 Scalable Closed Form Solutions

Since hundreds of thousands of density matrices need to be estimated (one for each document) and recomputed as new documents are added to the corpus, we opt to replace the exact but slow maximum likelihood computation with an approximate but fast closed form solution which we derive in this section.

We begin with the following observation. For each concept \( c_i \), the maximum likelihood estimate of \( T_d \) given that concept \( c_i \) is relevant is \( c_i c_i^T \). It stands to reason that our overall estimate of \( T_d \) should be similar to each of the \( c_i c_i^T \).

We will measure similarity by the Frobenius norm (square-root of the sum of the squared matrix entries) and thus set up the following optimization problem:

\[
\text{minimize } L(T_d) = \sum_{i=1}^{k} ||T_d - c_i c_i^T||_F^2,
\]

subject to the constraint that \( T_d \) is a density matrix. Taking derivatives, we get

\[
\frac{\partial L}{\partial T_d} = 2 \sum_{i=1}^{k} (T_d - c_i c_i^T) = 0,
\]

leading to the solution \( T_d = \frac{1}{k} \sum_{i=1}^{k} c_i c_i^T \). Now that we have a closed form estimate for \( T_d \) we can discuss how to measure the relevance between documents with respect to a concept.
Global Recommendation

Let $T_{d_1}$ and $T_{d_2}$ be the respective density matrices of the manuscript $d_1$ and a document $d_2$ from the corpus $D$. We define $\text{sim}(d_1, d_2)$, the overall relevance of $d_2$ to $d_1$, to be the probability that a random context drawn from the uniform distribution over contexts is relevant to both $d_1$ and $d_2$. After much mathematical manipulation, we get the following:

**Proposition 5.3.** Let $T_{d_1} = \frac{1}{k_1} \sum_{i=1}^{k_1} c_i c_i^T$ and $T_{d_2} = \frac{1}{k_2} \sum_{i=1}^{k_2} b_i b_i^T$. Then the relevance of $d_2$ to $d_1$ is:

$$\text{sim}(d_1, d_2) = \frac{1}{k_1 k_2} \sum_{i=1}^{k_1} \sum_{j=1}^{k_2} (c_i \cdot b_j)^2.$$  (1)

The detailed proof is given in Appendix B. Given query manuscript $q_1$, we use Equation 1 to rank documents for global recommendation.

Local Recommendation

If we are given a single context $c_*$ instead of a manuscript, we can still compute $\text{sim}(c_*, d_2)$, the relevance of a document $d_2 \in D$ to the context $c_*$. Letting $T_{d_2} = \frac{1}{k_2} \sum_{i=1}^{k_2} b_i b_i^T$, then by Gleason’s Theorem, we have:

$$\text{sim}(c_*, d_2) = \text{Trace}(T_{d_2} c_* c_*^T) = \frac{1}{k_2} \sum_{j=1}^{k_2} (b_j \cdot c_*^2).$$  (2)

We define $\text{sim}(d_1, d_2; c_*)$, the relevance of $d_2$ to $d_1$ with respect to context $c_*$ as the probability that $c_*$ is relevant to both documents. Applying Gleason’s Theorem twice:

$$\text{sim}(d_1, d_2; c_*) = \text{Trace}(T_{d_2} c_* c_*^T) \text{Trace}(T_{d_2} c_* c_*^T) = \frac{1}{k_1 k_2} \sum_{i=1}^{k_1} \sum_{j=1}^{k_2} (c_i \cdot b_j)^2.$$  (3)

Given query manuscript $d_1$, we use Equation 3 to rank documents for recommendations at the placeholder associated with context $c_*$ in $d_1$. If a citation context $c_*$ is given without $d_1$, then we use Equation 2.

