Progress Report (CSE 519): Ranking arXiv papers

Amol Damare
adamare@cs.stonybrook.edu
SBU ID: 107914028

Punit Mehta
pmmehta@cs.stonybrook.edu
SBU ID: 111461860

November 15, 2017

1 Introduction

Ranking of scientific paper is non-trivial task even for an expert. In our project proposal we had presented an idea for ranking the papers in arxiv. Specifically we had identified 3 main subproblems. Namely

1. Ranking the papers which are already published and are available on arXiv.
2. Ranking the papers which are not published anywhere but available on arXiv from quite some time.
3. Ranking new papers which are recently uploaded on arXiv and are not published or accepted anywhere yet.

We also described the data set we were going to use and our approach to solve these subproblems. Although our dataset remains the same, we have since changed our approach for solving these problems and evaluation criteria. We will be using an approach based on collective behavior of papers in citation graph as suggested by Prof. Skiena.In this progress report we will briefly describe the dataset and give in detail the approach we will be using to achieve goal of this project. Finally we will also present our baseline model and present it's evaluation. Next section describes the related research work that has influenced our approach.

2 Social behavior of scientific papers

A paper can be related to another paper via various different ways e.g. one paper cites other, both papers cite same paper, they are accepted in same journal/conference, they can have same authors etc. We can use these relationships to form a network or graph of papers. When we think of scoring these papers we can say that score of a paper will be dependent on scores the papers it is connected to in graph. There is correlation between scores of connected papers. So when we have a new paper we can score it by looking at the scores of the papers it is connected to. In this way network of papers behaves similar to any social network. This relationship allows us to apply recent developments in social relational learning [1] ,[11]to our problem. Also it allows us model our problem as a relational learning problem [3], [2].

Algorithm 1 High level algorithm to get ranks of papers

1: Create a graph of papers capturing relations such as citations, authors, conferences etc.
2: Get latent social embeddings for the vertices of this graph.
3: Create train and test data using some helper ranking function.
4: Train a model on this embedding to predict the score/rank of paper.

3 Approach

The basic approach that will be used in scoring papers is given in algorithm 1 Now we will go in details of each step in this algorithm. Step 1 is trivial for our baseline model. We have created the graph using just citations as criteria to get a citation graph of of papers. We can easily extend this graph to include other relationships as well, as we have the necessary data to build such graph. Figure 1 shows the fields we have for all the papers.

Field Name	Field Type	Description	Example
id	string	MAG or AMiner ID	53e9ab9eb7602d970354a97e
title	string	paper title	Data mining: concepts and techniques
authors.name	string	author name	Jiawei Han
author.org	string	author affiliation	department of computer science university of illinois at urbana champaign
venue	string	paper venue	Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial
year	int	published year	2000
keywords	list of strings	keywords	[“data mining”, “structured data”, “world wide web”, “social network”, “relational data”]
fos	list of strings	fields of study	[“relational database”, “data model”, “social network”]
n_citation	int	number of citation	29790
references	list of strings	citing papers' ID	[“53e99ef4b7602d97027c2346”, “53e9aa23b7602d970338fb5e”, “53e99cf5b7602d97025aac75”]
page_stat	string	start of page	11
page_end	string	end of page	18
doc_type	string	paper type: journal, book title…	book
lang	string	detected language	en
publisher	string	publisher	Elsevier
volume	string	volume	10
issue	string	issue	29
issn	string	issn	0020-7136
isbn	string	isbn	1-55860-489-8
doi	string	doi	10.4114/ia.v10i29.873
pdf	string	pdf URL	//static.aminer.org/upload/pdf/1254/ 370/239/53e9ab9eb7602d970354a97e.pdf
url	list	external links	[“http://dx.doi.org/10.4114/ia.v10i29.873”, “http://polar.lsi.uned.es/revista/index.php/ia/ article/view/479”]
abstract	string	abstract	Our ability to generate…

Figure 1: Paper data model and its fields [9]

