Can't See the Forest for the Trees? A Citation Recommendation System

Cornelia Caragea
Computer Science and Engineering
University of North Texas
ccaragea@unt.edu

Adrian Silvescu
Naviance Inc.
silvescu@gmail.com

Prasenjit Mitra
Information Science and Technology
Pennsylvania State University
pmitra@ist.psu.edu

C. Lee Giles
Information Science and Technology
Pennsylvania State University
giles@ist.psu.edu

Keywords: citation recommendation, singular value decomposition, collaborative filtering

Background and Motivation

As science advances, scientists around the world continue to produce large numbers of research articles, which provide the technological basis for worldwide collection, sharing, and dissemination of scientific discoveries. When writing a paper, an author searches for the most relevant citations that started or were the foundation of a particular topic, which would very likely explain the thinking or algorithms that are employed. The search is usually done using specific *keywords* submitted to literature search engines such as Google Scholar [3] and CiteSeer [2]. However, text-based search engines return poor results when there is vocabulary mismatch between a query and the relevant documents. Moreover, finding relevant citations is distinctive from retrieving articles that are only topically similar to an author's proposal. For example, Teufel et al. [6] showed that citations can be of various types, and provided an annotation scheme for the citation function, which consists of twelve different categories. Among these categories, some citations are topically similar, others are used as survey articles to provide background information to the reader, while yet others contain tools/algorithms/data that are adapted or modified in the new proposal [6].

What is a good strategy to uncover both topically-related and, at the same time, distant, but highly-relevant citations for a particular query, while filtering out irrelevant information, given today's very large collections of published articles? One promising line of research to information filtering is the design of recommender systems. Collaborative Filtering (CF), the commonly used method for recommender systems, relies on the assumption that *similar users express similar interests on similar items*. Two types of memory-based CF are: user-based CF, which computes the similarity between users, based on user profiles or history, and item-based CF, which computes the similarity between items, based on various item information.

However, memory-based CF algorithms have several limitations such as data sparsity and scalability [4]. Consider, for example, the problem of citation recommendation in CiteSeer [2], where the underlying citation graph tends to be noisy (e.g., due to errors in citing, and missing citations, etc.) and sparse (i.e., most papers in CiteSeer have on average about 30 citations, which is far below 1% of the more than two million papers indexed into the database). Memory-based CF algorithms, which are nearest neighbor based algorithms, rely on exact matches while computing (item or user) vector similarities, and hence, could result in loss of citation recommender system coverage and accuracy [5]. Moreover, the nearest neighbor algorithms require similarity computation that grows with the number of research papers. With more than two million papers available in CiteSeer, CF-based citation recommendation systems suffer from scalability problems.

Caragea, C., Silvescu, A., Mitra, P., & Giles, C. L. (2013). Can't see the forest for the trees? A citation recommendation system. *iConference 2013 Proceedings* (pp. 849-851). doi:10.9776/13434

Against this background, we address the problem of citation recommendation using singular value decomposition (SVD) [1] on the adjacency matrix associated with the citation graph to construct a latent “semantic" space, where citing and cited papers that are highly correlated are placed closed to each other. The main idea behind SVD is to project the original high-dimensional data into a lower-dimensional space, in which patterns in the data can be more easily identified.

We exploit information available in the online CiteSeer digital library of scientific publication to train and evaluate our models. The assumption is that, when writing a paper, an author has some background knowledge about the topic he writes about and that an initial set of citations (i.e., a “basket" of citations) is provided as input to the recommender system. We require the system to retrieve other relevant works that the author might have missed (works that should be cited or the author should be aware of).

Experiments and Results

The CiteSeer Dataset: The citation recommendation data set used in our experiments is compiled from the CiteSeer citation graph and the metadata available for each paper indexed in CiteSeer [2], as of December 2011.There are 1, 345, 249 unique citing papers and 9, 150, 279 unique citations in the CiteSeer citegraph. The total number of links in the graph, i.e., [citing paper → citation], is 25, 526, 384. From the CiteSeer citegraph and the available metadata, we constructed a smaller data set for the task of citation recommendation as follows: we filtered out papers that do not have title and abstract, as well as papers that are cited by other papers in the corpus less than 10 times and more than 100 times. In addition, we filtered out papers that cite less than 15 or more than 50 other papers. In our resulting citegraph, there are 81, 508 unique cited papers (citations), 16, 394 unique citing papers, and 341, 191 links.

Experimental Design: We address the following question: How does SVD compare with collaborative filtering approaches on the citation recommendation task? We split the data set into training and test sets by randomly selecting one non-zero entry from each citing paper, to be part of the test set, whereas the remaining non-zero entries are considered part of the training set (i.e., the “basket items" for each citing paper). We sampled a validation set, used to estimate model parameters, in a similar manner.

Figure 1 show the results of the comparison of SVD with CF algorithms, i.e., CF User-based Simple Weighted Sum (SWS), CF User-based most-frequent-item (F), and CF Item-based. Each algorithm returns a list of top N recommendations for each citing paper, with N ranging from 5 to 25 in steps of 5. If the hidden citation (in the test set) is part of the top N recommendation list returned by an algorithm, the algorithm was considered accurate for the particular citing paper. We repeated each experiment 5 times to ensure the results were not sensitive to a particular train-test split and averaged the results across the five runs. As can be seen in the figure, SVD outperforms the CF models in terms of both Recall and Precision, for all values of N . In future, because SVD-like models easily allow incorporation of additional information, we plan to integrate other types of information (e.g., textual information) into our models.

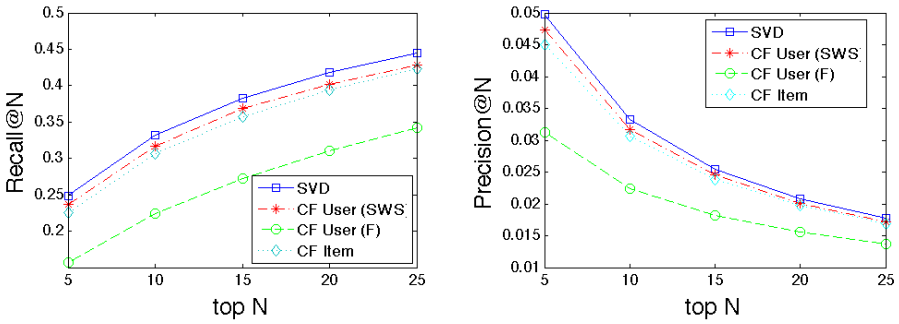

Figure 1: Comparison of SVD with collaborative filtering algorithms in terms of Recall and Precision.

References

[1] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[2] C. L. Giles, K. Bollacker, and S. Lawrence. Citeseer: An automatic citation indexing system. In *Digital Libraries '98*, pages 89–98, 1998.

[3] Google. Google scholar. In *http://scholar.google.com*.

[4] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Application of dimensionality reduction in recommender system a case study. In *WebKDD-2000 Workshop*, 2000.

[5] B. Sarwar, J. A. Konstan, A. Borchers, J. Herlocker, B. Miller, and J. Riedl. Using filtering agents to improve prediction quality in the grouplens research collaborative filtering system. In *Proceedings of the*
1998 ACM conference on Computer supported cooperative work, pages 345–354. ACM Press, 1998.

[6] S. Teufel, A. Siddharthan, and D. Tidhar. Automatic classification of citation function. In *Proceedings of Empirical Methods in Natural Language Processing*, 2006.