RefSeer: A Citation Recommendation System

Wenyi Huang
harrywy@gmail.com

Zhaohui Wu
laowuz@gmail.com

Prasenjit Mitra
pmitra@ist.psu.edu

C. Lee Giles
giles@ist.psu.edu

Information Sciences and Technology
The Pennsylvania State University
University Park, PA 16802

ABSTRACT

Citations are important in academic dissemination. To help researchers check the completeness of citations while authoring a paper, we introduce a citation recommendation system called RefSeer. Researchers can use this system while authoring papers to find related works to cite. It can also be used by reviewers to check the completeness of a paper's references. RefSeer presents both topic-based global recommendations and citation-context-based local recommendations. By evaluating the quality of recommendations, we show that such a recommendation system can recommend citations with good precision and recall. We also show that our recommendation system is very efficient and scalable.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords

Citation recommendation, RefSeer

1. INTRODUCTION

Citations are important in academic dissemination. Proper citations not only give credit to the work of others but also make it possible for readers to evaluate whether the cited works support the authors' claims. The literature review process usually starts with retrieving relevant documents, based upon a certain pre-selected set of keywords, from search engines such as Google Scholar[1], CiteSeer [1] or Microsoft Academic Search[2]. Researchers then have to go through the documents manually to find works that need to be cited. Such a process is a difficult task for both the junior and experienced researchers for two reasons: (1) the tremendous growth in the number of research articles in the past decade, and (2) introduction of new terminology as the science progresses and new knowledge accumulates.

[1] http://scholar.google.com/
[2] http://academic.research.microsoft.com/

978-1-4799-5569-5/14/$31.00 ©2014 IEEE.

To help researchers check the completeness of citations while authoring a paper, we introduce RefSeer, a citation recommendation system which automatically suggests candidate citations based on input queries. RefSeer has applications for both researchers and reviewers. While authoring a paper, researchers can use our citation recommendation system to find prior works related to the problem they seek to investigate. In turn, reviewers can use RefSeer to check whether a paper cites all relevant papers

Most current literature search engines concentrate on short queries. However, when dealing with long queries, keyword-matching-based search engines perform far from satisfactorily. Even for a keyphrase that appears in citation context, traditional literature search engines fail to retrieve and rank proper papers that need to be cited. For example, given the query "translation model", one may want to cite the first paper introducing the statistical machine translation models [2]. However, the keyphrase did not appear in the title or abstract of that paper. As a result, both Google Scholar and Microsoft Academic Search failed to retrieve this paper in the first page (top 10 results). Instead, Google Scholar and Microsoft Academic Search put [3] as the first result, which contains the keyphrase in the title. CiteSeerx put [4] as the first result while ranking the paper [2] third.

Unlike these traditional literature search engines, RefSeer is designed to deal with long queries. Given queries ranging from a sentence to an entire manuscript, RefSeer presents both topic-based global citation recommendation and also citation-context-based local citation recommendation. Figure 1 shows an example of the RefSeer interface. RefSeer was built using all paper metadata provided by CiteSeer [1]. The recommendation models were trained using all papers in the CiteSeer repository.

2. EXISTING SYSTEMS

The prototype system of RefSeer was built by He, et al. using the model proposed in [5]. This system was built on a cluster of 8 nodes with each node having 8 2.57GHz CPUs and 32GB memory. The system first retrieves a list of papers as a candidate list and then reranks the candidate list based on concept similarity. The complexity of generating the candidate list is linear to the size of the CiteSeer repository, and the computational complexity of reranking is $O(n^2)$ (where n is the number of concepts). Thus, the prototype RefSeer is not efficient and scalable for public uses.

