Prediction of academic mentorship

Yi Fang, Kaibin Zheng

Abstract

We use deep neural network to predict mentorship for Acemap. Our newly optimized dataset is better than previous dataset and 63 features is better than previous 22 features. Besides, our newly developed network performs better than previous 2*Dense network. Our contributions are

- Improve and perfect extracted features of (advisor, student) pairs.
- Optimize the counter-example dataset for training
- Expand the function by three-classification model
- Train a mentorship prediction model with high accuracy(98% compared to the previous 95%)

Contents

I. Introduction

First of all, we have to explain the reason for choosing the topic *Mining of academic mentorship* instead of the original chosen topic *Personalized User Profiling for recommdendations*. Actually, we are supposed to extract academic profile in the early stage as figure 1 shows. But soon we realized that a large number of websites, for example, Acemap, are doing the same work, and it's not wise to repeat it on our own. However, there's no website, even Acemap, perform excellently on extraction of academic mentorship. Then we are thinking about help to improve the accuracy of mentorship extraction.

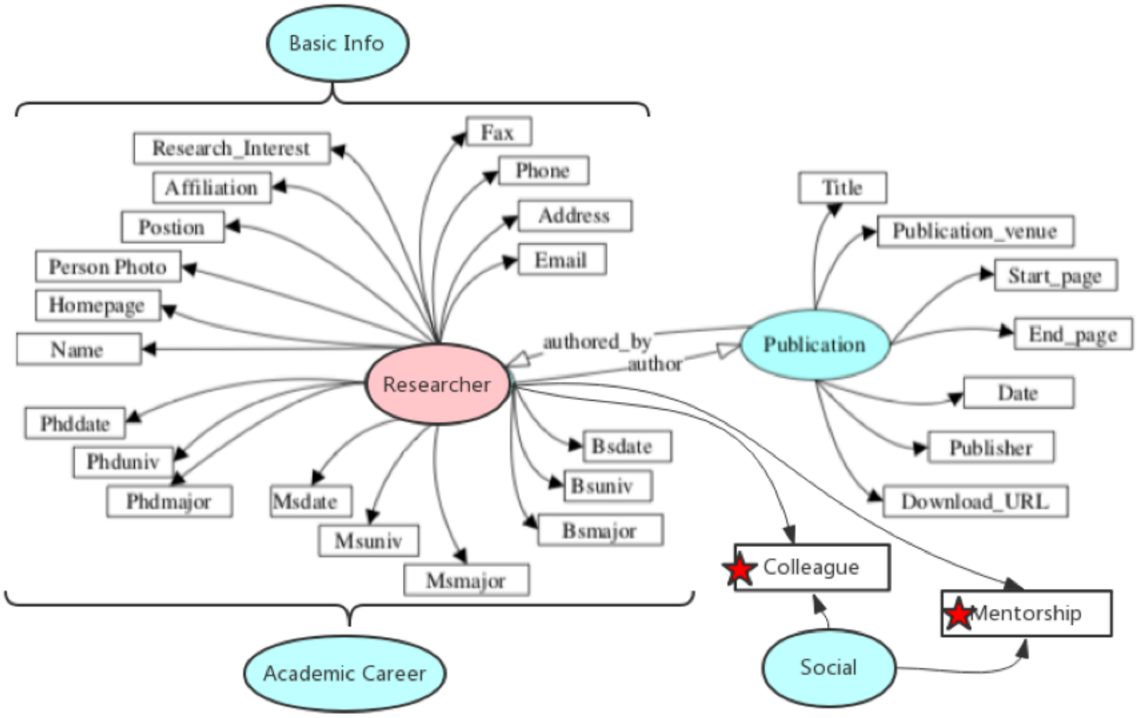


Fig. 1: Expected academic profile.

The process of our work is illustrated in figure 2 with details illustrated in later sections.

