Link Prediction

Andres Medina (828936), Santanu Nanda (975771) , Graham Barrington (197921017)
School of Computing and Information Systems, University of Melbourne

HIGHLIGHTS

- Link Prediction – In Class Kaggle Competition AUC score of most interesting submission: 76.74
- Pre-processing – Data reduction from 20,000 source nodes and 4,867,136 sink nodes, to a total training set of 714,603 nodes, producing 11,828,805 potential tuples
- Concatenate 20K real link tuples with a random 20K fake link tuple list to create a full balanced training dataset of positive (present) and negative (absent) links. The fake tuples were constrained to those that might potentially "triangulate".
- Feature engineering - we consider a large list of similarity features described in the literature with topographical node and edge characterisations, including a new measure – GB_index - to capture outlier behaviour
- Model selection – Consideration of Logistic Regression, SVC, Decision Tree Regression, Random Forest Regression, Neural Networks and KNeighbor Regression, settling finally on KNeighbor Regression with the number of neighbours reaching an elbow point at 30 with the full feature set.

ARTICLE INFO:

Article history:

Submitted -7 September 2018

Keywords:

Machine Learning
Link Prediction

ABSTRACT

We describe the evolution of an approach to Link Prediction deployed as part of a Kaggle competition: "In-Class Link Prediction". Given a training network - a list of pairwise relationships – representing a partial crawl of the Twitter Social network, we construct a sub-graph made of nodes in the network – Twitter Users – and directed edges between them representing that node A follows node B. From this graph and based on topological features alone, we produce a supervised machine learning algorithm that ascribes a probability to any tuple, thereby quantifying its potential as a "missing" link.

1. INTRODUCTION

A key social network analysis problem involves the prediction of links between individuals in networks containing hundreds of millions of users. Links may be missing due to imperfect acquirement, a disconnect between real and on-line friendship connections or deliberate subterfuge. Prediction of the existence of hidden links or of creating new links in social networks is commonly referred to as the **Link Prediction** problem and spans research as varied as recommending friends, identifying terrorists and predicting the spread of communicable diseases or new ideas.

2. APPROACH TAKEN TO LINK PREDICTION

The authors collaborated through the tools of GitHub and Sourcetree and utilised the numpy, pandas, networkx, and igraph libraries and the suite of machine learning models provided through sklearn to develop a supervised machine-learning algorithm to address the Link Prediction problem.

The algorithm developed involved the following decision-points:

i. **Initial data analysis**: Network scale and the distribution of in-bound (followed by) and outbound (following) links was determined across the network. The need for data reduction due to computational constraints was determined. Two characteristics were highlighted: the importance of the directed nature of the graph and the extreme imbalance in the dataset (the number of edges known to be present being orders of magnitude less than those known to be absent).

ii. **Pre-processing**: The number of nodes considered and the number of potential tuples assessed was filtered by a minimum neighbour threshold (nodes that followed less than a default (5) nodes were eliminated from consideration as a sink). Nodes that followed more than a max_neighbour threshold (default 1000) had its subgraph constrained to only the top max_neighbour threshold (1000) child nodes, using the number of followers of those child nodes as the rank metric. By this means data-reduction strategies were explored settling on a min_neighbour threshold of 5 and max_neighbour threshold of 1000.

iii. **Training set:** The need to address the imbalance problem gave rise to a strategy of creating 20,000 Fake-tuples (fabricated absent links) to pair to the 20,000 known positive links given in a training set representing a partial crawl of the Twitter Social network. A flag to choose either “random” or “triangulisable” fake links was set to explore the value of each of these strategies in choosing the fake link set. Potentially triangulating fakes were determined as preferred on theoretical grounds. A triangulating link is one where know A->B and B->C and we ask whether ?A->C. They stand in contrast to knowing A-> B and C->D and asking whether C->D, which is not triangulating.

