PaperMiner: An Assistant to Extract Information from Research Papers for Knowledge and Career Development

Zan Huang[*]
Georgia Institute of Technology
huangzan@gatech.edu

Kaiwen Luo[*]
Georgia Institute of Technology
kluo37@gatech.edu

Kai Li[*]
Georgia Institute of Technology
kaili@gatech.edu

Jiaji Liu[*]
Georgia Institute of Technology
jliu808@gatech.edu

Maiqi Ding[*]
Georgia Institute of Technology
mding41@gatech.edu

1 INTRODUCTION

Academic papers are great resources for knowledge discovery[9], data mining[7] and network analysis[10]. Also, they are valuable in guiding personal knowledge and career development[18].

Nowadays, the corpus of academic papers is huge with hundreds of millions of publications by authors from tens of thousands of institutions published at conferences and journals on a wide range of topics[6]. At the same time, the number of papers published each year keeps increasing[2]. On the other hand, academic papers are playing a more and more important role in student's early careers for building up a personal knowledge base.

However, the huge number of publications makes it hard for individuals to find relevant literature on a topic. Tools (Arxiv, sotawhat)[1, 4] have been developed to provide search services to reduce the pain by filtering out related papers. Other tools (SOTA, CSRanking)[3, 5] put more focus on providing categorical rankings based on the corpus.

With the advance in nature language processing[8] and more open access papers published in the computer science field, we have richer corpus for text mining compared to citation network analysis for investigating research papers. Existing tools like Google scholar[16] and Aminer[15] put more focus on whole academic paper corpus-based citation network analysis. Instead, we focus on a computer science field, put more focus on text mining for information extraction, go deeper on a subset of the corpus for new findings.

In this work, we present PaperMiner, a novel computer science literature search platform for college students. PaperMiner combines literature search, academic data presentation, interactive visualization, and open-source API for your research.

2 PROBLEM DEFINE

3 SURVEY

A lot of great works [11, 14–16] have been done based on the corpus to study the citation network, topics, and trends of research. Including applying NLP and reinforcement learning to improve the experience for exploring academic paper[17].

Google Scholar and Microsoft Academic are the most widely used literature search platform. They are powerful for searching papers but lack the summary of the data, i.e. researcher collaboration, current trends, etc. AMiner is another platform which combines literature search with data summarization and visualization. It is a data-driven platform that presents analysis results along with research paper. For insights of research advancement and

[*] Authors contributed equally to this work.

collaboration between experts, to support users retrieving academic papers multidimensionally.

AMiner is a powerful academic search platform which satisfies most people's needs, but users are still passive in literature survey which may not meet personalized needs. The searching algorithms used so far rest mostly on searching for titles, authors etc. rather than topics. Furthermore, for college students, it may be better to show recent research trends and recommend representative words besides providing a search bar, best if they can DIY in data analysis based on open-source projects.

We propose PaperMiner, a light-weight and reusable project, designed for students interested in computer science and want to explore the academic papers for their own research interests (through CS topic trends, top influencers and institutes on subfeilds, etc.). We collect data from multiple sources like dblp[11] and arxiv[12], perform data analysis and text mining on pre-processed data and finally integrate and visualize the results by a web app.

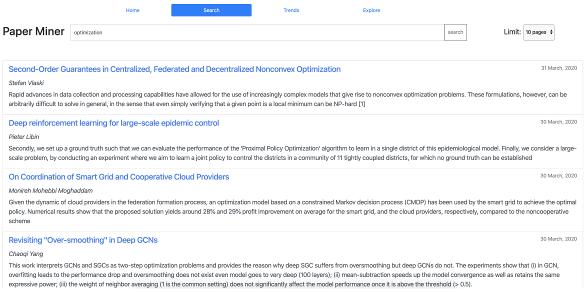
4 PROPOSED METHOD

4.1 Intuition

Comparing with other literature search platforms like Google Scholar, Microsoft Academic and AMiner. PaperMiner combines the literature search with academic data presentation. Apart from presenting the historical data, like the number of publication and citation, PaperMiner also present the insights like trends and topics extracted by machine learning algorithms. PaperMiner focuses on student community and computer science area, providing more tools for helping students' future knowledge and career development.