6. EXPERIMENTS

We built a real system in the CiteSeerX staging server to evaluate context-aware citation recommendations. We used all research papers published and crawled before year 2008 as the document corpus $D$. After removing duplicate papers and the papers missing abstract/title and citation contexts, we obtained 456,787 unique documents in the corpus. For each paper, we extracted its title and abstract as the cultural context or text content. Within a paper, we simply took 50 words before and after each citation as its local citation context. We removed some popular stop words. In order to preserve the time-sensitive past/present/future tenses of verbs and the singular/plural styles of named entities, no stemming was done. All words were transferred to lower-cases. Finally, we obtained 1,810,917 unique local citation contexts and 716,927 unique word terms. We used all 1,612 papers published and crawled in early 2008 as the testing data set.

We implemented all algorithms in C++ and integrated them into the Java running environment of CiteSeerX. All experiments were conducted on a Linux cluster with 128 nodes\(^2\), each of which has 8 CPU processors of 2.67GHz and 32G main memory.

6.1 Performance Measures

The performance of recommendation can be measured by a wide range of metrics and means, including user studies and click-through monitoring. For experimental purpose, we focus on four performance measures as follows in this paper.

Recall (R): We removed original citations from the testing documents. The recall is defined as the percentage of original citations that appear in the top $K$ recommended citations. Furthermore, we categorize recall into global recall and local recall for global and local recommendation respectively. The global recall is first computed as the percentage of original bibliography of each testing document $d$ that appears in the top $K$ recommended citations of $d$, and then averaged over all $1,162$ testing documents. The local recall is first computed as, for each citation placeholder, the percentage of the original citations cited by $c_*$ that appear in the top $K$ recommended citations of $c_*$, and then averaged over all $7,049$ testing out-link citation contexts.

Co-cited probability (P): We may recommend some relevant or even better recommendations other than those original ones among the top $K$ results, which cannot be captured by the traditional metric like precision. The previous work usually conducted user studies for this kind of relevance evaluation [16, 6]. In this paper, we instead use the wisdom of the population as the ground truth to define a systematic metric. For each pair of documents $(d_1, d_2)$ where $d_1$ is an original citation and the $d_2$ is a recommended one, we calculate the probability that these two documents have been co-cited by the popularity in the past as

$$P = \frac{\text{number of papers citing both } d_1 \text{ and } d_2}{\text{number of papers citing } d_1 \text{ or } d_2}.$$  (4)

The co-cited probability is then averaged over all $K \cdot l$ unique document pairs for the top $K$ results, where $l$ is the number of original citations. Again, we categorize this probability into a global version and a local version: the former is averaged over all testing documents and the latter is averaged over all testing citation contexts.

NDCG: The effectiveness of a recommendation system is also sensitive to the positions of relevant citations, which cannot be evaluated by recall and co-cited probability. Intuitively, it is desirable that highly relevant citations appear earlier in the top $K$ list. We use normalized discounted cumulative gain (NDCG) to measure the ranked recommendation list. The NDCG value of a ranking list at position $i$ is calculated as

$$NDCG@i = \frac{\sum_{j=1}^{i} 2^r(j) - 1}{\log(1 + j)},$$

where $r(j)$ is the rating of the $j$-th document in the ranking list, and the normalization constant $Z_i$ is chosen so that the perfect list gets a NDCG score of 1. Given a testing document $d_1$ and any other document $d_2$ from our corpus $D$, we use the average co-cited probability of $d_2$ with all original citations of $d_1$ to weigh the citation relevance score of $d_2$ to $d_1$. Then, we sort all $d_2$ w.r.t. this score (suppose $P_{max}$ is the highest score) and define 5-scale relevance number for them as the ground truth: 4, 3, 2, 1, 0 for documents

