Supervised Machine Learning for Academic Collaboration Prediction

Major Qualifying Project

Advisor:

Wilson Wong

Written By:

Thomas Koker

WPI

A Major Qualifying Project
Worcester Polytechnic Institute

Submitted to the Faculty of the Worcester Polytechnic Institute
in partial fulfillment of the requirements for the Degree of
Bachelor of Science in Computer Science.

January 2019 - October 2019

Contents

List of Tables

List of Figures

Abstract

Machine Learning has long been used for classification tasks. We show how applying a link prediction algorithm to a social network formed by co-authorship can be used to predict future collaboration in the Information Systems (IS) education community. We performed link prediction using Logistic Regression, Support Vector Machine, and Random Forest models.

1 Introduction

While a group of researchers will often be able to produce better work than that of an individual researcher, finding suitable collaborators can be challenging. Though there may be a large number of researchers in a given community, trying to predict which potential collaboration will be effective can be a difficult task. In communities such as Information Systems Education, collaboration often spans across different geographic locations, which can further increase the difficulty of the task.

By extracting publicly available co-authorship data from prominent organizations in Information Systems Education and applying various machine learning techniques, we can form predictions of future co-authorship. These predictions may then serve as recommendations for suitable authors to collaborate within the community.

2 Related Work

Logistic Regression and Support Vector Machines have been used to predict co-authorship among authors who are researching Coronary Artery Disease (Yu et al., 2014). In another study, a similar technique was performed on a collection of Computer Science papers published between 1990 and 2003, in which many models were used for link prediction on both an unweighted and weighted co-authorship network. It found that, out of many models, a decision tree classifier achieved the highest precision in predicting future co-authorship; however, it was inconclusive whether assigning weights to edges based on author contribution yielded improved results (Sa & Prudencio, 2010). On a study of link prediction on the co-authorship network formed from the Institute of Electronics Information and Communications Engineers (IEICE), a fitted SVM assigned low weights to the shortest path, Jaccard's coefficient, PageRank, and SimRank measures. This suggests that they may not be a good indicator of future co-authorship. Conversely, higher weights were assigned to the Katz and preferential attachment measures (Pavlov & Ichise, 2007).

3 Methodology

3.1 Supervised Learning for Link Prediction

A supervised learning model is built to map an input of similarity measures to probability of future collaboration. A data set of papers is split into three consecutive time frames, and, for each time frame, a network graph is formed. Each author is represented as a node, with edges connecting authors that have collaborated in that time frame. A training set is then created by selecting pairs of authors within the first time frame, computing the below similarity measures, and labeling whether or not the pair collaborate in the next time frame. A cross-validation set is created with the same method using the second and third time-frame. Finally, we fit a model to the training set and measure its accuracy on the cross-validation set.

3.2 Similarity Measures

While many similarity measures can be computed for a pair of nodes in a network graph, we selected the metrics listed in (Sa & Prudencio, 2010) with the addition of the Resource Allocation Algorithm, because it may be a better measure for link prediction (Zhou, Lü, & Zhang, 2009). These measures were computed with the help of NetworkX 2.2 (Hagberg, Swart, & S Chult, 2008).

3.2.1 Common Neighbors

Common neighbors are the number of coauthors that each author share. The number of common neighbors between authors u and v can be written as:

$$|N(u) \cap N(v)|$$

Where $N(u)$ denotes the set of all neighbors of node u.

3.2.2 Jaccard Coefficient

The Jaccard Coefficient, originally *coefficient de communauté* (Jaccard, 1908), measures the similarity between two sets by dividing the size of their intersection by the size of their union. We can compute the Jaccard Coefficient of the neighbors of authors u and v like so:

$$\frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|}$$

Two authors that share no common neighbors will have a coefficient of 0, while two authors that share the same set of neighbors will have a coefficient of 1.

3.2.3 Resource Allocation Algorithm

First introduced in (Zhou et al., 2009), the Resource Allocation Algorithm is defined as:

$$\sum_{w \in N(u) \cap N(v)} \frac{1}{|N(w)|}$$

If many of the common neighbors between u and v have a low number of neighbors themselves, any "resources" sent from u have a high likelihood of making their way to v and vice versa. In other words, the algorithm puts a weight on how important the two authors are to the authors that they share.