4.2 Data Collection

We collect data from different sources and perform data integration for later usage. Namely, we use dblp[11] dataset for tracking more than 5,000,000 computer science publications which could be downloaded as one big XML file from the official site[1]. Proceedings of some important conferences are well organized and publically available online, like NeurIPS[2]. We scraped data of all 9,722 papers till 2019 including (paper_id, title, author, abstract) information along with corresponding PDF file which got turned into a TXT file for potential use. Other platforms like Google Scholar and Arxiv and proceedings like ones listed on http://openaccess.thecvf.com/ also got used. The processed .csv is near 1GB.



Second-Order Guarantees in Centralized, Federated and Decentralized Nonconvex Optimization 31 March, 2020

Stefan Vlaski

Rapid advances in data collection and processing capabilities have allowed for the use of increasingly complex models that give rise to nonconvex optimization problems. These formulations, however, can be arbitrarily difficult to solve in general, in the sense that even simply verifying that a given point is a local minimum can be NP-hard [1]

Deep reinforcement learning for large-scale epidemic control 30 March, 2020

Pieter Libin

Secondly, we set up a ground truth such that we can evaluate the performance of the 'Proximal Policy Optimization' algorithm to learn in a single district of this epidemiological model. Finally, we consider a large-scale problem, by conducting an experiment where we aim to learn a joint policy to control the districts in a community of 11 tightly coupled districts, for which no ground truth can be established

On Coordination of Smart Grid and Cooperative Cloud Providers 30 March, 2020

Monireh Mohebbi Moghaddam

Given the dynamic of cloud providers in the federation formation process, an optimization model based on a constrained Markov decision process (CMDP) has been used by the smart grid to achieve the optimal policy. Numerical results show that the proposed solution yields around 28% and 29% profit improvement on average for the smart grid, and the cloud providers, respectively, compared to the noncooperative scheme

Revisiting "Over-smoothing" in Deep GCNs 30 March, 2020

Chaoqi Yang

This work interprets GCNs and SGCs as two-step optimization problems and provides the reason why deep SGC suffers from oversmoothing but deep GCNs do not. The experiments show that (i) in GCN, overfitting leads to the performance drop and oversmoothing does not exist even model goes to very deep (100 layers); (ii) mean-subtraction speeds up the model convergence as well as retains the same expressive power; (iii) the weight of neighbor averaging (1 is the common setting) does not significantly affect the model performance once it is above the threshold (> 0.5).

Figure 1: Search page

4.3 User Interface

Basically, we have four main sections: homepage, search, trends and explore. We have already complete the search page and prvoide some visualization work on other sections.

Homepage. The homepage will show the basic information of PaperMiner, including logo, search box, navigation bar, information of team members. We hope to keep it elegant and informative.

Search. The search page contains the input box, search results within the limit count, the logo and dropdown list, see Figure 1. The dropdown list provides the page limit options for users, including 10 pages, 20 pages, 50 pages, and 100 pages. For the search engine part, we plan to use a hybrid approach by customizing our own based on available service outputs and self-extracted data like the cluster labels of language model(currently using word2vec[13]) processed paper texts.

Trends. In this page, we aim to present a basic academic data analysis. For instance, we have completed the customized trend plot-generation element, see Figure 2. Users can customize their queries