TheAdvisor [6] is a recommendation system which takes a partial list of the bibliography as the input query. TheAdvisor expends the partial list of the bibliography using citation

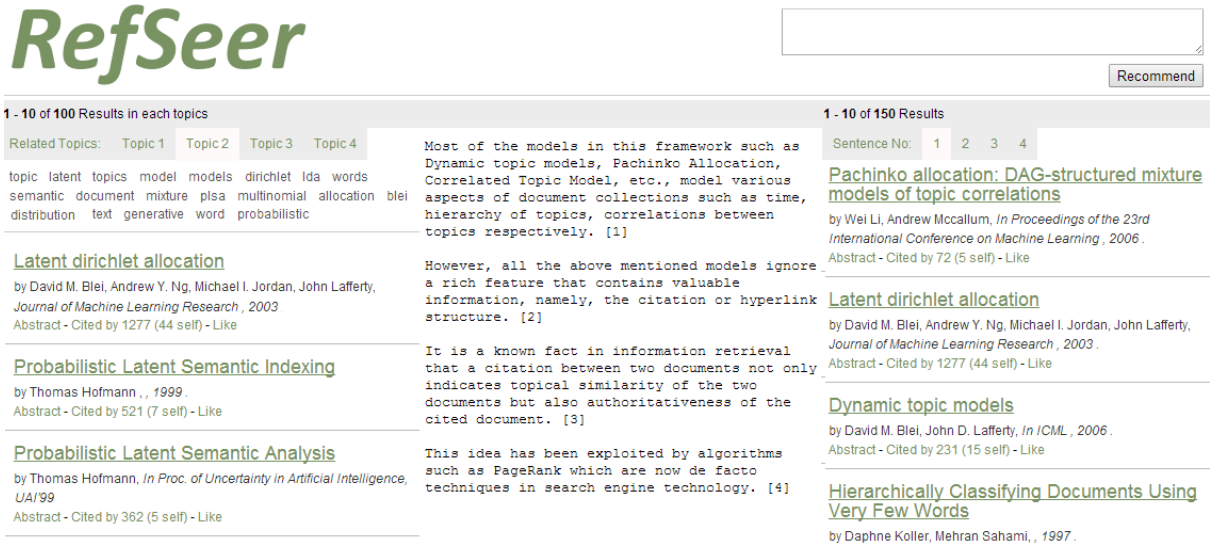

Figure 1: A screenshot of the RefSeer interface. The left column shows topic-based global recommendations with respect to the whole query. The middle column shows the input queries. The right column shows citation-context-based local recommendations with respect to each sentence.

graph information. It can also suggest venues for paper submission and potential reviewers for the paper. Since the recommendation model behind TheAdvisor uses the implementations of sparse matrix computation, the computational expense is relatively high. For each input query, the average execution time is around 1 second on a 50-nodes cluster, in which each node has a 2.4GHz CPU and 4GB memory. Such a system is sufficient for personal use; however, it is not very scalable as a public service.

3. THE REFSEER SYSTEM

3.1 Metadata

RefSeer uses all paper metadata provided by CiteSeer [1], which contains $1,017,457$ papers as of Oct. 2013. Papers' contents are parsed and used for training topic-based model [7]. Citations are extracted with a total number of $10,760,318$ citation links in the citation graph.

We also extracted the sentence where a citation is made. In addition, the sentences that appear immediately before and after the citation sentence are also extracted. These three sentences are then defined together as a citation context for a particular citation. The citation context table consists of $83,598,304$ million rows. Each row is a triplet entries: <Citing paper ID, Cited paper ID, Citation Context>.

3.2 Infrastructure

Figure 2 shows the infrastructure of the RefSeer System. The information flow is divided into two parts after query preprocessing. Global recommendations and local recommendations will be individually calculated. After the recommendation results are generated, the topic-based global recommendation results will be used to filter out irrelevant results from local recommendations.

3.3 Query Preprocessing

Given a query, before sending it to recommendation models, we use a sentence parser to split the query into a list

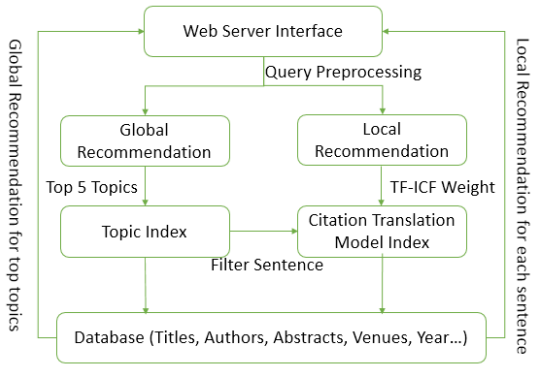

Figure 2: Overview of the RefSeer Infrastructure.

of sentences. Furthermore, we remove the stopwords in each sentence. For the global recommendation model, the entire query (stopword removed) will be used for the model to infer topics. For the local recommendation model, each sentence will be processed as a separate query, and a list of papers will be recommended for each sentence.

