Re-ranking Recommended Citations Based on Its Predicted Impact

Divit Singh
divit52@vt.edu

Harsh Agrawal
harsh92@vt.edu

Ambika Karanth
ambik89@vt.edu

ABSTRACT

When researching while writing a paper, it is not always the case that the first link returned by a search engine is the best one. Searching through engines involves using a few keywords that relate to your paper. Therefore, it is less likely that search engines might return desirable results when searching for entire sentences. Our approach aims to circumvent this bottleneck by providing recommendations that utilizes more information than a few keywords. Through the concept of context analysis (analyzing entire sentences scattered throughout a paper), our intent is to provide more accurate recommendations than would be found if simply searching for a few keywords. In addition to providing a recommendation set, we also want to experiment the relationship between relevance and high impact of a recommended citation. Specifically, we want to see if adding higher impact citation will produce a higher impact paper for the author.

Keywords

recommendations, prediction, high-impact, citation

1. INTRODUCTION

The most ubiquitous approach is to perform short keyword searches on search engines such as IEEE and ACM and analyze the list of papers that are retrieved based solely on those keywords. This generally leads to sequentially shifting through the bibliographic citations to retrieve other documents which may pertain to the searched keywords. Not only is this process time-consuming, but also limits the search space of papers to analyze.

Consider if a researcher was presented with a list of citations automatically based on analyzing context throughout their paper. A researcher would save time by having a generated list of relevant documents to read through. It is crucial that the documents provided are actually relevant to the context in which it is searched for. It is equally important that the speed of retrieval of these relevant papers from the massive set is fast and provide good coverage within the research field.

The goal of a researcher is not only to publish novel research, but also attract attention from people in their field to use their research. A popular belief in academia is that citing papers that have been highly cited (have high impact) tends to get more citations than citing a paper with lower impact. Our goal is to assess this myth by re-ranking our original recommendations based on the projected impact of its citations to see if it results in a higher impact paper.

There are many challenges in relation to predicting the impact of a paper by using recommended citations. First, we have to explore the aspects of a good recommendation engine. Next, we consider how to predict the impact of those citations. Finally, we want investigate whether re-ranking the recommended citations for a given paper is, in fact, useful.

2. PROBLEM DESCRIPTION

Given a graph G <V, E> where V in G represents the set of all the research papers, let a directed edge E from V_1 to V_2 represents V_1 citing V_2 in its bibliography[6]. For every edge E in G, we want to assign a weight representing the similarity between the two vertices. For a target paper T in the test set, we will use the weights of the outgoing edges from T, along with some feature representation of the abstract and full text of T, to recommend a set of papers C. Next, we re-rank the set of papers C based on the predictive impact of each citation $c_i \in C$. We use this metric to assess the co relation between the impact of cited papers and the impact of the target paper T.

3. LITERATURE REVIEW

3.1 Recommendation

In this section, we discuss related work in the fields of citation recommendation. One of the most popular approaches to recommendation in general is Collaborative Filtering [7][12]. Given an initial set of citations, McNee et al[9] used Collaborative Filtering (CF) to recommend additional papers. Traditional CF algorithms view the data set as a rating matrix. The rows in this matrix represent the set of users. The column represent the set of items. Each element in the matrix would be the ratings associated with a particular item for the corresponding user. In case of a movie recommendation system, the items would be movies and the values of each element in the matrix would be the ratings associated with the movies for the corresponding user.

To apply CF on the problem of citation recommendation McNee et al[9] assigned each row to be a paper with an initial bibliography. Every citation in the bibliography provided by these papers becomes its own column in the matrix. Each element $b_{i,j}$ is assigned 1 if i^{th} paper has the j^{th} citation. Here i represents the citing paper(rows) and j represents the citation (column). The author used the following

CF algorithms for recommendations: Co-Citation Matching, User-Item CF, and Item-Item CF.

Co-Citation counts the number of times a pair of citations occur together in a list of bibliographies. Let the set of all possible recommendations be R and a target paper be T. For each paper ($r \in R$), the co-citation count would be the sum of number of times r was co-cited with each paper in the bibliography of T. The recommended paper would be the paper which has the maximum citation count. **User-Item CF** constructs a similar matrix as Co-Citation matching. However, it returns a set of papers in R where the similarity of each paper $r \in R$ with respect to T is multiplied with the citation count of r. The similarity of r is determined by cosine similarity. **Item-Item CF** flips the User-Item CF and creates a neighborhood of items instead of users. It is discussed in more detail in [5].

