Predicting Scientific Success Based On Coauthorship Networks

Guosheng Hu
School of Electronic,
Information and Electrical Engineering
Shanghai Jiao Tong University
Shanghai, China 200240
Student ID: 515030910608
Email: hgs1217@sjtu.edu.cn

***Abstract*—The recurrent neural networks tool is getting popular in the research field of time series data prediction, due to its impressive model generalization. The objective of this report is to apply this technology to scientific success prediction based on coauthorship networks. This work has been done in classical machine learning methods, so I try to make use of the neural networks to gain a better performance. It is found that this model achieves around 70% accuracy, showing the feasibility of this method.**

I. Introduction

Today, there are a large quantity of methods in measuring the achievements of scientific researches. The number of total paper citation, the reputation of journal paper published, or even some other high level indices, like h-index [1] or g-index [2], are all metrics which can evaluate the performance and productivity of scientific researches.

However, few works have been done on prediction of scientific success. It is widely accepted that working with successful people leads to a shortcut to success, and so is in the research field. Therefore, coauthorship networks perhaps play an important role in scientific achievements although scientific success depends more on the research itself.

II. Existed Work

Some previous works have been done on this topic, among which the typical one is done by Sarigol *et al.* [3]. Sarigol mainly study how centrality in coauthorship network affects the paper citations. The paper shows that the more central the author is, the more successful his research is.

It is shown that if a paper is authored by an author among the top 10% centrality, the paper will be among 10% most cited paper five years later. This principle is used as the rule in naive Bayes classifier [4], which results in the prediction with about 36% accuracy. What's more, in order to gain a better performance, the paper applies another machine learning method called Random Forest classifier [5], which leads to about 60% accuracy.

Fig. 1. Illustration of correlation between citation success and centrality in the coauthorship network for year 2002 and 2007. [3]

Sarigol's work provides a feasible way in predicting scientific success based on coauthorship networks. Nevertheless, there are still much left for improvement. First,

the prediction accuracy is not ideal, only with about 60% accuracy, and thus something can be done for performance improvement. Besides, the previous work only tries traditional machine learning methods. However, deep learning is on fire these years, which is famous for its better performance and more extensive usage. Therefore, in this report, I try to apply recurrent neural networks to train a better model for scientific success prediction based on coauthorship networks.

III. Materials and Approaches

I aim to realize the scientific success prediction based on coauthorship networks through recurrent neural networks. When given an input, which is the coauthorship networks of a certain period of time, my model can give the prediction of paper as the result, telling that where the paper is ranked. The model is trained on the basis of a huge dataset, which has gone through a pretreatment process. The main structure of the whole system is shown in **Figure 2**.

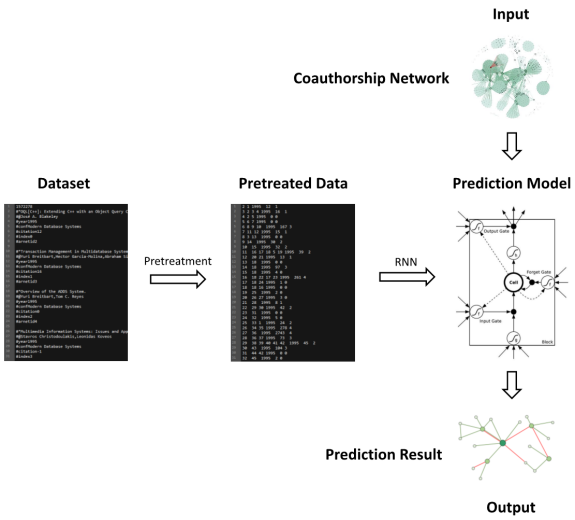

Fig. 2. The general architecture of my work, involving the dataset pretreatment and prediction by neural network model. The input is the previous several years' annual citations of different authors, representing the coauthorship networks. The output is the predictions whether the papers will achieve scientific success or not. The input figure is cited from Sarigol *et al.* [3].

