Link Prediction in Pairwise Relationships

Statistical Machine Learning

Abhishek Kumar Gupta (869632) *Bahriddin Abdiev (824110)* *Karan Razdan (883454)*

---

1. Introduction

Link prediction is one of the fundamental problems in social network analysis. The social network can be visualized as a graph, where nodes correspond to the entities and the edges (or links) represents relationships/interactions between entities. Since these networks are highly sparse in nature and the data completeness can't be ensured, forecasting a new connection based on the existing information is a task of great importance. In this project, the student groups are provided with social network data from Twitter, which consists of 20,000 crawled nodes (approximately 4.6 million in total) that generated around 24 million edges. This project then requires predicting the probability of the directed links between the 2000 test nodes $node_a$ to $node_b$. In this paper, $node_a$will be the source and $node_b$ will be the sink.

2. Related Work

As this project involves social interactions and relationships between people, the presence of common friends or nodes plays an important role. While looking for relevant literature, we found that this competition is very similar to IJCNN Social Network Challenge which was based on data from Flickr. Furthermore, some key differences were found between the two datasets: number of edges where 3 times lesser than ours, number of nodes with outbound edges was 2 times more and number of nodes with inbound edges was 4 times lesser. This shows crawled users (nodes with outbound edges) in Twitter have more connections on average than Flickr. Considering the fact Twitter has about 3 times more monthly active users than Flickr and the other competition was 7 years ago, it is not a surprising fact. Two key pieces of literature were written by the top teams of IJCNN competition [1][3]. The team who took first place used a deanonymization technique which was deemed impractical for our case due to several factors. The second placed team talked about several features which we explored as well.

3. Approach

We tried two main approaches to tackling the problem. These are:

a. Similarity-Based Approach: Similarity-based approaches are very popular in link prediction use cases as they are easy to build, relatively quick and offer good results. The higher the similarity measure, the higher the probability of two nodes forming a link in the future. We considered the similarity using the 3 approaches [1]:

1. How similar is $node_a$ to $node_b$?

2. How similar are the outbound neighbours of $node_a$ to $node_b$?
3. How similar are the inbound neighbours of $node_b$ to $node_a$?

From our experiments, approach 3 gave us the best results and had the highest scores in general. It can be reasoned that the number of outbound neighbours of $node_a$ is relatively larger than $node_b$'s inbound neighbours and each user who follows $node_b$ has higher weight when it comes to calculating the similarity. The graph was converted to an undirected version to make sure all the similarity measures work with them. Another challenge was to decide, in case of the second and third approaches, when we had to submit final predictions, it was found that taking averages for all non-zero values yielded the best results.

b. Learning-Based Approach: In this approach, we treat the problem as a binary classification task, where we use typical machine learning models i.e. classifiers to predict the probabilities. This includes first extracting the features from the training and test data, then building a classifier and making the prediction using that and lastly, evaluating the classification accuracy.

Table 1 AUC scores for different similarities

Method	**Best AUC**	**Method**	**Best AUC**
Common Neighbours	0.63092	The Hub Promoted Index (approach 3)	0.75337
Jaccard (approach 3)	0.88667	The Hub Depressed Index (approach 3)	0.88688
Adamic – Adar	0.815792	Preferential Attachment (approach 3)	0.86457
The Salton Index (approach 3)	0.88326	Resource Allocation	0.84973
The Sørensen Index (approach 3)	0.88756	Cosine Similarity (approach 3)	0.88326

4. Results and Analysis

We realised soon that most network-based packages (NetworkX, iGraph and Snap) did not work well with the size of the data and limited power of our machines and hence we created our own framework to work with local subgraphs in most cases. Our initial approach was to begin by creating some negative samples and running supervised learning algorithms. This assumed that any edge not occurring in the training set could be counted as a negative sample. Moreover, we focussed on looking at the distribution in the data and how nodes were associated with each other. It was found that there was a positive correlation between the number of neighbours of the source and sink. It was also found that in the training data, 60% of the sink nodes had only one inbound edge with 40% being more than one. This split was 92-8% in the case of the test data, which was considered while sampling data while running our test so that we could match our own distribution more like the testing distribution. Simple features like in and out degrees were used

as a baseline gave us an AUC of 0.761. Soon, it was realised that creating negative edges might mean that we were making assumptions about the data and we might be assuming a negative edge where it a) might not have been crawled b) might have existed in a later point in time. For this reason, we tried to explore the option of a single class SVM, which is more commonly used in the cases of anomaly detection but it could not offer promising results with AUC of 0.7362.

With creating negative samples not seeming like the right approach, we tried to look at various similarity measures to see how they perform. Jaccard similarity tries to explore, out of the combined neighbourhood of $node_a$ and $node_b$, how many of them are common. This becomes a strong measure in case of real world friendships since people with more common connections are definitely more likely to be connected. Hub Promoted Index tries to give larger weights to nodes with higher degree whereas the Hub Depressed Index does the opposite. The later works better for social networks where our hypothesis was that larger degree nodes can signify a celebrity that most people follow whereas, lower degree nodes can represent actual friendships that form stronger connections. The key is negatively rewarding the overall number of neighbours, since more neighbours a node has, the less significant their number of common neighbours become. The Adamic/Adar index is based on the inverse log degree of the shared neighbours of $node_a$ and $node_b$, giving higher preferences to a larger shared neighbourhood. All these measures try to give value to shared neighbourhoods and hence were chosen as good candidates to model friendships. Other similarities mentioned in Table 1 were used as well.

Similarity measures ended up scoring much better than our classifier. With this in mind, we continued creating more powerful similarity measures and comparing the scores. With this approach, the two most powerful measures we got were Jaccard Coefficient (approach 3) and Hub Depressed Index (approach 3), both giving an AUC 0.886. With none of the features improving, we tried to take a weighted average of the various similarities according to their score (mimicking machine learning!) to try and get the most optimal result. Although this ran a risk of overfitting to the public leaderboard, we felt it was worth taking the chance. The final submission was a weighted average of the local similarities and gave an AUC of 0.89538.

5. Future improvements and Conclusions

Due to several constraints, several approaches could not be tried out. Most notably, we ended up with 23 features but we had planned on many more like EdgeRank, Katz, SimRank and BayesianSets. We would have also liked to generate negative samples more intelligently and play around with more models to get better accuracy. All in all, link prediction in social media is still a challenging problem due to the incompleteness and dynamic nature of the data. Though similarity measures delivered impressive results in our case, it also can't be denied that a machine learning model can deliver better results given enough features and data for the correct classifier.

6. References

[1] William Cukierski, Benjamin Hamner, Bo Yang, 2011, Graph-based Features for Supervised Link Prediction

[2] WANG Peng, XU BaoWen, WU YuRong, ZHOU XiaoYu, 2015, Link Prediction in Social Networks: the State-of-the-Art

[3] Arvind Narayanan, Elaine Shi, Benjamin I. P. Rubinstein 2011, Link Prediction by De-anonymization: How We Won the Kaggle Social Network Challenge