Predicting Links in Social Network

Krishna Das and Smriti. K. Sinha
Department of Computer Science & Engineering
Tezpur University

Abstract

Link prediction in social networks has attracted increasing attention from both mathematics and computer science communities. Various algorithms can be used to extract missing information, identify spurious interactions, network evolving mechanisms, expansion of asocial network communities and so on. Given a snapshot of a social network, can we infer which new interactions among its members are likely to occur in the near future? This study summarizes recent progress about link prediction algorithms, some applications, experimental results and outlines of future challenges of link prediction research problem.

***Keywords*:** Social network, Link Prediction, Complex Networks, Similarity measure, Supervised learning.

Introduction

Social networks such as Facebook, Twitter etc. have spurred lots of research in link prediction and recommendation, which aim at predicting unobserved or missing connections based on existing structure in a network. Relationships in a network are represented by a set of nodes and edges, in which nodes are actors and edges are interactions between those actors. In a real world setting, edge information is missing due to reasons such as incomplete data collection process or uncertainty of relationships or resource limitation. Besides, to predict future connections in a dynamic network is also a hot topic. Social networking websites would like to customize new friend suggestions for users; intelligence agencies can prevent and predict criminal activities by monitoring potential relationships in malicious networks; financial organizations would like to detect fraudulent activities by inspecting transactional networks. Therefore, establishing a robust machine learning model to effectively predict potential links are worthwhile.

In this report, we reviewed a collection of state-of-the-art approaches in link prediction area, inspired by which, we conducted experiments using different methods. In a nut shell, we have a train data file formatted as adjacent list and a test data formatted as source- destination pairs. For each pair in the test, we need to determine whether this edge is real (missing edge) in the train graph or not. Intuitively, the training network is a sub-graph obtained from the whole Twitter data. Basically, we employed both supervised and unsupervised methods to our learning models and surprisingly discovered that the unsupervised method outperformed supervised methods, which is further discussed in following sections.

Related Work

The link prediction problems are challenging by its sparse nature. Research has typically tackled this by using unsupervised approaches, and most of

which either generate score based on node neighborhoods or path information. Liben-Nowell and Kleinberg [1] thoroughly evaluated a range of unsupervised methods and concluded that the Adamic-Adar measure of node similarity performed best. They also found the baseline common neighbors predictor worked surprisingly well compared to Jaccard's coefficient, SimRank [2] and random walk based hitting time. In [3], the author utilized a modified random walk approach--Personalized PageRank [4] to calculate rank values for each node.

The authors in [5] carefully explained some of the properties of imbalance |7| in spare networks with its relationship to graph distance, and how to overcome this by supervised learning. They finally achieved desirable results by extracting such as in-degree, out-degree and some unsupervised measures such as number of common neighbours, Adamic-Adar, shortest path as features to train some ensemble classifiers like Random Forests. Likewise, similarity scores were extracted as features for supervised classification in the studies of |6|, which enriched the feature set in to a more comprehensive one with 39 different features (e.g. Cosine similarity, Bayesian Sets, EdgeRank etc).

1. Methodology

1.1 Overview

In our experiment, the training graph G is expressed as an adjacency matrix A consisting of 4,867,136 users joined by 20,000,000 edges. To predict whether a specific testing edge is real or fake, we pre-processed the training dataset by uniformly sampling 10,000 positive edges and 10,000 negative pairs based upon matrix A, which was chiefly due to the limitations of memory and processor. Then, each of these training edges would be transformed to a vector of features with a label of *real* or *fake*. Thus, link prediction problem could be solved by employing unsupervised and supervised learning models with effective features.

1.2 Training Data Preprocessing

Followers matrix generating -- The training data given for experiments is a tab-delimited adjacency matrix, where each row represents a user and its outbound neighbours (followers). Hence, we pre-generated a follower's matrix B from original graph, for the sake of cheaply obtaining followers statistics.

Celebrities data removal - In our training data, the average number of followers for each user is 93, and hence nodes having more than 100 followers have been discarded for the sake of obtaining a more accurate classifier. Similarly positive edges sampling and negative edges generating and sampling has been done in training data set to make ease of computational analysis.

1.3 Feature Set Extraction

In this report, we have typically explored the use of three feature sets: proximity, ego-centric and aggregation [8]. We have considered mainly three features in this analysis. **Proximity** features are characteristics that represent some form of proximity between the pair of nodes [2]. **Ego-centric features** are those features concentrating on the local network of u or v. An example would be the number of followers of u. **Aggregation function** is applied to generate effective features for link prediction, which are sum of followees, sum of followers sum of friends, sum of neighbours, respectively.

1.4 Edge Classification

A vast number of classification algorithms can be chosen for link predictions. In this report, we

performed some experiments on unsupervised learning methods by employing above mentioned similarity features [7] as well as supervised learning algorithms including KNN, Random Forests, Random Forests and SVM. For implementation of the algorithms, an external machine learning package called WEKA has been be employed.

1.4.1 Unsupervised Learning

We would start classification with unsupervised learning [11] since it is more straightforward than supervised learning. To ensure that unsupervised learning would produce desirable results, we need to apply some features that in a way can reflect the overall statistical pattern of the testing data. Specifically, the similarity score (Cosine, Jaccard and Adamic-Adar) of u and v would be computed and then scaled to [0,1] as the confidence score of whether the link between them is real.