\(^2\)Due to the PSU policy, we could only use 8 of them.
Table 1: Compare different candidate sets. Numbers are averaged over 1,612 test documents.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Coverage</th>
<th>Candidate Set Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1000</td>
<td>0.84</td>
<td>5,000</td>
</tr>
<tr>
<td>L1000</td>
<td>0.55</td>
<td>341</td>
</tr>
<tr>
<td>LC1000</td>
<td>0.63</td>
<td>674</td>
</tr>
<tr>
<td>L1000</td>
<td>0.69</td>
<td>2,844</td>
</tr>
<tr>
<td>LC1000</td>
<td>0.78</td>
<td>5,692</td>
</tr>
<tr>
<td>Author</td>
<td>0.05</td>
<td>17</td>
</tr>
<tr>
<td>L1000+CitHop</td>
<td>0.61</td>
<td>1,327</td>
</tr>
<tr>
<td>L1000+CitHop</td>
<td>0.72</td>
<td>9,049</td>
</tr>
<tr>
<td>LC1000+G1000</td>
<td>0.73</td>
<td>3,561</td>
</tr>
<tr>
<td>G1000+CitHop</td>
<td>0.73</td>
<td>3,790</td>
</tr>
<tr>
<td>LC1000+CitHop</td>
<td>0.83</td>
<td>22,629</td>
</tr>
<tr>
<td>Author+CitHop</td>
<td>0.15</td>
<td>63</td>
</tr>
<tr>
<td>L1000+G1000</td>
<td>0.75</td>
<td>1,312</td>
</tr>
<tr>
<td>LC1000+G1000</td>
<td>0.79</td>
<td>1,618</td>
</tr>
<tr>
<td>(L1000+CitHop)+G1000</td>
<td>0.79</td>
<td>2,279</td>
</tr>
<tr>
<td>(LC1000+CitHop)+G1000</td>
<td>0.85</td>
<td>4,460</td>
</tr>
<tr>
<td>(LC1000+G1000)+CitHop</td>
<td>0.92</td>
<td>24,793</td>
</tr>
<tr>
<td>(LC1000+G1000)+AuthorCitHop</td>
<td>0.79</td>
<td>1,674</td>
</tr>
<tr>
<td>(LC1000+G1000)+AuthHop</td>
<td>0.88</td>
<td>39,496</td>
</tr>
</tbody>
</table>

We compare CRM with 7 other context-oblivious baselines: HITs [13]: the candidates are ranked w.r.t. their authority scores in the candidate set subgraph. We choose to compare with HITs because it is interesting to see the difference between the popularity (link-based methods) and the relevance (context or document based methods).

Katz [15]: the candidates are ranked w.r.t. the Katz distance \( \sum \beta^i N_i \), where \( N_i \) is the number of unique paths of length \( i \) between the query manuscript/context and the candidate and the path should go through the top \( N \) similar documents/contexts of the query manuscript/context, and \( \beta \) is a decay parameter between 0 and 1. We choose to compare Katz because this feature has been shown to be the most effective among all text/citation features in [23] for document-based global recommendation. Note that [23] used the ground truth to calculate the Katz distance, which is impractical.

l-count and g-count: the candidates are ranked according to the number of citations in the candidate set subgraph (l-count) or the whole corpus (g-count).

textsim: the candidates are ranked according to similarity with the query manuscript using only title and abstract. This allows us to see the benefit of a context-aware approach.

diffusion: the candidates are ranked according to their topical similarities which are generated by the multinomial diffusion kernel [14] to the query manuscript, \( K(\theta_1, \theta_2) = \left( \frac{4\pi t}{\theta_2} \right)^{\frac{1}{2}} \exp \left( -\frac{1}{2} \arccos^2 \left( \frac{\sqrt{\theta_1} \cdot \sqrt{\theta_2}}{\sqrt{\theta_1 + \theta_2}} \right) \right) \), where \( \theta_1 \) and \( \theta_2 \) are topic distributions of the query manuscript \( d_1 \) and the candidate \( d_2 \) respectively, \( t \) is the decay factor. Since we only care about the ranking, we can ignore the first item and \( t \).

We choose to compare with it because topic-based citation recommendation [24] is one of related work and the multinomial diffusion kernel is the state-of-the-art tool in topic-based text similarity [8]. We run LDA [5] on each candidate set online (by setting the number of topics as 60) to get the topic distributions for documents.

mix-features: the candidates are ranked according to the weighted linear combination of the above 6 features. We choose to compare it because in [23], a mixture approach considering both text-based features and citation-based features is the most effective.