3.2.4 Preferential Attachment

Preferential attachment is the product of the size of each node's neighbor set:

$$|N(u)||N(v)|$$

Two authors that both have a large number of collaborators will have a large preferential attachment, which may indicate a higher probability of future collaboration (Pavlov & Ichise, 2007)

3.3 Learning Methods

With a dataset labeled with the above features, we built a classification model to predict the probability of future collaboration between two authors. Logistic Regression is the most widely used for classification problems. Support Vector Machines (SVM) may perform differently because it finds the optimal hyperplane separating the classes (Cortes & Vapnik, 1995). A decision tree has also shown to work well for link prediction (Sa & Prudencio, 2010), so we chose to also use a Random Forest model, an ensemble of decision trees.

3.3.1 Logistic Regression

Since author pairs are classified as either future coauthors or not we use a binomial model. Given label y, features x and parameters θ, we write out hypothesis function like so:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

The model can then be fit by minimizing cross-entropy loss function J:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)}\log(h_\theta(x^{(i)})) + (1-y^{(i)})\log(1-h_\theta(x^{(i)}))] + \frac{\lambda}{2m}\sum\nolimits_j \theta_j^2$$

$J(\theta)$ can be minimized via gradient descent. Parameters θ can then be viewed to obtain feature importance. A comparatively large positive parameter indicates that its corresponding feature is more important in predicting a positive label. Conversely, a small parameter indicates that its corresponding feature is less important in predicting future collaboration. λ is used for regularization to address any multicollinearity.

3.3.2 Support Vector Machines

SVM function in a similar way to Logistic Regression, however, the hypothesis function is written as:

$$h_\theta(x) = \begin{cases} 1 & \text{if } \theta^T x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

The model is then fit by minimizing the hinge loss function $J(\theta)$ with gradient descent:

$$J(\theta) = C\sum\nolimits_i \max(0, 1 - y^{(i)}\theta^T x^{(i)}) + \frac{1}{2}\sum\nolimits_j \theta_j^2$$

Unlike the cross-entropy loss function, the hinge loss function penalizes even a correct classification if its prediction does not exceed the classification by some margin. The result of minimizing the hinge loss is a hyperplane that separates the classes of data with an optimal margin (Cortes & Vapnik, 1995). Similar to λ in the cross-entropy loss function, C is used for regularization.

3.3.2.1 The Kernel Trick

In some cases, samples cannot be linearly separated. A solution to this is by applying kernels to the data, which maps the data into a higher-dimensional space. This transformation allows the data to be separated with a linear hyperplane, improving the classification ability of the model. For our model, we will be using the Radial Basis Function kernel, which maps data into an infinite-dimensional space, and is commonly used with SVMs (Chang, Hsieh, Chang, Ringgaard, & Lin, 2010).

3.3.3 Random Forest

Sa et al. showed that decision trees can be used for link prediction with higher accuracy than other models (Sa & Prudencio, 2010). As a single decision tree can easily overfit training data, we use a Random Forest model instead. Each tree is formed with a bootstrapped sample of the training set, so each tree is fit to a different subset of the data. By building many "de-correlated" trees, and then averaging the predictions, models generally perform with higher accuracy and lower variance (Hastie, Tibshirani, Friedman, & Franklin, 2005).

With a fitted random forest, feature importance can be obtained using mean decrease impurity; that is how much each feature's splits increase the purity of the classification. We use the Random Forest classifier implemented in scikit-learn 0.20.3 (Pedregosa et al., 2011), which uses the Gini Index as a measure of purity.

4 Data

To obtain a dataset of authors in the IS education communities, we collected papers and authors from the following journals and conferences: Information Systems Education Conference (ISECON), EDSIG Conference on Information Systems and Computing Education (EDSIGCON), Journal of Information Systems Education (JISE), and Information Systems Education Journal (ISEDJ).

4.1 Collection

For each organization, we developed a web scraping tool to extract author names, paper names, and year from the organizations' website. Since each organization has a differently structured website (some changing in structure over different years), special care was taken to ensure that the data was extracted correctly. It is worth noting, however, that some authors were listed with different spellings of their names in different organizations, resulting in multiple authors being extracted instead of one. The following URLs were used as a starting point for the web scraper:

Table 1: Data sources for each organization.