II. Improvements in extracted features

A. How to extraction features for (advisor, student) pairs

First, all of our work is based on the groundtruth of mentorship collected by Acemap group. Scholars provided their mentors' or students' basic information. And then by mapping there

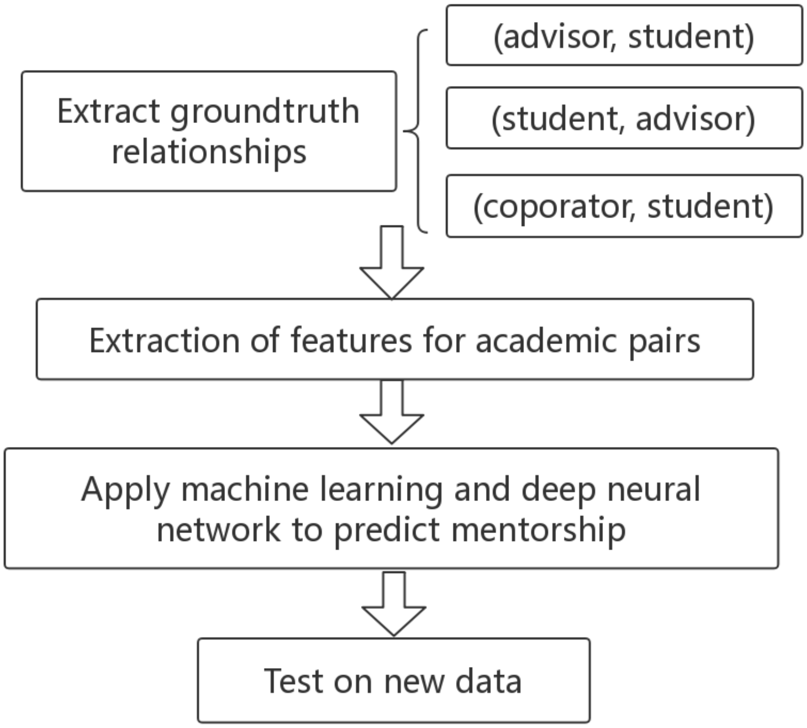

(advisor, student)
(student, advisor)
(coporator, student)
Extract groundtruth relationships
Extraction of features for academic pairs
Apply machine learning and deep neural network to predict mentorship
Test on new data

Fig. 2: Technology roadmap for mining of academic mentorship.

information to scholar's ID in MySQL database, we can get the groundtruth (advisor, student) pairs for training.

Items in groundtruth data is shown as follows:

- advisor ID and student ID
- Start year and finish time for the relationship

Then we search MySQL database for publication information for scholars in the groundtruth mentorship pairs and calculate features, such as number of publication, co-publication and number of cooperators.

B. Problems in previous features

Previous features include:

- number of co-published papers, length of years for co-publication

- quantity and time of previous publication for the advisor and the student before the mentorship begins
- during the relationship on, the quantity of papers cooperate with other scholars and the number of cooperators
- the number of co-published papers with advisor(student) as the first author, second author and third author

But these features do not illustrate how the author byline distributes for (advisor, student) pairs specifically.

C. Advanced feature extraction

To solve the aforementioned problem, we add new features to describe the relationship with more details. And since a great number of mentors will be the second author or third author of their students' papers and some mentors would like to be the last author, we consider new features, for example quantity of papers with different author byline combinations as the figure 3 shows.