Figures 5 (a), (b) and (c) illustrate their performances on recall, co-cited probability and NDCG, respectively.

Among all methods, g-count is the worst one simply because it is measured over the whole corpus, not the candidate set. Context-oblivious content-based methods like textsim and diffusion come to heel, indicating that abstract/title only are too sparse to portray the specific citation functions well. Moreover, they cannot find the proper related work that uses different conceptual words. The diffusion is better than textsim, indicating that topic-based similarity can capture the citation relations more than raw text similarity. The social phenomenon that the rich get richer is also common in the citation graph, since the citation features including l-count, HITs, and Katz work better than the abstract/title-based text features. Interestingly, [23] claimed that citation features did a poor job at coverage. But on our data, they have higher recall values than text features. A combination of these features (mix-features) can further improve the performance, especially on the recall and NDCG, which means that if a candidate is recommended by multiple features, it

in \( (3P_{max}/4, P_{max}], (P_{max}/2, 3P_{max}/4], (P_{max}/4, P_{max}/2], (0, P_{max}/4] \) and 0 respectively. Finally, the NDCG over all testing documents (the global version) or all testing citation contexts (the local version) is averaged to yield a single qualitative metric for each recommendation problem.

Time: We use the running time as an important metric to measure the efficiency of the recommendation approaches.

6.2 Retrieving Candidate Sets

Table 1 evaluates the quality of different candidate set retrieval techniques (see Section 4 for notation and detailed descriptions). Here we measure coverage (the average recall of the candidate set with respect to the true bibliography of a testing document) and candidate set size. A good candidate set should have high recall and a small size since large candidate sets slow down the final ranking for all recommendation systems. For context-aware methods, we feel LC1000+G1000 achieves the best tradeoff between candidate set size and coverage. For context-oblivious methods, G1000+CitHop works reasonably well (but not as well as LC1000+G1000). However, the retrieval time of context-oblivious methods is around 0.28 seconds on an 8-node cluster. On the other hand, the retrieval time of context-aware methods ranges from 2 to 10 seconds on the same cluster (depending on the number of out-link contexts\(^3\) of a query manuscript). Our goal for the final system is to use LC1000+G1000 and to speed up retrieval time using a combination of indexing tricks and more machines (since this retrieval is highly parallelizable). Note, however, that ranking techniques are orthogonal to candidate set retrieval methods. Thus, when we compare our ranking methods for recommendations with baselines and related work, we will use LC1000+G1000 as the common candidate set, since our eventual goal is to use this retrieval method in our system.

6.3 Global Recommendation

In this section, we compare our context-aware relevance model (CRM for short, Section 5) with other baselines in global recommendation performance, since the related work only focused on recommending the bibliography.

\(^3\)Due to OCR and crawling issues, there were an average of 5 out-link contexts per testing document.
should be ranked higher. An interesting finding is that the performance of HITs (especially on NDCG) increases significantly after more candidates are returned, indicating that some candidates with moderate authoritative scores are our targets (people may stop citing the most well-known papers after they become the standard techniques in their domains and shift the attentions to other recent good papers). Katz works the best among single features, partially because it implicitly combines the text similarity with the citation count. It works like the collaborative filtering, where candidates often cited by similar documents are recommended. Finally, our CRM method leads the pack on all three metrics, implying that after considering all historical in-link citation contexts of candidates (then the problem of using different conceptual words for the same concept would be alleviated), CRM is effective in recommending bibliography.

**Ranking Time**

Time is not a major issue for most ranking algorithms except for topic-based methods, where LDA usually took tens of seconds (50 \sim 100) for each new candidate set. Thus, topic-based methods including diffusion and mix-features are not suitable for online recommendation systems. All other ranking algorithms need less than 0.1 seconds. Limited by space, the detailed comparisons are omitted here.