1.4.2 Supervised Learning

In supervised learning, we performed experiments on four algorithms, which are KNN, Random Forests, Random Forests and non-linear SVM, respectively. KNN was chosen as our classifier candidate because KNN is a very simple classifier that works well on basic recognition problems. Additionally, KNN is robust to noisy training data and effective when the training data is large. By applying Bagging which is one of the ensemble techniques, we expect that the performance would be significantly better than the base classifier. As for non-linear SVM [12], even though the training time is relatively long [10], it is capable of capturing complex relationships between nodes in the context of link prediction.

Experimental Result and Analysis

In both training and testing dataset, counts of real edges and fake edges were almost the same, which means a baseline classifier would have an accuracy 50% by predicting all testing edges are 1 or 0.

Table 1: Unsupervised Classification results

Similarity Measure	Accuracy Level	Average Accuracy
Cosine Similarity	81.00 %	80.00 %
Jaccard Coefficient	79.00 %	
Adamic-Adar similarity	80.45 %	

Table 1 depicts the performance results of unsupervised learning on similarity features. It can be seen that all the similarity features we attempted achieved accuracy above 80%, which indicates that these similarity features have a good capability of discriminating real and fake links. Table 2 shows the performance comparison for different supervised classifiers on training and testing dataset.

Table 2: Supervised Classification results

Name of Classifier	Accuracy Level	Average Accuracy
KNN	73.00 %	74.43 %
Random Forests	74.00 %	
SVM	76.30 %	

The performance results in Table 1 were produced by 10-fold cross-validation on training data and Table 2 was constructed based on the AUC scores over testing data. As we can see from these two tables, non-linear SVM performed the best for both datasets with an accuracy of 76.3% and

75.5%, respectively. Moreover, all classifiers achieved a better performance on training data than testing data, which reveals that even though cross-validation is applied to mitigate the risk of over fitting, these classifiers still over-fit training data to some extent. To compare the results of Table 1 and Table 2, it can be surprisingly found that unsupervised learning methods perform significantly better than supervised learning models. Additionally, it can be inferred that wrong classifications are most likely contributed by the fractions sitting under the critical overlap regions for most features [8].

Conclusion

Link prediction in large graph is still very challenging and attracting area for research in social network. A lot of research focus has been given in this area. We found that in unsupervised domain, Cosine Similarity performed best followed by Adamic-Adar and then Jaccard's coefficient. In supervised domain, we extracted 14 features from the network structure and applied classifiers on them. To our surprise, Random Forests did not achieve the best performance but *None-Linear SVM* out performed it instead. The reasons were discussed in this report, and we believe our sampling and feature extraction need to be improved for a better result in terms of supervised learning.

There were many promising approaches which were not implemented and tested due to hardware or time constraints; we would like to explore more possibility in link prediction area in the future. This analysis suggests that users with similar topical interests are more likely to be friends, and therefore semantic similarity measures among users based solely on their annotation metadata should be predictive of social links. This research work is continuing to test more number of algorithms both in supervised & unsupervised domains to find out more accurate results.

References

1. Liben-Nowell, D., & Kleinberg, Jon., (2003). The link prediction problem for social networks. In Proceedings of the twelfth international conference on Information and knowledge management, New York, NY, USA, CIKM '03, ACM, pp. 556-559.

2. Jeh, G., & Widom, J., (2002). Sim Rank: A measure of structural-context similarity. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

3. Chen, E., Edge Prediction in a Social Graph: My Solution to Facebook's User Recommendation Contest on Kaggle. Retrieved September 15,2013 from http://blog.echen.me/2012/07/31/edgeprediction-in-a-social-graph-my-solution-to-facebooks-user-recommendation-contest-on-kaggle/

4. Page, L., Brin, S., Motwani, R., & Winograd, T., (1999). The Page Rank citation ranking: Bringing order to the Web. Technical report, Stanford University.

5. Lichten Walter, R. N., Lussier, J. T., & Chawla, N. V., (2010). New perspectives and methods in link prediction. In Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 243-252.

6. Cukierski, W., Hamner, B., & Yang, B., (2011). Graph-based features for supervised link prediction. In Neural Networks (IJCNN), The 2011 International Joint Conference, IEEE. pp. 1237-1244.

7. Backstrom, L., & Leskovec, J. (2011). Supervised random walks: predicting and recommending links in social networks. In Proceedings of the fourth ACM international conference on Web search and data mining, ACM, pp. 635-644.

8. Hasan, M. A., Chaoji, V., Salem, S., & Zaki, M., (2006). Link prediction using supervised learning. In SDM' 6: Workshop on Link Analysis Counter-terrorism and Security.

9. Li, P., Liu, H., Yu, J. X., He, J., & Du, X., (2010). Fast Single-Pair Sim Rank Computation. In SDM, pp. 571-582.

10. Latha, R. H., & Kumari, K. S., Survey On Link Prediction In Facebook And Twitter.

11. Rowe, M., Stankovic, M., & Alani, H., (2012). Who will follow whom? exploiting semantics for link prediction in attention-information networks. In The Semantic Web-ISWC Springer Berlin Heidelberg, pp. 476-491.

12. Gupta, P., Goel, A., Lin, J., Sharma, A., Wang, D., and Zadeh, R., (2013). The who to follow service at twitter. In Proceedings of the 22nd international conference on World Wide Web International World Wide Web Conferences Steering Committee, pp. 505-514.