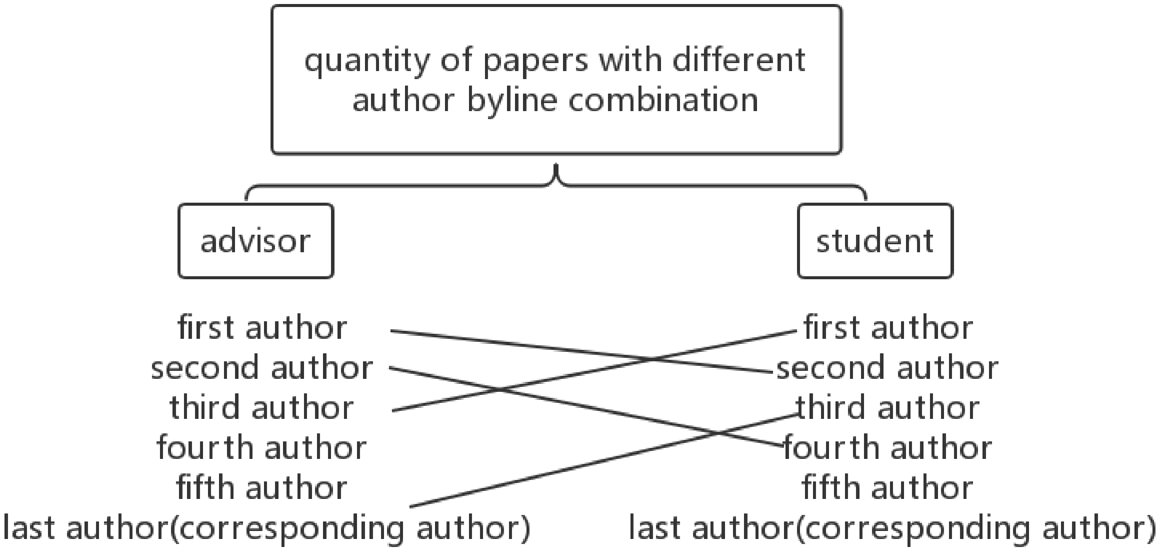
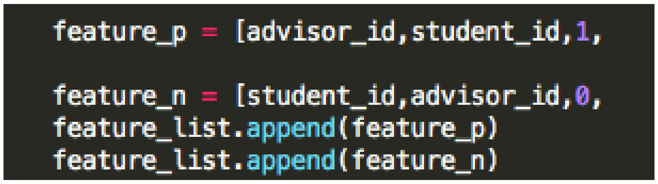


Fig. 3: Features with different author byline combination.

III. Improvements in counterexample dataset

The previous counterexample dataset is inversed (advisor, student) pairs, that is (student, advisor) pairs, and their features. Obviously, the previous members didn't take cooperators who are not the students' mentor into consideration.



```
feature_p = [advisor_id,student_id,1,

feature_n = [student_id,advisor_id,0,
feature_list.append(feature_p)
feature_list.append(feature_n)
```

Fig. 4: Previous dataset with (student, advisor) pairs as counterexamples.

By searching the database for other authors of the student's papers except for his mentors, we derive a new relation mapping cooperators to students. Then our dataset contains three parts as figure 5 shows.

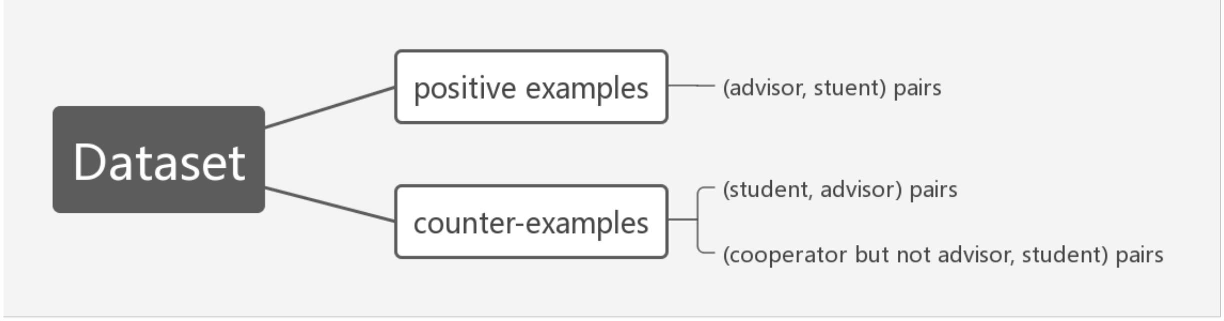


Fig. 5: Optimized dataset.