### 6.4 Local Recommendation

Local recommendation is a novel task proposed by our context-aware citation recommendation system. Here, we do not compare with the above baselines because they are not tailored for local recommendation. Instead, we evaluate the impact of the absence/presence of the global context and other local contexts, and analyze the problem of context-aware methods.

If a user only inputs a bag of words as the single context to request recommendations, we can then only use Equation 2 to rank candidates. We name this method as CRM-singlecontext. If a user inputs a manuscript with placeholders inside, we can then rank candidates for each placeholder using Equation 3. We name this method as CRM-crosscontext. Figure 6 illustrates the performance of these two kinds of local recommendations on recall, co-cited probability and NDCG.

CRM-crosscontext is rather effective. In Figure 5 (due to crawling and OCR issues, our testing documents have an average of 5 out-link citation contexts that point to documents in our corpus, so “top 5 per context” corresponds to “top 25 per document”), the performance of CRM-crosscontext is very close to CRM in global recommendation. We know CRM-crosscontext and CRM have the same input (the same amount of information to use). However, CRM-crosscontext tackles a much harder problem than CRM, and has to assign citations to each placeholder. Thus, CRM-crosscontext is more capable than CRM. CRM-crosscontext outperforms all baselines of global recommendation and thus is also superior than their local versions. Limited by space, we omit the details here.

CRM-crosscontext is able to effectively rank candidates for placeholders. For example, more than 42% original citations can be found in the top 5 recommendations, and frequently co-cited papers (w.r.t. original citations) also appear early as indicated by NDCG. On the other hand, CRM-singlecontext uses much less information (without global context and other coupling contexts) but still achieves reasonable performance. For example, if a user only inputs 100 words as the query, around 34% original citations can be found in the top 5 list.

One may wonder what would happen to some documents in the corpus which do not have enough in-link citation contexts in history. Given an original citation from a query manuscript, we examine the correlation of its missing/hit probability to its number of historical in-link citation contexts. Figure 7.
The correlation results clearly indicate that the missing/hit probability of an original citation declines/raises proportionally to its number of historical in-link contexts. In fact, for those new corpus documents without any in-link contexts, our context-aware methods can still conduct context-oblivious document-based recommendation, except that we still enhance the specific citation motivations of a query manuscript using its out-link local contexts.

7. CONCLUSIONS

In this paper, we tackled the novel problem of context-aware citation recommendation, and built a context-aware citation recommendation prototype in CiteSeerX. Our system is capable of recommending the bibliography to a manuscript and providing a ranked set of citations to a specific citation placeholder. We developed a mathematically sound context-aware relevance model. A non-parametric probabilistic model as well as its scalable closed form solutions are then introduced. We also conducted extensive experiments to examine the performance of our approach.

In the future, we plan to make our citation recommendation system publicly available. Moreover, we plan to develop scalable techniques for integrating various sources of features, and explore semi-supervised learning on the partial list of citations in manuscripts.

8. ACKNOWLEDGEMENT

This work is supported in part by the National Science Foundation Grants 0535656, 0845487, and 0454052, an NSERC Discovery grant and an NSERC Discovery Accelerator Supplement grant. All opinions, findings, conclusions and recommendations in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

APPENDIX

A. PROOF OF PROPOSITION 5.2

By the spectral theorem, we can choose all the \( t_i \) to be orthogonal. Each \( t_i \) can also be expressed as \( \alpha_1 s_1 + \cdots + \alpha_\eta s_\eta + \beta_1 u_1 + \cdots + \beta_\nu u_\nu \) where the \( s_i \) and \( u_j \) are orthogonal unit vectors such that the \( s_i \) span the same space as the \( c_i \). The log likelihood is then