A. Dataset with pretreatment

My dataset is constructed by DBLP Citation Network V5 from Aminer [6]. DBLP Citation Network consists of $1,572,277$ papers and $2,084,019$ citation relationships ranging from 1936 to 2012. The DBLP dataset has the size of 772MB, involving the papers' titles, authors, citation numbers, published years, abstracts and so on. The year and citation number distributions are shown in **Figure 3**, demonstrating that a huge number of papers are published within 1971 to 2011, with the proportion of 99.01%, and most papers are cited below 50 times, with the rate of 93.57%.

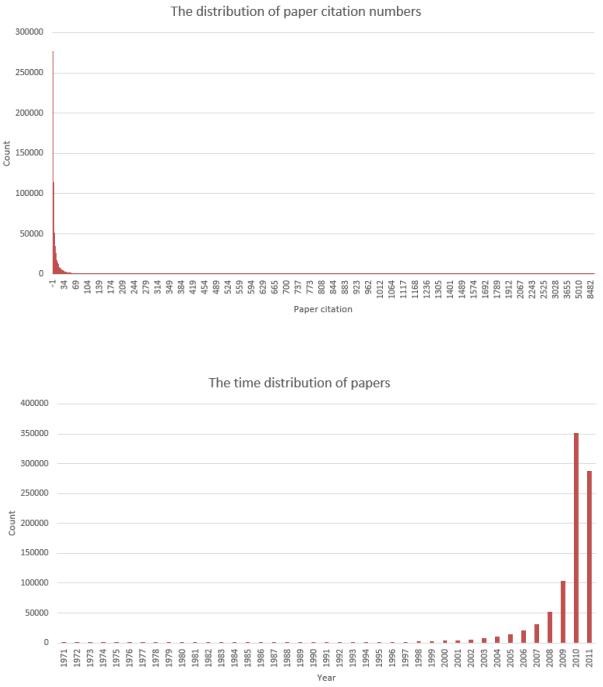

Fig. 3. These two figures are the citation number distribution of papers and the time distribution of papers. We can find that most papers have low citations as well as most papers are published in the $21st$ century.

Therefore, I take the papers published within 1971 to 2011 as my network dataset so that it prevents sparsity which may do harm to the performance. Also, I classify the papers into five categories according to their citation numbers. The citation thresholds are the proportion of 50%, 75%, 90% and 97% among all papers. Besides, I have to construct a metric which reflects the achievements in coauthorship networks each year. I choose the total citation number of papers published each year as the evaluation criterion. If author A and B are both authors of a certain paper, both of them add the paper's citation number that year. As a result, each author gets the annual citations of 40 years. However, there are over $900,000$ authors, and many of them have published only few papers or gain very few citations, which increases the sparsity of the problem. Thus I make a selection to prune the authors with low productivity and reduce the number of authors to about $10,000$. The annual citations of a certain author is also divided into five categories, the same to

paper citations.

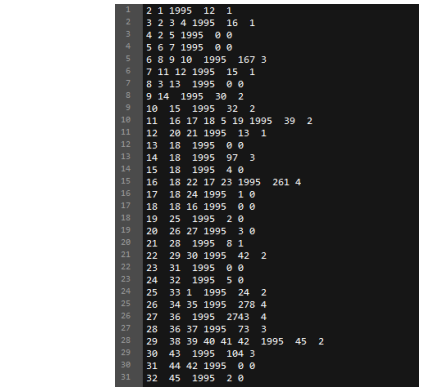

Fig. 4. The dataset after pretreatment. The first figure shows the paper data, with the author ids, published year and citations. The second figure depicts the author data, with annual citations of 40 years.

B. Neural Network

Since my goal is to predict whether a certain paper can be successful according to the scientific achievements of the previous five years, the problem can be described as below:

Given the paper authors' annual citation levels of the previous five years before the published year, the model outputs the citation level of the paper in the future.