Actually, we spend lots of time on improving the dataset for the Acemap group did not extract authors of students correctly.

IV. Train a model for mentorship prediction

Deep neural network tools we used are as follows:

- Keras. Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result with the least possible delay is key to doing good research.
- Scikit-learn. Sklearn is a machine learning tools with features such as simple and efficient tools for data mining and data analysis, accessible to everybody, and reusable in various contexts and built on NumPy, SciPy, and matplotlib

Since it's a classic binary classification problem, we take some SVM and ensemble methods, such as forests of randomized trees, gradient tree boost and XGBoost into consideration, especially newly proposed XFGBoost. But these networks cannot perform well on our problem.

We prefer to account this phenomenon for that the number of features of our np matrix for training is not as many as those problems with thousands of features. Actually, we find that simple fully connected layers perform better than complex networks.

Then we propose a simple network as figure 6 shows. It's a two-layer network with densely-connected layer as the input layer with the activation function tanh and the fully-connected network as the output layer. In order to avoid over-fitting, we apply Dropout layer to the input. Dropout layer consists in randomly setting a fraction rate of input units to 0 at each update during training time, which helps prevent over-fitting. The output layer of the network is set to be the fully connected layer with output dimension being equal to the number of classification, i.e.two for classification of mentorship and three for classify of mentorship and cooperation. So we add a fully connected layer with 2 units as the output layer. After that, we calculate the accuracy and loss in the testing set and obtain the result. Later, we predict the relationship in new pair of authors. The network is as figure 6 shows.

While tuning, we find that adding layers help a lot, as the number of fully connected layers represents a the order of feature combinations For example, one fully connected layer means the linear combination of features. Besides, the units of each layer, especially the first fully connected layer is essential for final result. It's because the more units, the more combinations of features are tried. And the more combinations, the larger possibility for our network to find the underlying influence factors.

After experiments, we propose another useful network which achieves the accuracy of 99% on test dataset, shown in figure 7. The first fully connected layer contains 1000 units. And to

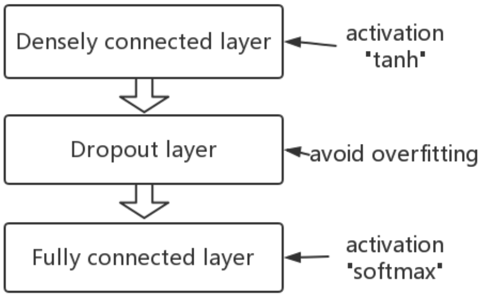


Fig. 6: Newly optimized dense connected network with dropout layer.

avoid the difference in performance of our network on training dataset and test dataset, we add a dropout layer with rate of 0.1. Then comes 7 fully connected layers with units of 500, 200, 100, 50, 24, 12, and 8. Note that these fully connected layer activated with function ”relu”, while the output layer activated with function ”softmax”.

Our model performs well on the new dataset as table I shows. The first column *Dataset* are explained as item *Ex* represents for the previous dataset without cooperators and item *Optimized* is for optimized dataset with cooperators. The second column *Features* represents for the number of previous 22 features and newly optimized 63 features. The third column *Network* represents for the construction of deep neural networks. Note that Complex densely connected network is for the network illustrated in figure 7. The last four column show the loss and accuracy of training set and test set.

We can draw a conclusion that newly optimized dataset is better than previous dataset and 63 features is better than previous 22 features. As you can see, our newly network perform better than previous 2*Dense network.