\[
\sum_{i=1}^{k} \log p_{gen}(c_i) + \sum_{i=1}^{k} \log p_d(c_i)
= \sum_{i=1}^{k} \log p_{gen}(c_i) + \sum_{i=1}^{k} \log \sum_{j=1}^{\eta} c_i^T c_i \delta_j^T \delta_j = \sum_{i=1}^{k} \log p_{gen}(c_i) + \sum_{i=1}^{k} \log \sum_{j=1}^{\eta} (\alpha_1 s_1 + \cdots + \alpha_\eta s_\eta + \beta_1 u_1 + \cdots + \beta_\nu u_\nu) \cdot c_i^2
= \sum_{i=1}^{k} \log p_{gen}(c_i) + \sum_{i=1}^{k} \log \sum_{j=1}^{\eta} (\alpha_1 s_1 + \cdots + \alpha_\eta s_\eta + \beta_1 u_1 + \cdots + \beta_\nu u_\nu) \cdot c_i^2.
\]

Since the \( u_j \) are orthogonal to the \( c_i \), we can increase the likelihood by replacing each \( t_i = \alpha_1 s_1 + \cdots + \alpha_\eta s_\eta + \beta_1 u_1 + \cdots + \beta_\nu u_\nu \) with \( t_i' = \alpha_1 s_1 + \cdots + \alpha_\eta s_\eta \) to get the matrix \( T_d = \sum_{i=1}^{k} t_i' t_i'^T \) without changing the likelihood. However,

\[
Trace(T_d) = \sum_{i=1}^{k} t_i' \cdot t_i' = \sum_{i=1}^{k} (\alpha_1^2 + \cdots + \alpha_\eta^2) \text{ since the } \alpha_i \text{ and } \beta_j \text{ are all orthogonal}
\leq \sum_{i=1}^{k} (\alpha_1^2 + \cdots + \alpha_\eta^2 + \beta_1^2 + \cdots + \beta_\nu^2) = Trace(T_d) = 1.
\]

Thus we need to multiply each \( t_i' \) by a constant \( \gamma > 1 \) to increase the trace of \( T_d \). Multiplication by such a \( \gamma \) will then increase each of the terms in the likelihood function, thus \( \gamma T_d \) would have a higher log likelihood than \( T_d \). Thus in the maximum likelihood solution \( T_d = \sum_{i=1}^{k} t_i' t_i'^T \), each \( t_i' \) is in the subspace spanned by the contexts \( c_i \) and representing all of the \( t_i \) requires \( O(r^2) \) parameters. So, a numerical solution will have polynomial complexity in \( r^2 \).

B. PROOF OF PROPOSITION 5.3

Recall that we treat concepts as one-dimensional vector spaces and thus can represent them as unit vectors. Let \( p \) be the uniform distribution over unit vectors. It is well-known that sampling from \( p \) is equivalent to sampling \( |W| \) independent standard Gaussian random variables (one for each dimension) and dividing them by the square root of their sum of squares. The similarity between \( d_1 \) and \( d_2 \) is:

\[
sim(d_1; d_2) = \int \text{Trace}(T_{d_1} w w^T) \text{Trace}(T_{d_2} w w^T) p(w) dw
= \frac{1}{|W|^2} \int \frac{1}{|W|} \sum_{i=1}^{k} \sum_{j=1}^{k} (c_i \cdot w)^2 (b_j \cdot w) p(w) dw
= \frac{1}{|W|^2} E \left[ \frac{1}{|W|} \sum_{i=1}^{k} \sum_{j=1}^{k} \left( \sum_{a \in W} c_i^a b_j^a (c_i \cdot w)^2 + \sum_{a \in W} \sum_{\gamma \neq \eta} c_i^a c_\gamma^a w \cdot b_j^\gamma w \right) \right].
\]

Now, if all coordinates of \( w \) other than \( w_1 \) (for some \( i \)) are fixed, it is easy to see that the probability of \( w_1 = x \) and \( w_1 = x \) are the same, and so the coordinates of \( w \) are jointly uncorrelated. This means that all terms with an odd power of some coordinate will become 0 in the expected value. Thus \( \text{sim}(d_1; d_2) \) simplifies to:

\[
\text{sim}(d_1; d_2) = \frac{1}{|W|^2} \sum_{i=1}^{k} \sum_{j=1}^{k} \left( E \left( \sum_{a \in W} c_i^a b_j^a w_a^2 + \sum_{\gamma \neq \eta} \sum_{a \in W} c_i^a b_j^\gamma w_a w_\gamma \right) + 4 \sum_{b \in W} \sum_{\gamma \neq \eta} c_i^b c_\gamma b_j^\gamma w_b w_\gamma \right).
\]

