TheAdvisor: A Webservice for Academic Recommendation

Onur Küçüktunç[1,2], Erik Saule[1], Kamer Kaya[1], and Ümit V. Çatalyürek[1,3]
[1]Dept. of Biomedical Informatics, The Ohio State University
[2]Dept. of Computer Science and Engineering, The Ohio State University
[3]Dept. of Electrical and Computer Engineering, The Ohio State University
kucuktunc.1@osu.edu, {esaule,kamer,umit}@bmi.osu.edu

ABSTRACT

The academic community has published millions of research papers to date, and the number of new papers has been increasing with time. To discover new research, researchers typically rely on manual methods such as keyword-based search, reading proceedings of conferences, browsing publication lists of known experts, or checking the references of the papers they are interested. Existing tools for the literature search are suitable for a first-level bibliographic search. However, they do not allow complex second-level searches. In this paper, we present a web service called the**advisor** (`http://theadvisor.osu.edu`) which helps the users to build a strong bibliography by extending the document set obtained after a first-level search. The service makes use of the citation graph for recommendation. It also features diversification, relevance feedback, graphical visualization, venue and reviewer recommendation. In this work, we explain the design criteria and rationale we employed to make the the**advisor** a useful and scalable web service along with a thorough experimental evaluation.

Categories and Subject Descriptors

H.3.5 [**Information Search and Retrieval**]: On-line Information Services

Keywords

Literature search; citation graph; random walk; paper recommendation; relevance feedback; visualization; result diversification

1. RECOMMENDATION FRAMEWORK

Here we describe the components of the service, and present our design choices. Figure 1 gives an overview of the**advisor**'s framework and the relationship between its components.

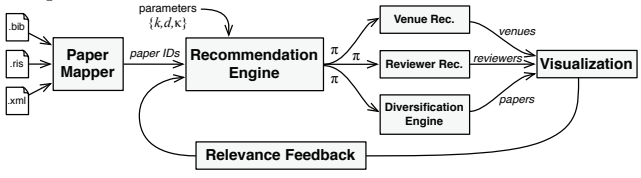

Figure 1: Overview of the theadvisor framework.

JCDL'13, July 22–26, 2013, Indianapolis, Indiana, USA.
ACM 978-1-4503-2077-1/13/07.

The framework has four main components [6]:
Paper mapper. The recommendation process starts with uploading the user's bibliography file to the website in BibTeX, RIS, or EndNote XML format. While mapping the papers given in different file formats, we extract the alphanumeric characters from each title, split them into words, and use an inverted index on title words to find the matched papers in our database. Since there can be very similar titles and/or a conference version of an extended journal paper, or vice versa, we also employ the publication date and Levenshtein distance to find the exact match. This process makes the paper matching step more accurate and efficient.
Recommendation engine highlights a diversified set of papers/citations, venues, and experts to the user.
Visualization uses graph drawing techniques on the recommended paper set to visualize the relations between them.
Relevance feedback gets the comments from the user on the recommendations and refines the search accordingly.

1.1 Recommendation engine

Direction-aware Random Walk with Restart. We define a *direction awareness* parameter $\kappa \in [0,1]$ to obtain either more recent or more traditional results in the top-k documents [3]. Given a query with inputs k, a *seed* paper set $\mathcal{Q}$, a damping factor d, and a direction awareness parameter κ, Direction-aware Random Walk with Restart (DARWR) computes the steady-state probability vector π. The rank vector at iteration t is computed with the following linear equation $\mathbf{p}_{t+1} = p^* + \mathbf{A}\mathbf{p}_t$, where p^* is an $n \times 1$ restart probability vector, and $\mathbf{A}$ is a structurally-symmetric $n \times n$ transition matrix of edge weights, such that $a_{ij} = \frac{d(1-\kappa)}{\delta^+(i)}$ if $(i,j) \in E$, and $a_{ij} = \frac{d\kappa}{\delta^-(i)}$ if $(j,i) \in E$, and $a_{ij} = 0$, otherwise.

Venue and reviewer recommendation. Since we target the manuscript preparation and submission process, venue recommendation queries are useful while deciding the conference or journal for submission, and reviewer recommendation queries are useful while submitting a manuscript to some journals which require a set of names of potential reviewers or the editor of a journal looking for reviewer for a particular paper.

Improving the efficiency of the ranking. As described above, each iteration of the ranking algorithm DARWR contains a sparse-matrix dense-vector multiplication (SpMV). This sparse linear algebra kernel is the building block of the**advisor**'s recommendation engine. We observed that the nonzero pattern of the citation matrix is highly

irregular and the computation suffers from this irregularity due to the high number of cache misses. Since there will be a penalty for each cache miss, we apply preprocessing steps, partitioning and ordering, to reduce the number of cache misses and make the ranking algorithm faster. Using a hypergraph partitioning model and Approximate Minimum Degree (AMD) ordering technique, we obtained 3× speedup [1, 2].