V. Conclusion

In a brief, our contributions are

- Improve and perfect extracted features of (advisor, student) pairs.
- Optimize the counter-example dataset for training
- Expand the function by three-classification model

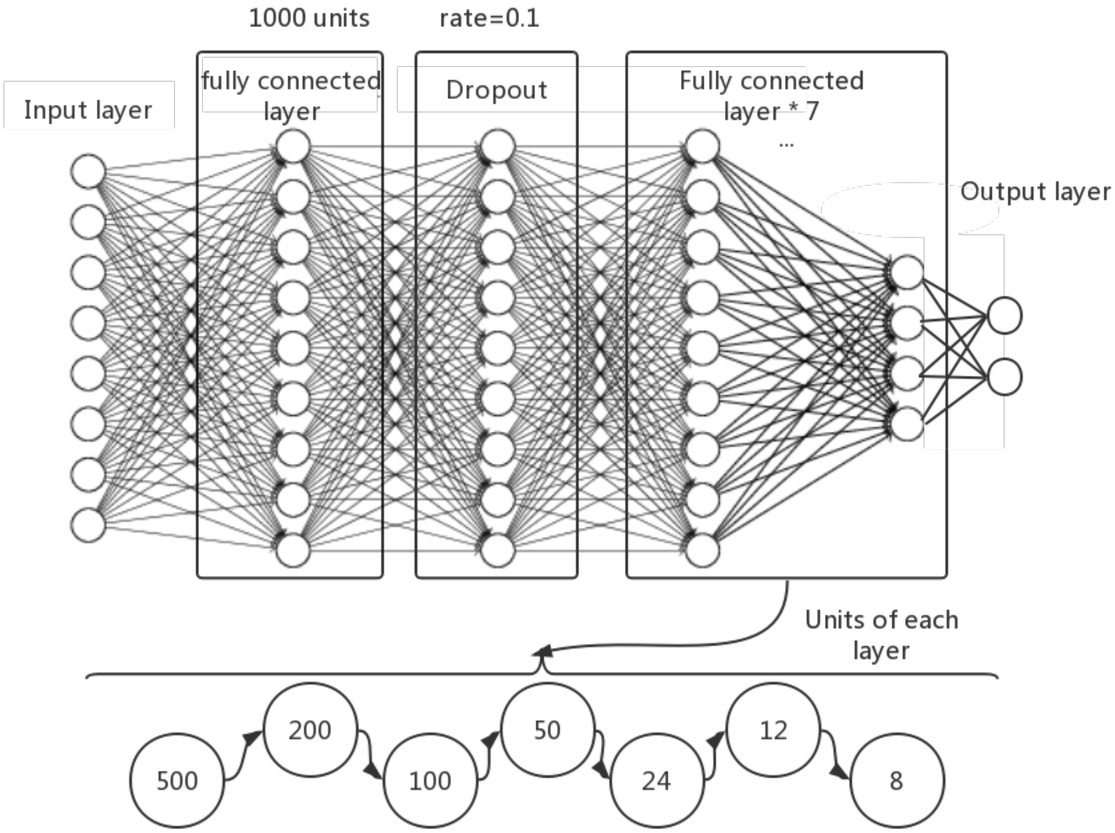


Fig. 7: Complex densely connected network with dropout layer.

TABLE I: Results of mentorship prediction on different dataset with different number of features and different networks.

Dataset	Features	Network	Train loss	Train accuracy	Test loss	Test accuracy
Ex	Ex 22	2*Dense	*	98%	*	95%
Optimized	Ex 22	2*Dense	0.1209	96.01%	0.11471	96.247%
Optimized	Newly 63	2*Dense	0.1018	96.49%	0.10262	96.48%
Optimized	Ex 22	Dense-Dropout-Dense-Dense	0.0953	0.9669%	0.0872	97.077%
Optimized	Newly 63	Dense-Dropout-Dense-Dense	0.0809	97.23%	0.07723	97.38%
Optimized	Ex22	Complex densely connected network	0.0521	98.35%	0.05644	98.29%
Optimized	Newly 63	Complex densely connected network	0.0057	99.81%	0.07877	99.08%

- Train a mentorship prediction model with high accuracy(98% compared to the previous 95%)

At last, I'd like to thank professor Fu and doctor Jia for their guide. Special thanks for Wenzheng Tao and Ziyu Wang. They help us a lot. Thanks for my partner Kaibin Zheng. She is an excellent girl.