As this is a prediction problem based on time series data, recurrent neural network is a good idea. Recurrent neural network (RNN) [7] is a class of artificial neural network where connections between nodes form a directed graph along a sequence. This allows it to exhibit dynamic temporal behavior for a time sequence. Recurrent neural network is widely used in handwriting recognition and speech recognition.

Although recurrent neural network is a strong implement in tackling the prediction problem, it contains some defects such as gradient disappearance and gradient explosion. To solve them, a structure called Long short-term memory (LSTM) [9] is proposed, which can remember values over arbitrary time intervals. LSTM constructed network can have a better performance in time series prediction, so my model is composed of LSTM layers.

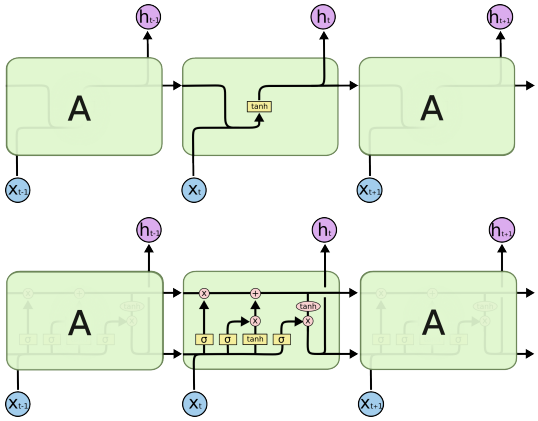

Fig. 5. The structure of recurrent neural network and long short-term memory. RNN only contains a simple tanh layer, while LSTM contains 3 sigmoid layers as well as one tanh layer. An LSTM structure is composed of a cell, an input gate, an output gate and a forget gate. [8]

I have tried different numbers of LSTM layers in the model, so as to compare the performance with each other. Beside the LSTM layers, weighted loss is used in loss function. For the reason that different citation levels take different proportions, weighted loss with suitable proportion is essential in training, or otherwise, the model will turn to extreme one class prediction. In addition, we should also consider the accuracy of all classes rather than the total one, as predicting all papers to be the bottom 50% can still leads to 50% total accuracy. This way also fits our subjective judgment in reality.

IV. Results

As mentioned before, I have tried various structures of the model, involving the layer number, timestep and so on. The number of nodes in LSTM has been set to 256 or 1024, and the LSTM layers have been set to 1 or 3. These parameters affects the size of fields which makes an effect on prediction accuracy. The timestep is defined as the length of time taken into account before the paper is published. The default timestep mentioned before is 5, I have also tried 10 to see whether I can get a better prediction. All models are trained for 3000 epoches, with the batch size of 1000. Results are shown in **Table 1**.

Another type of trials is about the number of classes in prediction. The annual citation numbers and paper citations are both classified into five classes mentioned before. The changes are merging some classes into one, and that is, five classes are merged into two, with the

TABLE 1. These are the results of different models, involving the total loss, total accuracy, bottom paper accuracy and top paper accuracy. I have tried different layer numbers (L), different node numbers in each layer (N) and different timestep (Y). The performance comparison shows that the 1024 single layer structure relatively performs well, along with the fact that 10-year timestep predicts well in top papers, in spite of its low total prediction accuracy.

Structure	Loss	Total accu	Bottom pred	Top pred
1L+256N+5Y	0.5428	0.7398	0.7574	0.4068
3L+256N+5Y	0.5422	0.7408	0.7583	0.4056
1L+1024N+5Y	0.5374	**0.7602**	**0.7808**	0.3702
1L+1024N+10Y	**0.5301**	0.6306	0.6278	**0.5927**

TABLE 2. These are the results of different class numbers in input and output, based on the 1024 single layer with 5-year timestep. The performance comparison shows the model of 5 in classes with 2 out classes predicts well with the top 10% papers, while the model of 2 in classes with 2 out classes has a relatively higher accuracy.