3.4 Global Recommendation

For global recommendation, RefSeer internally computes topical compositions for each paper using the Cite-PLSA-LDA model [7]. This model extended the original LDA model by assuming that the words in the citation context are not only related to the topics of the citing document but also the topics of the cited document. Thus the words in the citation contexts and citations are generated from topic-word and topic-citation multinomial distribution. Word-topic $\Pr(t|w)$ and topic-citation $\Pr(d|t)$ distribution was inferred over all documents in the CiteSeerX repository. When training the topic model, the number of topics is set to 1000.

3.4.1 Recommendation

When facing a new query, RefSeer will infer the top 5

topics (at most) from the input query using the word-topic distribution. For each top topic, a list of citations will be recommended using topic-citation distribution.

Topic-based recommendation is very effective for global recommendation. In addition, the recommendation lists were generated with respect to topics, so the results were "naturally" clustered when recommended. Thus the topic-based global recommendation is very convenient for users to utilize and to understand how the results were generated.

However, such a method suffers from a critical problem – for a certain topic, the recommendation list is fixed (ranked by topic-citation distribution). To overcome the problem, we introduce the citation-context-based local recommendation.

3.5 Local Recommendation

For local recommendation, RefSeer uses a Citation Translation Model [8] to learn the "translation" probability of citing a document given a word $\Pr(d|w)$. This model assumes that the citation context is the "source language," and the "target language" is the references list, where each referenced paper is considered as a "word." The intuition is that authors will explain the details related to the cited paper in the citation context; thus the cited paper can be regard as a "word" that was translated from the citation context.

3.5.1 Learning the Model

A paper consists of two parallel languages: words in all citation contexts are the "source language" d, and the reference list is the "target language" r. One entry of the training data:

$$\begin{array}{ll} \text{Source} & t_{c_1,1}, \cdots, t_{c_1,|c_1|}, \cdots, t_{c_k,1}, \cdots, t_{c_k,|c_k|} \\ & \Downarrow \\ \text{Target} & r_1, r_2, \cdots, r_m \end{array}$$

where $t_{c_i,j}$ is the jth word appearing in the ith citation context of d and r_i is the ith cited paper in r.

The IBM Model-1 models the translation process based on word-level alignment. In our case, a word aligned to a paper indicates that the word needs that particular citation. According to an alignment $A = [a_1, \cdots, a_m]$, where $a_i = j$ means r_i is aligned to t_j, the objective function for training on whole dataset is formulated as:

$$\text{Maximize} \quad \Pr(r|d) = \sum_{a_1=1}^{l} \cdots \sum_{a_m=1}^{l} \prod_{i=1}^{m} \Pr(r_i|t_{a_i})$$

$$\text{Subject to} \quad \sum_{i=1}^{m} \Pr(r_i|t_j) = 1 \qquad j = 1, 2, \cdots, l$$

We modify the GIZA++ toolkit [9] to learn translation probabilities using the IBM Model-1.

3.5.2 Recommendation

Given a query $Q = [t_1, t_2, \cdots, t_l]$, local recommendations $R = [r_1, r_2, \cdots, r_m]$ were generated using all words in the query and assigned the score for each reference r_i as:

$$\Pr(r_i|Q) = \sum_{j=1}^{l} \Pr(r_i|t_j) \Pr(t_j|Q) \tag{1}$$

where $\Pr(r_i|t_j)$ is the probability of citing r_i given word t_j , $\Pr(t_j|Q)$ is the probability that word t_j needs citations.

We use term-frequency-inverse-context-frequency (TF-ICF) to measure $\Pr(t_j|Q)$, the probability of a citation need. Given a query Q, TF_t is defined as the number of times a given word t appears in Q. ICF gives a measure of whether the word is common or rare across all citation contexts. $\text{ICF}_w = \log \frac{|C|}{|w \in c; c \in C|}$, where C is the set all of citation contexts, and $|w \in c; c \in C|$ indicates the number of citation contexts that contain the word w.