3.1.1 Context-Aware Recommendation

One of the limitations using approaches that require the presence of an initial bibliography is that it places the burden on the user. The user is expected to provide a list of related work that they previously gathered. In order to circumvent this limitation, approaches started leveraging the idea of context. A context for a document is a bag of words model[3]. The global context is the title and the abstract of a paper. The local context is the neighboring text of an in-line citation placeholder. It is shown that citation context is a good summary of the motivation behind using existing set of citations[3]. Leveraging both global context and local context satisfy different information needs at different phases of writing a paper[1].

Initially, the author tends to find a generic set of papers based on prior knowledge of related literature. For this purpose, the global context would be the optimal choice since it leverages general information such as venue, co-citations, and prior publication history of authors. However, during the writing process, the author might need suggestions to support a particular argument in a given section. For this scenario, the local context would be the optimal choice as it utilizes surrounding text to give recommendations[3].

3.1.2 CiteSight System

Based on the previous model, a CiteSight system was recently introduced. The model builds a cache of citations to provide instantaneous recommendations at the time of active writing. It continuously updates the list of recommendations by analyzing the contextual metadata of the entire database of indexed papers[1]. Since the resulting index is large, it is impossible for the system to give recommendations in real-time, hence, caches are used. The cache is updated as information such as the title, abstract, and the context of citations are added by the author. The median response time for a cache was found to be 6.2ms, whereas the median response time of using a full index was observed to be 452 ms [1].

3.1.3 Direction-Aware Recommendation

The previous approaches do not take into account user preferences. For example, a user would not be able to obtain citations from a certain time period if they were to use any of the previously discusses approaches. [6] introduces direction aware algorithms which are able to filter recommendations based on user preferences. Consider a graph that consists of out-going and incoming citations, this approach changes the weights assigned to each edge based on the search criteria. Traditional algorithms such as page rank[6] base their recommendations on the number of edges that are connected to a citation. Thus, it is more likely that older papers will get recommended above more recent papers as they have been available to the community for a longer period of time. [6] introduces a direction-awareness paramater λ to obtain either recent or more traditional results in the top k relevant documents. This is done by modifying the parameter λ to change the contribution of tradition papers or recent papers based on user preferences.

3.2 Prediction

We have discussed different approaches that have been explored when recommending citations for a given paper. Now we shift our focus to analyze approaches that predict how well a given paper will be received by the community. We analyze this topic to investigate the myth of whether citing higher impact papers will result in a higher citation count.

3.2.1 Time-Series Based Approaches

When predicting citation count of papers, there have been time-series based approaches [18][17]. [18] converts the citation data into time-series data that is suitable for prediction using regression. The time-series data using the format: <citing paper id, citing paper id> helps in investigating the change in citations of a number of papers over time. [18] divided the citation data into overlapping periods of time. Specifically, they divided their data set into overlapping 3-month increments (e.g. January to March, February to April etc). This approach not only increased the amount of training data available, but also reduce noise (possible errors in publication date of a given paper).

This table was used to create another table which contained differences between the citation count of adjacent intervals. This table served as the input for a regressor to predict the change in citation of a given paper over time. [17] not only used time-series features, but also a set of keywords that represented the current topic trends. To capture the intuition that the paper of a famous author will be better received by the community, relation based features such as h-index of authors were taken into consideration.

3.2.2 Using Decay In Time-Series

While measuring paper impact [14] takes into account citations over several years. This is done to differentiate papers that have a lasting influence from the papers that were popular over a short period of time. This is done by using exponential decay which contains a parameter r which controls the rate of decay. Most of the prediction algorithms used standard machine learning implementations such as decision trees, random forests etc.

4. DATASET

The KDD Cup was one of the first attempts at creating a standard data set for the problem of predicting citations [16]. KDD Cup 2003 consists of the citation graph of the papers in hep-th (high energy physics theory)section of arXiv which is a popular repository of electronic preprints [15]. The node in the dataset are papers and if a paper i cites paper j, the graph contains a directed edge from i to j. The challenge also provides full text and abstract of the papers.