Structure	2 in + 2 out	5 in + 2 out	5 in + 5 out
Loss	0.5374	**0.5244**	1.1245
Total accuracy	**0.7602**	0.6346	0.5142
Bottom 50% Accuracy			0.6418
50% – 75% Accuracy	0.7808	0.6315	0.0000
75% – 90% Accuracy			0.0448
90% – 97% Accuracy	0.3702	0.5858	0.1064
Top 3% Accuracy			0.4674

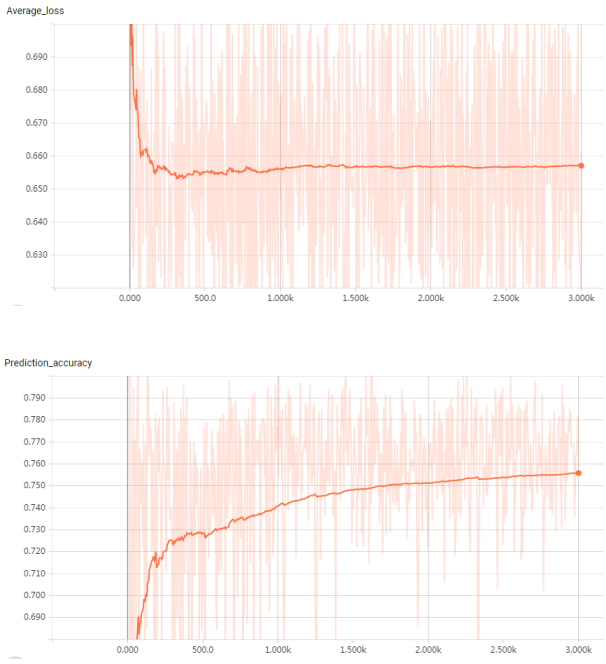

Fig. 6. These are the training data diagrams of 1024 single layer model with 5 timestep, which involves average loss and prediction accuracy. The figure shows the model has already converged.

threshold 90%. Thus the papers are categorized into the top 10% group and bottom 90% group, which is the same to the definition of scientific success in Sarigol's paper [3]. These results are shown in **Table 2**.

From the results, we have an about 70% accuracy in scientific success prediction. The best model to predict top papers is the 10-year timestep one, which reaches nearly 60% precision in top papers prediction. In comparison, Sarigol's model based on Random Forest achieves 60% total precision, which is slightly lower than my neural network model.

V. Conclusion

Like other time series prediction problem, scientific success prediction based on coauthorship networks is also done well with the help of neural networks. The classical machine learning classifiers can deal with the prediction problem with good mathematical explanation, but their performance is not decent. Neural network method can further improve the prediction accuracy. Nevertheless, my pretreatment is relatively simple. Perhaps other forms of pretreatment may lead to a even better performance.

References

[1] S. Alonso, F. J. Cabrerizo, E. Herrera-Viedma, and F. Herrera, "h-index: A review focused in its variants, computation and standardization for different scientific fields," *Journal of Informetrics*, vol. 3, no. 4, pp. 273–289, 2009.

[2] L. Egghe, "Theory and practice of the g-index," *Scientometrics*, vol. 69, no. 1, pp. 131–152, 2006.

[3] E. Sarigol, R. Pfitzner, I. Scholtes, A. Garas, and F. Schweitzer, "Predicting scientific success based on coauthorship networks," *Epj Data Science*, vol. 3, no. 1, p. 9, 2014.

[4] I. Rish, "An empirical study of the naive bayes classifier," *Journal of Universal Computer Science*, vol. 1, no. 2, p. 127, 2001.

[5] T. K. Ho, "Random decision forests," in *International Conference on Document Analysis and Recognition*, 1995, p. 278.

[6] Aminer, "Aminer citation network dataset," *https://www.aminer.cn/citation*, 2011.

[7] W. contributors, "Recurrent neural network," *https://en.wikipedia.org/wiki/Recurrent_neural_network*, 2018.

[8] Orisun, "Rnn and lstm," *https://www.cnblogs.com/zhangchaoyang/articles/6684906.html*, 2017.

[9] W. contributors, "Long short-term memory," *https://en.wikipedia.org/wiki/Long_short-term_memory*, 2018.