Note that not all sentences in a query need citations. We use the topic distribution of the input query inferred in the phase of global recommendations to rule out sentences that do not requires a citation. If the topic distributions of top recommendations for a sentence have a very low similarity[3] with the topic distribution of the query, we will ignore the sentence.

4. EVALUATION

In this section,we first evaluate the performance of citation recommendations on two small datasets. We report the training and recommending complexity of both global and local recommendations. Also, we report the training and recommending time of our real system. For all evaluations we use the papers' reference lists as ground truth.

4.1 Datasets

The first dataset, collected from the CiteSeer digital library, was created by Lise Getoor's research group at the University of Maryland. This data has been widely used for citation recommendation by Kataria, et al. [7], Tang and Zhang [10] and Nallapati, et al. [11]. The second dataset was acquired from CiteULike[4] from November 2005 to January 2008. Besides these two small datasets, we also trained the global and local recommendation models on the whole CiteSeer repository. The characteristics of all three datasets are shown in Table 1.

Table 1: D is the number of documents, C is the number of citation contexts, R is the number of unique references, and $\bar{N}_c$ is the average number of citations a paper has.

Data	D	C	R	$\bar{N}_c$
CiteSeer(small)	3,312	26,597	2,138	18.01
CiteULike	14,418	40,720	5,484	8.61
CiteSeer(whole)	1,017,457	10,760,318	856,758	20.73

For all datasets, we removed the stopwords. For the small datasets, CiteSeer(small) and CiteULike, we randomly partitioned them into 5 subsamples and then performed 5-fold cross validations on these datasets.

4.2 Complexity Analysis

We will compare the complexity of global and local recommendation methods for both the learning and recommending tasks. Assume the number of training iterations for Cite-PLSA-LDA and Citation Translation Model as I (note that I actually varies among different methods), the number of topics for Cite-PLSA-LDA as K, the average number of words each citation context has as $\bar{N}_{cc}$, the average number of words each paper has as $\bar{N}_w$, and the average number of citations every paper cites as $\bar{N}_c$.

For the training stage, the complexity of Cite-PLSA-LDA is $O(IKD\bar{N}_w)$, and that of CTM is $O(ID\bar{N}_{cc}\bar{N}_c{}^2)$. Note that

[3] We use KL-divergence to calculate the similarity of the two distributions.

[4] `http://www.citeulike.org/`

$\bar{N}_c$ is usually around 20, which is 10 to 20 times less than K (ranging from 200 to 500 or even more) and $\bar{N}_{cc}\bar{N}_c < \bar{N}_w$.

For the recommending stage, suppose that we have a new query q with N_q words. The complexity of the global recommendation is $O(IKN_q)$, and that of the local recommendation is $O(N_q\bar{R}_q)$, where $\bar{R}_q$ is the average number of dictionary entries for each word in q. The number $\bar{R}_q$ usually drops tremendously (to around 20 to 50) after several iterations if we wipe out those entries with translation probabilities that are too low.

Table 2: Training and recommending time cost on `CiteSeer(small)` and `CiteULike` datasets and whole CiteSeer database. The recommending time for `CiteSeer(whole)` is per query.

	Training		Recommending	
	Global	Local	Global	Local
CiteSeer(small)	594.115s	53.372s	1.845s	1.480s
CiteULike	8949.210s	71.460s	20.154s	4.904s
CiteSeer(whole)	4d01h49m	5h32m03s	<5ms	<5ms

Table 2[5] and the above analysis show that both global and local recommendation models are comparatively simpler and more efficient for recommending tasks, which makes it possible for RefSeer to become a real-time recommendation system. Although the training phase is comparatively time-consuming due to the big data size of CiteSeer, we are able to update the recommendation models semi-monthly or even weekly.

4.3 Recommendation Results

We reported our experiment on two small datasets. In Table 3, we show the Bpref and MRR results (the larger the better) for both global and local recommendations. From the MRR results in Table 3, we can see that the first correct recommendation will most likely appear among the top 2 for local recommendations. For global recommendations, the first correct recommendation mostly appears around the top 5 recommendations. These results prove our system's usability – most of the time, users can get correct recommendations from the first page (10 recommendations per page).