The data covers papers from January 1993 to April 2003 (124 months). It begins within a few months of the inception of the arXiv, and thus represents essentially the complete history of its hep-th section. The dataset constitutes of 30,119 papers written by 57,448 authors comprising in total 1.7GB of LaTeX sources with 719,109 total citations in the papers. 363,812 of these citations cited a paper outside hep-th, and 355,297 citations cited papers from the hep-th section.

5. PLAN OF ATTACK

First all the members of this group will start by cleaning the different parts of the data set such as abstract, full text, and citation. Next, Harsh and Divit will determine the useful information from the data set and extract initial set of features from the cleaned data. Ambika will work primarily on visualizing the data sets and computing statistics. All of the fore-mentioned tasks will take approximately 10 days to complete. Then, Harsh and Divit will implement simple recommendation models while Ambika starts to work on the prediction model.

All of us will spend the next two weeks analyzing the results of the initial experiments and making sure that they are consistent with the results reported by past works. These initial models will act as the baselines for the more complex models. By the midterm evaluation, our group should hopefully have a decent understanding of the data set and will know the desired features to be extracted from the data set. The group will then combine these features to model more complex relationships to feed into recommendation models. Since this is a complex task, the group will keep tuning parameters until satisfactory results are obtained. We estimate that this process will take about 20 days.

The final stretch of our project will involve the group trying to formulate various algorithms to re-rank recommendations and analyze the role it plays in determining the future impact of the target paper. This will leave the group a period of 1 week to document the code and publish the results.

6. EXPERIMENTS AND EVALUATION

There are two stages for which evaluations need to be performed for this project: Recommendation and Prediction. We perform experiments to evaluate the performance of both of the tasks separately. To measure the performance of the recommendation model, we first use three simple evaluation metrics: recall, F1-measure, and mean reciprocal rank (MRR) [2][5] and then use more appropriate metrics.

Recall is the fraction of the documents that were retrieved successfully from the set of all relevant documents.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{1}$$

Here, True Positive contains the set of documents that were correctly classified as relevant. False Negatives contains the set of documents incorrectly classified as not relevant.

F1-measure is the harmonic mean of precision and recall. It is often used in information retrieval for search, document classification, and query classification.

MRR is the average of the reciprocal rank of the results for a sample query. MRR is often used to evaluate how good a list of responses are to a particular query.

$$MRR = \frac{1}{\mod Q} \sum_{i=1}^{n} \frac{1}{rank_i} \tag{2}$$

Another, more accurate form of evaluation of a recommendation model would be to compare the recommended citations to the actual ground truth, i.e, existing list of citations for a published paper. However, this approach may suffer from missing citations due to space constraints. As a result, the ground truth is not complete. In this scenario, the performance is measured using a metric based on relevancy score between the recommended paper and the originally cited paper.

Relevancy score is proportional to the number of times a paper was co-cited with the ”true” citation .

$$res_s = \frac{no.ofpaperscitingst}{numberofpaperscitingt} \tag{3}$$

where s is a paper from the list of recommended papers, and t is the paper that actually existed in the bibliography of the test paper.

Discounted Cumulative Gain (DCG) utilizes the relevancy score and measures the usefulness of a document based on its position in the result list[1]. The logic behind this measure is that highly relevant documents appearing lower in the search result gets penalized. In other words, in addition to the relevance, the order in which the citations are presented are also taken into account. We can use this metric to validate the myth of whether high impact citations lead to higher impact papers.

$$DCG_p = rel_i + \sum_{i=2}^{p} \frac{rel_i}{\log i} \tag{4}$$

To evaluate the prediction task , we compare our methods against the submissions in the KDD Cup 2003 challenge. The specific task is to predict the difference of a well cited paper between (a) the number of citations it received from February 1, 2003 and April 1, 2003 and (b) the number of citations received during the period from May 1, 2003 to July 31st, 2003. The evaluation metric here is the L_1 distance between the vector representing the predicted change for each of these periods[15].

7. CONCLUSION

In this project, we want to explore the set of features that significantly contribute towards the improvement of the recommendation model. This problem is nontrivial as it requires significant data-cleaning in order to convert the raw data into useful information. We explore a diverse set of features which include various similarity citation metrics, co-citation matrix, author-paper matrix, and local/global context of the paper. We then experiment with different recommendation models and evaluate the performance of context aware models vs. bibliography based models.