[1] https://dblp.uni-trier.de/faq/How+can+I+download+the+whole+dblp+dataset

[2] http://papers.nips.cc/

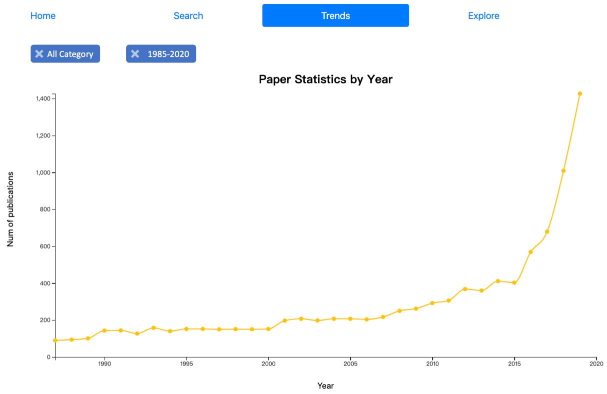


Figure 2: Customized trend plot generation

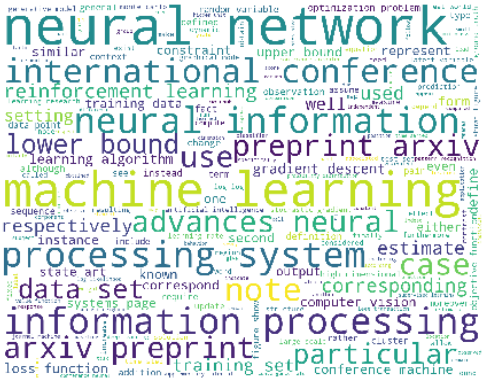


Figure 3: Word cloud

by adding labels. This element can be used as different purposes, such as paper statistics, publications of a professor/college/company. More interactive basic plots will be shown in future work.

Explore. The page left for showing more interesting things, which offer more information with more interactivity and enjoyment. We have already completed the word cloud element, see Figure 3. Word cloud could shows the hot topics with links, so users can dig more information by clicking the text. Other visualization and more meaningful topic generation are still in exploration. We aim to add more interesting stuff on this page based on in-progress work of text mining.

4.4 Algorithm

For PaperMiner users, we hope to return the most related papers need for showing up as result. Behind the page, the clustering algorithm will be implemented to exploit the similarity of papers and group them according to features. To be more specific, the Word2Vec algorithm, a neural network implementation that learns distributed representations for words, and the K-means algorithm will work together for our current clustering. As we are about halfway through this project, we have extracted the keywords in each paper's abstract and divide all the keywords into ten groups. The ten groups of keywords can not only provide us with good understandings of papers' category but also could guide us to filter out more "stop" words to train a better Word2Vec model.

To gain a comparative result, Google's pre-trained Wordto Vector Model was loaded to compare with our trained from-scratch Word to Vector Model, and Google's pre-trained Word2Vec model's result was better-off. Take one of the clusters as an example, 'stochastic', 'non-deterministic' and 'probabilistic' are in the same group which shows that a number of stochastic process applications in computer science are discussed in past papers. In the future days, parameters in the model can be tuned for a more accurate outcome and more algorithms could be added for better user experience.

5 EXPERIMENTS/EVALUATION

Besides details on data-processing and text-mining concerning our current focus on data collection and algorithms. We will also compare PaperMiner with AMiner (The most similar one with our platform) under different tasks. The subjects of the experiments are students including ourselves.

We will test the user experience by concrete scenarios. For instance, we will try to investigate the new hot topics and top researchers in Reinforcement Learning after 2018. We will also evaluate by answering our proposed specific questions, like *"They are undergraduates seeking PhD program and are interested in Machine Learning but have no idea which college to apply. What information you can give to help make decisions?"*.

We will also test the response time and returned results with respect to our platform and AMiner. Try to evaluate the system from several aspects, i.e. speed(for search and navigating visualizations), usability(are visualized results meaningful and helpful for research novice?), etc.

5.1 Experiment Design

5.2 Evaluation

6 CONCLUSION AND DISCUSSION

(1) Apart from presenting the historical data, like the number of publication and citation, PaperMiner also present the insightful stuff like emerging topics and trends sorting to machine learning algorithm.
(2) PaperMiner investigate more and text mining and provide more interactive graphs and plots, to increase user interactivity and enjoyment.
(3) PaperMiner focuses on computer science area and student community, to help those who are preparing for a PhD, or those who aim to stand out in the job market.