3.1 Latent social representation of graph

In step 2 of our algorithm 1 we need to get latent social representation of the graph we constructed in step 1. We have used ”DeepWalk” algorithm presented in [1] to get the social embedding of the citation graph. ”DeepWalk” applies deep learning techniques to learn the latent social representations of each vertex in the network. To do this [1] has presented idea that random walks in graph is similar to short sentences in a special language whose vocabulary is the set of vertices of the network. Hence we can apply same deep learning techniques that are used in neural language models to learn latent representation of each vertex. We will be using the same algorithm to get the representation of each vertex in our citation graph. There are other methods to get social representation of a vertex such as Spectral clustering[11] which uses spectral graph theory(eigen values of Laplcian of graph), [2] uses eigen values of Modularity matrix of graph etc. Deepwalk works well on all types graph and works very well even if the graph is sparsely labeled in relational classification task [1], which is exactly kind of representation we would need to rank the papers as we have very huge graph(166,192,182 vertices) with very sparse labelling (as we will be using experts to rank the few papers).

3.2 Intuitive Ranking Helper Function

We don't have any intuitive way to rank or score papers based on social representations we got from running DeepWalk on the graph. Also it is very difficult to get a generalized scoring/ranking function that works universally well. But if we just consider one of the parameters to decide if paper is good or not then we can have a intuitive ranking function that we can easily explain. But this ranking function does not generalize well. In this project we propose an idea that we will generalize this naive but intuitive ranking function by training some regression model on latent social representation of the graph. For baseline we are considering a naive function that ranks a paper based on conference or journal it was accepted in. We can easily get the impact factors of conferences and journals. We assume that if venue at which paper is published is good then paper is as good as the venue itself. Using this naive intuitive function we can sparsely get the scores of paper in our graph. We can then train a regression model on this data to generalize the ranking function to look at other aspects of the paper as well. In our base line model, graph is constructed using citations and our ranking helper function is only considering conferences that these papers were published in.

3.3 Prediction of the score/rank of a paper

To rank/score the papers, our initial approach was to get the feature vectors from deepwalk and then come up with a function of this feature vector to get the score of the paper. But after getting these feature vectors for each paper we came to conclusion that its very hard to interpret what each of this dimension represent intuitively. e.g. does feature 0 represent citation relation or does it represent author relation etc. So it was hard to come up with a function which would make sense and give us good results for all the papers. We have a powerful tool in our hand in the form these feature vectors we got from deepwalk, and we want to utilize their full potential. So we tried to model ranking as a regression problem. Let's say we have ranks/scores of some of the papers then we can easily train a model on feature vectors to predict the rank or score of the rest of the papers. And as show in deepwalk paper such model works well even in very sparsely labeled dataset such as youtube [1]. So for our baseline model we ranked few papers manually. We have used the ranks of conferences or journals to get estimate of rank/score of paper that was published in that particular conference/journal. We have made an assumption that if conference/journal is good then paper accepted in that journal/conference will be a good paper too. We used this logic to create the training dataset for our models.

4 Baseline model and Evaluation

Our main goal of baseline model is to test whether our approach presented in algorithm 1 is correct or not. Also we wanted to test how accurately does the representations from deepwalk capture relationships not implicitly used in building the graph. To answer both these question we created our baseline model by creating graph using citations of a paper (i.e. there will be edge between 2 papers if one paper cites the other). We designed our helper ranking function naively by assuming that paper is as good as venue it is published in. Using these sparsely labelled papers we trained a linear regression model on feature vectors we got by running DeepWalk on our graph. Algorithm 2 gives detailed algorithm for baseline model.

Now we will present the results of our experiment.

Algorithm 2 Algorithm for Baseline model

Create graph $G(V, E)$ using citations.
G_{train}, G_{test}=Split(G) ▷ Get random papers from graph to create training and testing datasets
IntuitiveRankingHelperFunction(V_{train})
IntuitiveRankingHelperFunction(V_{test})
Initialize F as array of feature vectors
F=DeepWalk(G) ▷ This runs deepwalk on graph to get vector representation of each vertex
model=LinearModel(G_{train}, F) ▷ Initialize a linear model
Evaluate *model* on G_{test}
ranks=model.predict(G)
return ranks

Algorithm 3 Algorithm for intuitively ranking papers

procedure IntuitiveRankingHelperFunction(V)
    Initialize VenueRank table ▷ this table has ranks of each conference and/or journal
    for $vinV$ **do**
        $rank[v] = rank[Venue[v]]$ ▷ Naive way to rank the paper, Paper is as good as conference/journal it is published in
    return $rank$