Finally, we want to discredit or validate the myth that the choice of citations used for a target paper affects the impact it will have in the community. We do this by re-ranking the recommended set based on the impact of the citations and measure if there is a significant increase in the predicted impact of the target paper.

8. REFERENCES

[1] Avishay Livne, Vivek Gokuladas, Jaime Teevan, Susan T. Dumais, and Eytan Adar. Citesight: Supporting contextual citation recommendation using differential search. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, pages 807–816, New York, NY, USA, 2014. ACM.

[2] Xiang Ren, Jialu Liu, Xiao Yu, Urvashi Khandelwal, Quanquan Gu, Lidan Wang, and Jiawei Han. Cluscite: Effective citation recommendation by information network-based clustering. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 821–830, New York, NY, USA, 2014. ACM.

[3] Qi He, Jian Pei, Daniel Kifer, Prasenjit Mitra, and Lee Giles. Context-aware citation recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 421–430, New York, NY, USA, 2010. ACM.

[4] Wenyi Huang, Saurabh Kataria, Cornelia Caragea, Prasenjit Mitra, C. Lee Giles, and Lior Rokach. Recommending citations: Translating papers into references. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 1910–1914, New York, NY, USA, 2012. ACM.

[5] Cornelia Caragea, Adrian Silvescu, Prasenjit Mitra, and C. Lee Giles. Can't see the forest for the trees?: A citation recommendation system. In *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '13, pages 111–114, New York, NY, USA, 2013. ACM.

[6] K Onur, Erik Saule, Kamer Kaya, and Umit V Catalyürek. Direction awareness in citation recommendation. 2012.

[7] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, January 2004.

[8] Joeran Beel, Stefan Langer, Marcel Genzmehr, Bela Gipp, Corinna Breitinger, and Andreas Nürnberger. Research paper recommender system evaluation: A quantitative literature survey. In *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation*, RepSys '13, pages 15–22, New York, NY, USA, 2013. ACM.

[9] Sean M. McNee, Istvan Albert, Dan Cosley, Prateep Gopalkrishnan, Shyong K. Lam, Al Mamunur Rashid, Joseph A. Konstan, and John Riedl. On the recommending of citations for research papers. In *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work*, CSCW '02, pages 116–125, New York, NY, USA, 2002. ACM.

[10] Joeran Beel, Marcel Genzmehr, Stefan Langer, Andreas Nürnberger, and Bela Gipp. A comparative analysis of offline and online evaluations and discussion of research paper recommender system evaluation. In *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation*, RepSys '13, pages 7–14, New York, NY, USA, 2013. ACM.

[11] Trevor Strohman, W. Bruce Croft, and David Jensen. Recommending citations for academic papers. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 705–706, New York, NY, USA, 2007. ACM.

[12] Seikyung Jung, Juntae Kim, and Jonathan L. Herlocker. Applying collaborative filtering for efficient document search. In *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, WI '04, pages 640–643, Washington, DC, USA, 2004. IEEE Computer Society.

[13] Khalid El-Arini and Carlos Guestrin. Beyond keyword search: Discovering relevant scientific literature. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 439–447, New York, NY, USA, 2011. ACM.

[14] Daniel McNamara, Paul Wong, Peter Christen, and Kee Siong Ng. Predicting high impact academic papers using citation network features. In *Trends and Applications in Knowledge Discovery and Data Mining*, pages 14–25. Springer, 2013.

[15] Johannes Gehrke, Paul Ginsparg, and Jon Kleinberg. Overview of the 2003 kdd cup. *SIGKDD Explor. Newsl.*, 5(2):149–151, December 2003.

[16] Dimitris Bertsimas, Erik Brynjolfsson, Shachar Reichman, and John M Silberholz. Moneyball for academics: Network analysis for predicting research impact. *Available at SSRN 2374581*, 2014.

[17] Claudia Perlich, Foster Provost, and Sofus Macskassy. Predicting citation rates for physics papers: Constructing features for an ordered probit model. *SIGKDD Explor. Newsl.*, 5(2):154–155, December 2003.

[18] J. N. Manjunatha, K. R. Sivaramakrishnan, Raghavendra Kumar Pandey, and M Narasimha Murthy. Citation prediction using time series approach kdd cup 2003 (task 1). *SIGKDD Explor. Newsl.*, 5(2):152–153, December 2003.