Organization	URL
ISECON 2000-2009	http://proc.edsig.org/xref/title.html
ISECON 2010-2014	http://proc.edsig.org/{year}/index.html
ISECON 2015-2018	http://proceedings.isecon.org/dt-row-data
EDSIGCON 2015-2018	http://proc.iscap.info/{year}/
JISE	http://jise.org/archives.html
ISEDJ	http://isedj.org/archives.html

In the case of ISECON from 2000 to 2014 and EDSIGCON 2015-2018, paper titles were extracted from the "Titles" page of the respective year, and then corresponding authors were extracted from the "Authors" page. ISECON paper metadata from 2015 to 2018 was downloaded as a single JSON document from the above URL. JISE and ISEDJ required programmatically navigating through each journals' volumes, and then each paper within the volumes to extract the relevant information. After all of the titles, authors, and years of each organization's papers were extracted, the data was stored in an SQLite database. A total of 3522 papers from years 2000 to 2018 along with 3076 unique authors were extracted. Although some papers from conferences and their corresponding journals may contain duplicates, we did not assign any weight to the number of times two coauthors collaborated, so the resulting network graph was not affected.

4.2 Visualization

Once the data was collected, we began to analyze the social network graph formed by co-authorship. To promote a better understanding of this social network, we built a web application that allowed querying of the dataset and interactive exploration of the network graph, which can be seen in Figure 1.

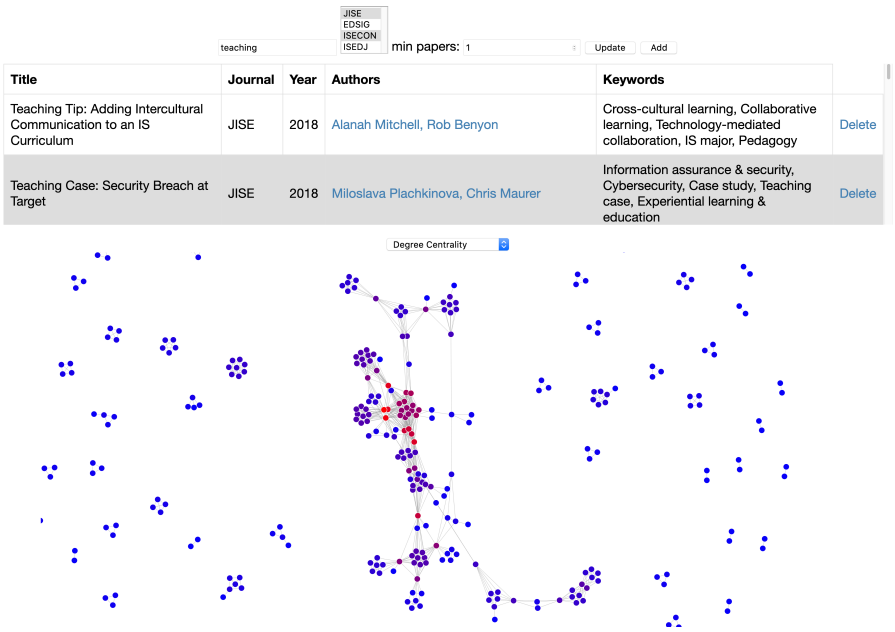

Figure 1: Screenshot of application.

4.3 Preprocessing

In order to fit our collaboration prediction models, our dataset was split into three time frames of equal length: $T_1 = [2000 - 2005]$, $T_2 = [2006 - 2011]$, and $T_3 = [2012 - 2017]$. A training set was formed consisting of author pairs who were active in both T_1 and T_2, *did not* collaborate in T_1, and *did* collaborate in T_2. These pairs were labeled **positive**. Additionly, an equal number of author pairs who were active in both T_1 and T_2, but did *not* collaborate in either T_1 or T_2 were randomly sampled and labelled **negative**. The same procedure was followed to create a cross-validation set over T_2 and T_3. The resulting training set consisted of 370 samples, while the cross-validation set contained 106 pairs.

The above-listed similarity measures were then computed for each pair over the network graphs formed in T_1 and T_2 for the training set and validation set, respectively. It is recommended to standardize input features before training (Hastie et al., 2005), which was done by subtracting the mean, then dividing by the standard deviation for each feature.

5 Results

The training set was then fit to Logistic Regression, Support Vector Machine, and Random Forest models, with accuracy measured on the cross-validation set. We also measured each model's receiver operator characteristic (ROC) curve, which shows the relationship of true positives vs. false positives as the decision threshold is varied. The area under this curve (AUC) is then used to compare the different models' classification ability.

Table 2: Accuracy and AUC for each learning method.

Model	Accuracy	AUC
LR	0.806	0.870
SVM	**0.827**	**0.871**
RF	**0.827**	0.844

We found the SVM and RF models to have the highest accuracy, being able to predict future collaboration with an 82.7% accuracy (compared to 50% randomly guessing). To see which similarity measures are more indicative of future collaboration, we plotted the feature importance of both the LR and RF models, seen in Figure 3.