4.1 Experiment

We have a data set consisting of 166,192,182 papers which is a huge number to deal with for baseline model, so we decided to limit ourselves to papers just in computer science field. We have selected total of $10,000$ papers for this phase of the project. We divided this dataset in to 70%/30% training and testing splits. Table 1 shows the results of our training. We got 14% error on our test datasets. Considering that we constructed graph using just citations and ranking function used for training was based on conferences , we think this is an impressive result for baseline model. Figure 2 shows the histogram of errors for our model. It shows scores given by our model varies within 1 or 2 points of actual score. This is desirable property for our model as we want the model to generalize and take in to consideration other affiliations of the paper as well while scoring. R^2 value of our model is 0.095534, which is indicates that our model not that good and can be improved. Figure **??** shows the p-test result for our model. Our p-value is. 0.043, which indicates that our model is good and we can reject the null hypothesis.

From the evaluation of our model it is clear that hypothesis we made are actually true. We can say that approach we are using works and we can make further progress by improving our model to get good results. We also shown that "DeepWalk" indeed captures the relationship which are not explicit at the time of creation of graph. We created the graph using citations and still got good results when we trained the model to basically predict the relationship based on conferences.

Data Set	Training	Testing
Mean Error	1	13.97%
Median Error	1	13.87%

Table 1: Error rates for linear regression base model

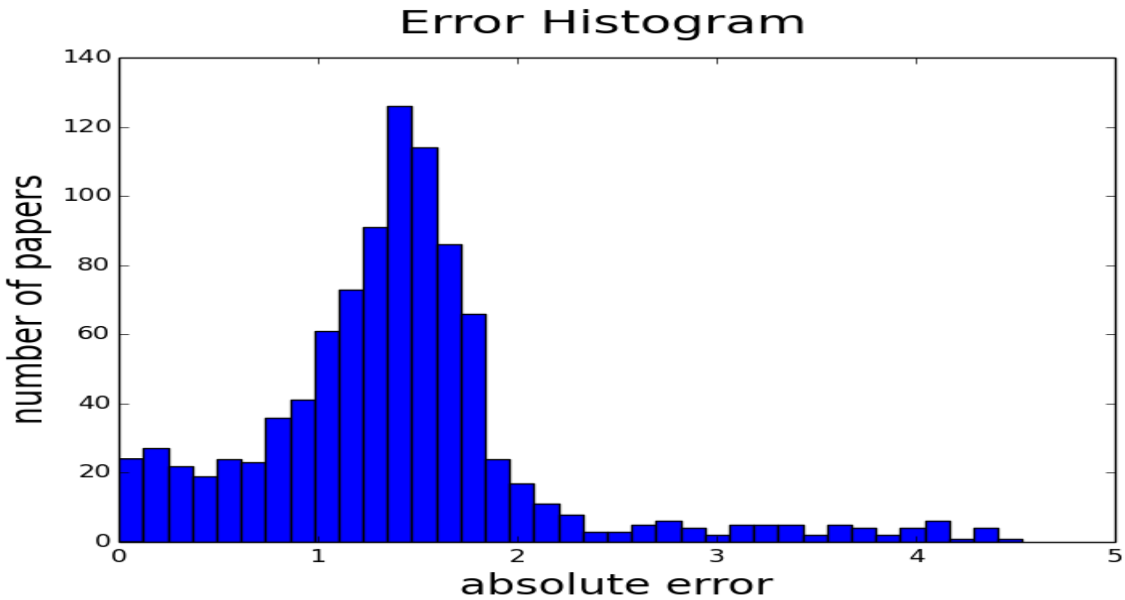

Figure 2: Error histogram of baseline model

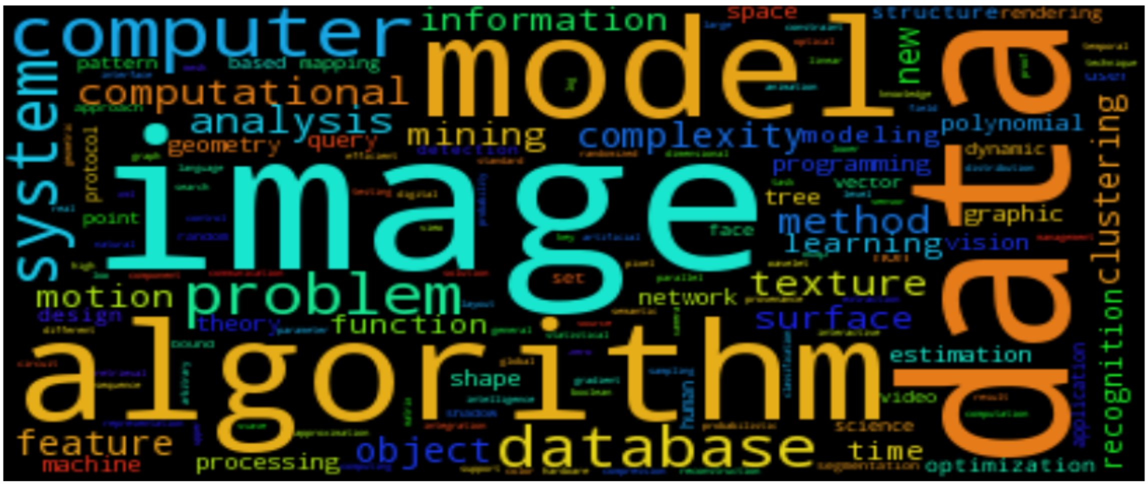

Figure 3: Most prominent topics in computer science

5 Future Work and Conclusion

As seen from results we presented we can see that ranking function really depends on our helper function and the graph we constructed using papers. We can see now a few ways to improve our model

1. Instead of using a helper function we can get an expert to rank or we can rank ourselves, a small number of papers, so we get a more robust initial estimate of ranks of papers. We think this kind of function will generalize better using our approach.

2. We can construct graph using various other homophily such as authors, conferences etc.

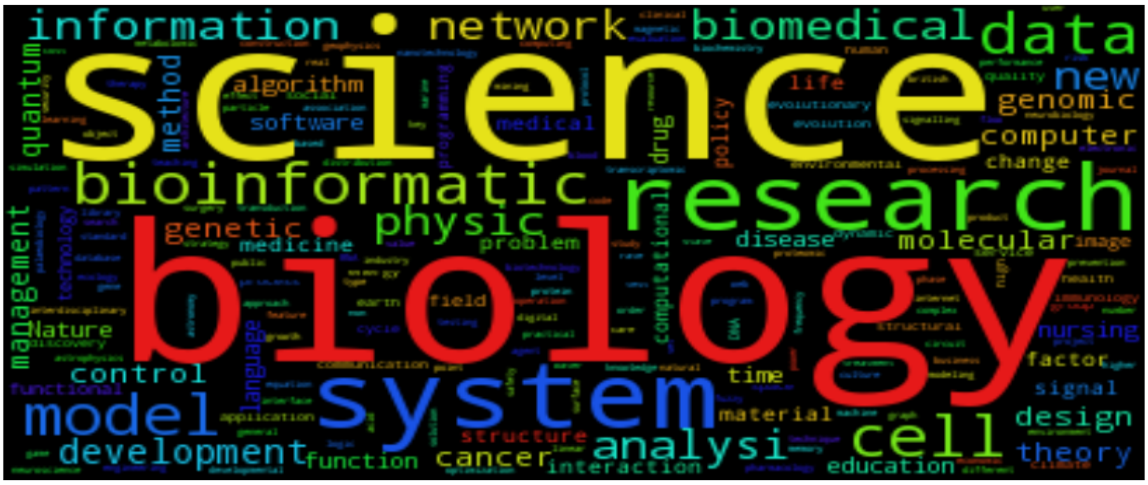

Figure 4: Most prominent general topics

   We think more relationship we capture in our initial graph, we will get better results for papers which do not all the information such as citations, conferences etc.

3. We can also improve model by using non-linear regression model in last step of our algorithm.

We will try above approaches in an attempt to improve our baseline model. We also think, we will better results if we couple the training of DeepWalk and regression linear model together, similar to the deep learning models used in NLP tasks. And if time permits we want to try this approach. This will also require deep understanding of DeepWalk algorithm at coding level.