Result diversification. Methods such as DARWR tend to naturally return many recommendations from the same area of the graph, which leads to a poor coverage of the potential interests of the users. Diversifying the recommendations refers to the methods that increase the amount of distinct information one can reach via an automatized search. We argue that finding the vertices which are locally maximum in the graph w.r.t. their ranks and returning the k most relevant ones will diversify the results and increase the coverage of citation graph. In order to keep the results within a reasonable relevancy threshold and to diversify them at the same time, we incrementally compute the local maximas only within the top-γk ranked results and remove the selected vertices from the subgraph for the next iteration until k results are obtained. We refer to this algorithm as parameterized relaxed local maxima (γ-`RLM`) [4, 5].

1.2 Relevance feedback

Users of the**advisor** are given the option of providing explicit relevance feedback to the set of recommended papers. The feedbacks can be either positive or negative, making a recommendation relevant or irrelevant for the query. When the user refines the query with the relevance feedback, the relevant results are added to $\mathcal{Q}$, and the irrelevant results are removed from the citation graph with all of their incident edges. Relevance feedback can also be incorporated with the diversification feature explained in [4]. As the diversified set of results may represent different sets of papers from different areas of interest, the user of the service can easily guide the recommendation process towards the fields that she is interested in.

1.3 Data visualization

Representation of the recommendations in a web service is crucial for user experience. Aside from displaying the full bibliographic entries for the list of suggested papers, we also visualize the results and their relationships to the references and papers that are given positive feedback before. The sample graph (see Fig. 2) consist of the seed papers $\mathcal{Q}$ (*blue*), recommendations (*green*), and top-100 relevant papers (*white*). The subgraph is extracted from those vertices and all the edges within this subset. We then apply a force-directed layout algorithm to improve the representation as well as expose the paper clusters.

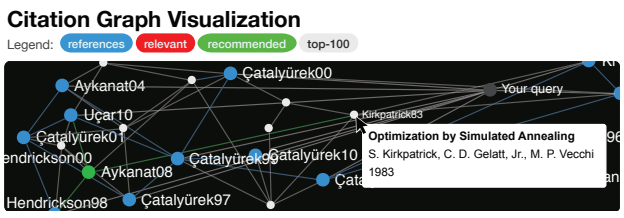

Figure 2: Visualization of a sample query.

1.4 API

For the users and third-party application developers, we provide an interface to query our citation database with an API (`http://theadvisor.osu.edu/apireference.php`) similar to RESTful. The API enables users to query the**advisor** database with a bibliography file, obtain metadata for a given list of papers (as internal IDs), and run the recommendation algorithm with various settings (e.g., specify the direction awareness parameter, ask for a diverse set of results, provide positive and negative feedback, etc.). The complete the**advisor** service, in fact, can be externally implemented using only the provided actions. This API allows to interconnect external applications and web services with the**advisor**. For instance, websites that let users create and group papers of interest (e.g., Mendeley, CiteULike, etc.) could perform recommendation using the**advisor**.

2. CONCLUSIONS

Bibliographic search has become a herculean task due to the vast literature published every year by researchers. Web services have appeared to help researchers in this task. Yet, we felt that these methods only provided a first insight in the existing literature. To further help the researchers, we developed the**advisor**, a publicly available web service which extends a set of given documents via recommendation based on the citation analysis. It provides multiple features such as direction awareness, relevance feedback, graphical visualization, venue recommendation, and reviewer recommendation. It can also be used automatically thanks to a well defined API. All the features of the system are based on sound algorithmic decisions and are shown to be very beneficial in practice. We believe that the**advisor** will find its place in the ecosystem of academic web services.

Acknowledgments

This work was partially supported by the U.S. Department of Energy SciDAC Grant DE-FC02-06ER2775 and NSF grants CNS-0643969, OCI-0904809 and OCI-0904802. The authors would like to thank DBLP and CiteSeer^x for making their datasets available.

3. REFERENCES

[1] O. Kucuktunc, K. Kaya, E. Saule, and U. V. Catalyurek. Fast recommendation on bibliographic networks. In *Proc. IEEE/ACM Int'l Conf. Advances in Social Networks Analysis and Mining (ASONAM)*, 2012.

[2] O. Kucuktunc, K. Kaya, E. Saule, and U. V. Catalyurek. Fast recommendation on bibliographic networks with sparse-matrix ordering and partitioning. *Social Network Analysis and Mining (SNAM)*, 2013. (to appear).

[3] O. Kucuktunc, E. Saule, K. Kaya, and U. V. Catalyurek. Direction awareness in citation recommendation. In *Proc. Int'l Workshop on Ranking in Databases (DBRank'12) in conjunction with VLDB'12*, 2012.

[4] O. Kucuktunc, E. Saule, K. Kaya, and U. V. Catalyurek. Diversifying citation recommendation. Technical Report arXiv:1209.5809, ArXiv, Sep 2012.

[5] O. Kucuktunc, E. Saule, K. Kaya, and U. V. Catalyurek. Diversified recommendation on graphs: Pitfalls, measures, and algorithms. In *Proc. Int'l Conf. World Wide Web (WWW)*, 2013.

[6] O. Kucuktunc, E. Saule, K. Kaya, and U. V. Catalyurek. Towards a personalized, scalable, and exploratory academic recommendation service. In *Proc. IEEE/ACM Int'l Conf. Advances in Social Networks Analysis and Mining (ASONAM)*, 2013.