6 Discussion

All three models were able to predict future co-authorship quite well, using only topological features from the co-authorship network. As has been shown

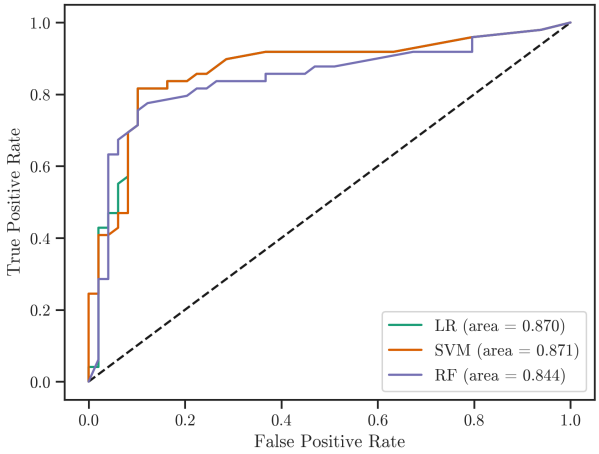

Figure 2: ROC curves for each model.

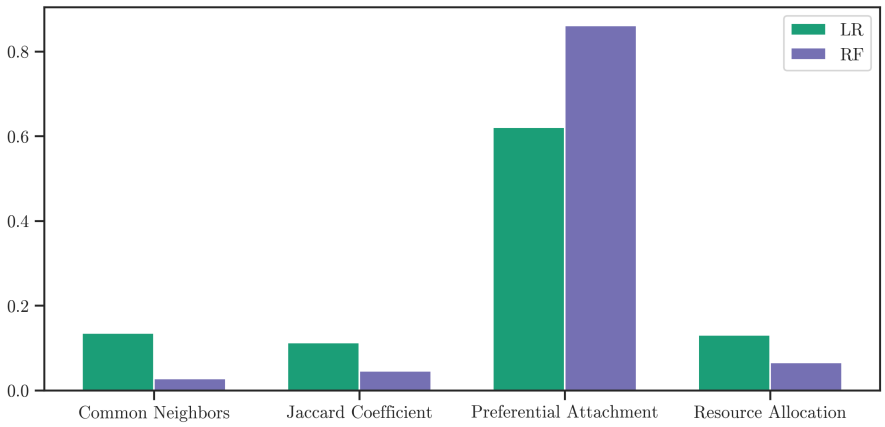

Figure 3: Feature importance for LR and RF models.

in past literature (Pavlov & Ichise, 2007), preferential attachment is a strong indicator of future co-authorship. That is, in a relatively small community such as that of Information System Education, the greater the number of coauthors two individuals have, the more likely they are to work together in the future.

7 Conclusion

Such a model may be used as a recommendation system to suggest future coauthors for an individual by listing the authors in which he or she has the highest probability of working with. Results may be improved by adding more features such as commonality of keywords or language used in papers.

References

Chang, Y.-W., Hsieh, C.-J., Chang, K.-W., Ringgaard, M., & Lin, C.-J. (2010). Training and testing low-degree polynomial data mappings via linear svm. *Journal of Machine Learning Research*, *11*(Apr), 1471–1490.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297.

Hagberg, A., Swart, P., & S Chult, D. (2008). *Exploring network structure, dynamics, and function using networkx*. Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Hastie, T., Tibshirani, R., Friedman, J., & Franklin, J. (2005). The elements of statistical learning: Data mining, inference and prediction. *The Mathematical Intelligencer*, *27*(2), 83–85.

Jaccard, P. (1908). Nouvelles recherches sur la distribution florale. *Bull. Soc. Vaud. Sci. Nat.*, *44*, 223–270.

Pavlov, M., & Ichise, R. (2007). Finding experts by link prediction in co-authorship networks. *FEWS*, *290*, 42–55.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Sa, H. R., & Prudencio, R. B. (2010). Supervised learning for link prediction in weighted networks. *III international workshop on web and text intelligence*.

Yu, Q., Long, C., Lv, Y., Shao, H., He, P., & Duan, Z. (2014). Predicting co-author relationship in medical co-authorship networks. *PloS One*, *9*(7), e101214.

Zhou, T., Lü, L., & Zhang, Y.-C. (2009). Predicting missing links via local information. *The European Physical Journal B*, *71*(4), 623–630.