References

[1] Perozzi, B. ,Al-Rfou, R.,Skiena, S. DeepWalk: Online Learning of Social Representations Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining KDD '14

[2] Tang, Lei and Liu, Huan Relational learning via latent social dimensions Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining,2009

[3] Tang, Lei and Liu, Huan Scalable learning of collective behavior based on sparse social dimensions Proceedings of the 18th ACM conference on Information and knowledge management,2009

[4] J. E. Hirsch. An index to quantify an individual's scientific research output, 2005, Proc.Nat.Acad.Sci.46:16569,2005; arXiv:physics/0508025. DOI: 10.1073/pnas.0507655102.

[5] Egghe, L. Theory and practise of the g-index Scientometrics (2006) 69: 131. https://doi.org/10.1007/s11192-006-0144-7

[6] Page, L., Brin, S., Motwani, R., Winograd, T. The PageRank citation ranking: Bringing order to the web (Technical Report). Stanford InfoLab.

[7] Garfield, E. Citation analysis as a tool in journal evaluation Science,178,471-479.

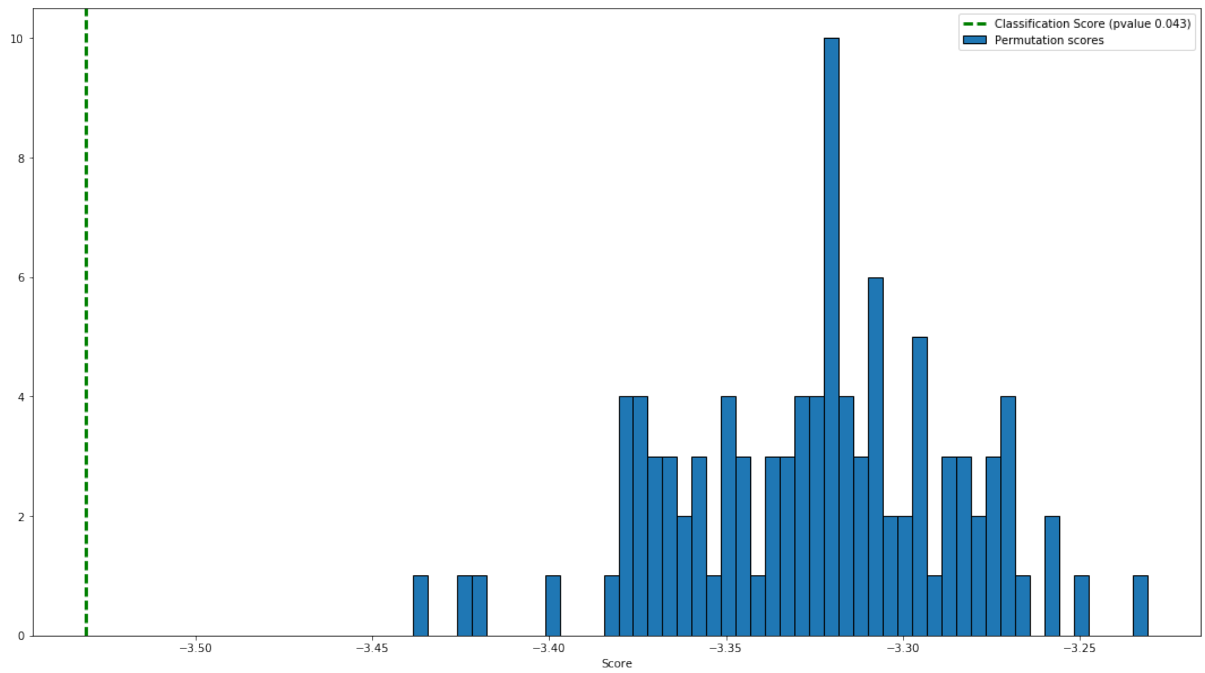

Figure 5: P-test for model

[8] Pinski, G., Narin, F. Citation influence for journal aggregates of scientific publications: Theory, with application to the literature of physics Information Processing and Management, 12(5), 297-312

[9] Microsoft Academic Graph https://www.openacademic.ai/oag/

[10] arXiv.org https://arxiv.org/

[11] Tang, Lei and Liu, Huan Leveraging social media networks for classification Data Mining and Knowledge Discovery,2011,Springer

List of Figures