Table 3: Bpref and MRR metrics on `CiteSeer(small)` and `CiteULike` datasets with 20 recommended papers.

	CiteSeer		CiteULike	
	Bpref	MRR	Bpref	MRR
Global	0.459	0.285	0.260	0.143
Local	0.645	0.529	0.627	0.467

Table 4: Precision and Recall metrics on `CiteSeer(small)` and `CiteULike` datasets with 10 recommended papers.

	CiteSeer		CiteULike	
	Pre.	Rec.	Pre.	Rec.
Global	0.13	0.38	0.08	0.49
Local	0.15	0.48	0.12	0.57

Table 4 shows the precision and recall results for both global and local recommendations. With the top 10 recommendations, the local recommendation model achieves a recall around 50%, which indicates that local recommendations can recommend correct citations among the top 10 recommendations for about half of the input sentences. The precision scores for both global and local recommendations indicate that there will be at least 1 correct recommendation among top 10 recommendations. An example of the recommendation results is also shown in Figure 1.

Although the evaluation scores are lower than those of other recommendation tasks, these scores are not an indication that the performance is poor. We use the papers' reference lists as the ground truth for evaluation. Therefore, the citations are made with author biases: different authors may use different citations according to their own knowledge and perception of the field.

[5]Experiments were tested on the same single machine with 8 CPUs processors of 2.50GHz and 32G memory

5. CONCLUSION AND FUTURE WORK

We presented RefSeer, a citation recommendation system which can be used to check the completeness of references based on the content of a paper manuscript. Experiments on real datasets show that our system recommends citations with good quality. The complexities of training and recommending show that our system is efficient and scalable.

For future work, we plan to make automatic research timeline generation a public service, which will make literature review easier for researchers.

Acknowledgments

We gratefully acknowledge partial support from the National Science Foundation.

6. REFERENCES

[1] C. L. Giles *et al.*, "Citeseer: an automatic citation indexing system," in *Proc. of the 3rd ACM Conf. on Digital Libraries.* New York, NY, USA: ACM, 1998, pp. 89–98.

[2] P. F. Brown *et al.*, "The mathematics of statistical machine translation: parameter estimation," *Computational Linguistics*, vol. 19, no. 2, pp. 263–311, Jun. 1993.

[3] K. Yamada and K. Knight, "A syntax-based statistical translation model," in *Proc. of the 39th Association for Computational Linguistics.* ACL, 2001, pp. 523–530.

[4] P. Koehn *et al.*, "Statistical phrase-based translation," in *Proc. of the 2003 Conf. of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1.* Stroudsburg, PA, USA: ACL, 2003, pp. 48–54.

[5] Q. He *et al.*, "Context-aware citation recommendation," in *Proc. of the 19th Int. Conf. on World wide web.* New York, NY, USA: ACM, 2010, pp. 421–430.

[6] O. Kucuktunc *et al.*, "Fast recommendation on bibliographic networks," in *Proc. of ASONAM2012.* IEEE Computer Society, 2012, pp. 480–487.

[7] S. Kataria *et al.*, "Utilizing context in generative bayesian models for linked corpus," in *Proc. of the 24th AAAI Conf. on Artificial Intelligence*, pp. 1340–1345.

[8] W. Huang *et al.*, "Recommending citations: Translating papers into references," in *Proc. of CIKM'12*, 2012, pp. 1910–1914.

[9] F. J. Och and H. Ney, "Improved statistical alignment models," in *Proc. of the 38th Annu. Meeting on Association for Computational Linguistics.* Stroudsburg, PA, USA: ACL, 2000, pp. 440–447.

[10] J. Tang and J. Zhang, "A discriminative approach to topic-based citation recommendation," in *Proc. of the 13th PAKDD.* Springer-Verlag, 2009, pp. 572–579.

[11] R. M. Nallapati *et al.*, "Joint latent topic models for text and citations," in *Proc. of the 14th ACM SIGKDD.* New York, NY, USA: ACM, 2008, pp. 542–550.