7 DISTRIBUTION OF TEAM MEMBER EFFORT

So far, we have completed data collection and cleaning, so the data are ready for displaying and being put into the algorithm. As mentioned above, some graphs and plot elements skeletons have already been done. Algorithms, like PageRank Word2Vec, Clustering, have been used to search papers, extract word frequency from abstract and classify papers.

We will have 18 days to finish our project (Previous Plan see Appendix A). we decide to divide it into two parts. In the first week we will focus on finishing all the functions mentioned above, and embedded our algorithms and functional elements into our platform. In the next week, we will focus on debugging and report writing. Finally, we will complete the experiments, finish the text mining and visualization part, report the evaluation results.

Our group is well designed to assign the same workload to each team member based on our backgrounds. So far Zan was responsible for data collection, Jiaji was responsible for writing UI, Kai, Kaiwen were responsible for the algorithm development, Maiqi and Kaiwen were responsible for writing the report.

All group members then will participate in every part of the whole projects and each part has a leader. Zan will be in charge of data collection, Kaiwen will be in charge of data cleaning and integration, Kai and Maiqi will be in charge of data mining and data modeling, Jiaji will be in charge of data visualization. The entire group will complete the report and presentation together.

REFERENCES

[1] 2017. arxiv-sanity. http://arxiv-sanity.com/. Accessed: 2020-02-20.

[2] 2017. NIPS Accepted Papers Stats. https://medium.com/machine-learning-in-practice/nips-accepted-papers-stats-26f124843aa0. Accessed: 2020-02-20.

[3] 2017. A web app for ranking computer science departments according to their research output in selective venues. https://github.com/emeryberger/CSrankings. Accessed: 2020-02-21.

[4] 2018. sotawhat. https://github.com/chiphuyen/sotawhat. Accessed: 2020-02-20.

[5] 2019. Browse State-of-the-Art. https://paperswithcode.com/sota/. Accessed: 2020-02-21.

[6] 2020. Microsoft Academic. https://academic.microsoft.com/home. Accessed: 2020-02-21.

[7] Shane Dawson, Dragan Gašević, George Siemens, and Srecko Joksimovic. 2014. Current state and future trends: A citation network analysis of the learning analytics field. In *Proceedings of the fourth international conference on learning analytics and knowledge*. 231–240.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[9] Zhen Guo, Zhongfei Zhang, Shenghuo Zhu, Yun Chi, and Yihong Gong. 2009. Knowledge discovery from citation networks. In *2009 Ninth IEEE international conference on data mining*. IEEE, 800–805.

[10] Norman P Hummon and Patrick Dereian. 1989. Connectivity in a citation network: The development of DNA theory. *Social networks* 11, 1 (1989), 39–63.

[11] Michael Ley. 2009. DBLP: some lessons learned. *Proceedings of the VLDB Endowment* 2, 2 (2009), 1493–1500.

[12] Gerry McKiernan. 2000. arXiv. org: the Los Alamos National Laboratory e-print server. *International Journal on Grey Literature* (2000).

[13] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[14] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. *The pagerank citation ranking: Bringing order to the web.* Technical Report. Stanford InfoLab.

[15] Jie Tang. 2016. AMiner: Toward understanding big scholar data. In *Proceedings of the ninth ACM international conference on web search and data mining*. 467–467.

[16] Rita Vine. 2006. Google scholar. *Journal of the Medical Library Association* 94, 1 (2006), 97.

[17] Kuansan Wang, Zhihong Shen, Chi-Yuan Huang, Chieh-Han Wu, Darrin Eide, Yuxiao Dong, Junjie Qian, Anshul Kanakia, Alvin Chen, and Richard Rogahn. 2019. A Review of Microsoft Academic Services for Science of Science Studies. *Frontiers in Big Data* 2 (2019), 45.

[18] Stephen L Wright, Michael A Jenkins-Guarnieri, and Jennifer L Murdock. 2013. Career development among first-year college students: College self-efficacy, student persistence, and academic success. *Journal of Career Development* 40, 4 (2013), 292–310.