To compute the necessary moments, we change basis from Cartesian coordinates to hyperspherical coordinates as:

\[
w_1 = r \cos \theta_1, \ w_2 = r \sin \theta_1 \cos \theta_2, \ w_3 = r \sin \theta_1 \sin \theta_2 \cos \theta_3, \ldots, w_{|W|-1} = r \sin \theta_1 \sin \theta_2 \ldots \sin \theta_{|W|-2} \cos \theta_{|W|-1}, \ w_{|W|} = r \sin \theta_1 \sin \theta_2 \ldots \sin \theta_{|W|-2} \sin \theta_{|W|-1};
\]

\[
r = \sqrt{\sum_{i=1}^{k} w_i^2} \geq 0;
\]

\[
\theta_1 \in [0, \pi], \ \theta_2 \in [0, \pi], \ldots, \theta_{|W|-2} \in [0, \pi], \ \theta_{|W|-1} \in [0, 2\pi);
\]

the absolute value of the determinant of the Jacobian is

\[
r_{|W|-1} \prod_{i=1}^{k} \sin^{w_i-1-i}(\theta_i).
\]

Let us also define

\[
\Phi \equiv \frac{\sum_{i=1}^{k} \sum_{j=1}^{k} \sin^{w_i-1-i}(\theta_i) \prod_{i=1}^{k} \sin^{w_j-1-j}(\theta_j)}{4|W|^2}.
\]

The moment calculations are thus as follows:

\[
E[w_i^4] = \frac{1}{|W|^2} \int \frac{1}{|W|} \sum_{i=1}^{k} \sum_{j=1}^{k} (c_i \cdot w)^2 (b_j \cdot w) p(w) dw
= \frac{1}{|W|^2} \sum_{i=1}^{k} \sum_{j=1}^{k} \left( \sum_{a \in W} c_i^a b_j^a (c_i \cdot w)^2 + \sum_{a \in W} \sum_{\gamma \neq \eta} c_i^a c_\gamma^a w \cdot b_j^\gamma w \right).
\]

by Wallis’s formula [12] where \( n! = 2^{n-1} (n-1)!! \) is the double factorial (1 if \( n = 0 \) or 1; \( n! = n \times (n-2)!! \) otherwise. Also, \( E[w_i^4 w_j^4] = \frac{1}{|W|^2} \int \frac{1}{|W|} \sum_{i=1}^{k} \sum_{j=1}^{k} (c_i \cdot w)^2 (b_j \cdot w) p(w) dw \) is as above.

Thus we have \( E[w_i^4] = 3E[w_i^4 w_j^4] \). Thus \( \text{sim}(d_1; d_2) \) is proportional to \( E[w_i^4 w_j^4] \), a universal constant. We drop this constant to simplify calculations. Thus our similarity is:

\[
\text{sim}(d_1; d_2) = \frac{1}{|W|^2} \sum_{i=1}^{k} \sum_{j=1}^{k} \left( \sum_{a \in W} c_i^a b_j^a + \sum_{a \in W} c_i^a b_j^a w_a^2 + 4 \sum_{b \in W} c_i^b c_\gamma^a b_j^\gamma w_b w_\gamma \right).
\]

Finally, after removing the additive and multiplicative constants (they don’t affect ranking), we can use the following equivalent formula:

\[
\text{sim}(d_1; d_2) = \frac{1}{|W|^2} \sum_{i=1}^{k} \sum_{j=1}^{k} \left( c_i \cdot b_j \right)^2 + 1, \text{ since the } c_i \text{ and } b_j \text{ are unit vectors}.
\]

430