iv. **Feature Engineering:** A range of features (20) were calculated to capture similarity measures of nodes and edges and so characterise the topological character of a graph relevant to predicting missing links. Most of the features used are detailed in Fire et al (2011) - ('opp_dir_flag', 'common_friends_in' , 'common_friends_out', 'total_friends', 'friends measure', 'preferential attachment_score', 'transitive_friends', 'shortest_path', 'in-degree source', 'in-degree sink', 'out-degree source', 'out-degree sink', 'bi-degree source', 'bi-degree sink', 'Jaccard similarity in', 'Jaccard similarity out', 'Jaccard similarity all', 'Adamic Adar score', 'Resource Allocation score'). We propose a further feature of an edge - 'GB index' which classifies a node according to it being an outlier in regards to the number of nodes it either follows or is followed by, either below a lower-bound (default 5^{th}) percentile or above an upper-bound (default 95^{th}) percentile of its in-bound or out-bound degree. Nodes are thereby classified into categories of “Irrelevant” (low inbound and outbound edge numbers), “Broadcaster” (low inbound and high out-bound edge numbers), “Distillller” (high inbound and low out-bound edge numbers), or “Networker” (high inbound and out-bound edge numbers) to capture information processing characteristics. The measure is formed by the product of the out-bound and -in-bound percentile ranks for the source and sink nodes respectively, or the square-root of this product for non-outlier nodes.

v. **Feature Selection:** it was recognised that feature engineering was an important part of the development of the algorithm, but that managing complexity was also important. In this setting feature ***selection*** was equally important. The feature set of 20 did not appear to produce the desired performance in terms of AUC and so a method of reducing this number was explored (reduced to 6: 'friends measure', 'pa score', 'shortest path', 'Jaccard similarity out', 'Adamic Adar score', 'Resource Allocation score' , 'GB index'). The criteria for this reduction was: minimal consequent change to AUC, the shape of the test set prediction curve and guidance from a correlation matrix identifying redundancy. This allowed computation of a larger training set, but without improvements in AUC.

vi. **Train/Holdout set:** a hold-out set was defined to allow for cross-validation

vii. **Model Selection** (Consideration of Logistic Regression, SVC, Decision Tree Regression, Random Forest Regression, Neural Networks and KNeighbor Regression, settling finally on the last. The appeal of this model stems from the focus on features that model similarity as a local phenomenon and the importance of the k closest training examples in the feature space. We had expected

Random Forest and other Ensemble methods to be attractive, but empirically this did not prove to be the case.

viii. **Model tuning –** a process of Grid- search and optimising the number of neighbours reached the elbow point at 30.

3. RESULTS

Features						AUC	K	
All 18 features	SP	JAC-OUT	AA			0.7674	30	***Legend:***
Friends measure	SP	JAC-OUT	AA	RA	GB	0.623	30	
	SP	JAC-OUT	AA	RA	GB	0.727	30	AA : Adamic Adar
	SP	JAC-OUT	AA	RA		0.722	30	SP : shortest path
	SP	JAC-OUT	AA			0.721	30	JAC-jaccard_out
	SP		AA			0.715	30	GB – GB_index
	SP	JAC-OUT	AA	RA	GB	0.731	50	RA: Resource A
	SP	JAC-OUT	AA	RA	GB	0.731	100	
	SP	JAC-OUT	AA	RA	GB	0.737	200	

4. PERFORMANCE, DISCUSSION, ALTERNATIVE APPROACHES – LESSONS LEARNED

Hypotheses that we explored for why our performance on the AUC Kaggle measure was not higher:

i. Too many features, inadequately managing complexity and computational trade-offs – yet we dropped all features other than 4 (Jaccard similarity out, Adamic Adar, Resource Allocation score, shortest path) with the AUC dropping only from 76 to 73.

ii. Information lost in pre-processing, data-reduction decisions – computational constraints required that we filter some nodes and potential links (tuples) from the graph considered

iii. Insufficient sampling – inadequate training set size – we did train over 25000 rows which took up to 9 hours of computation time

iv. Standardisation – we did standardise features, important in using KNeighbour Regression

v. Assumptions imposed through the creation of our fake tuple set – the use of the triangulising Fake-tuple set as the negative link examples in the training set may have mis-directed our search

vi. Insufficient “tuning” gains -the computational load of our feature calculations impeded our exploration of models and the tuning phase that may have provided further performance improvements

5. REFERENCES

Fire, M. & Tenenboim, L. et. al., "Link Prediction in Social Networks Using Computationally Efficient Topological Features," *2011 IEEE Third International Conference on Privacy, Security, Risk*, 73-80. (2011)

Pan, L. & Zhou, T. et. al. “Predicting missing links and identifying spurious links via likelihood analysis”, Nature: Scientific Reports volume6, Article number: 22955 (2016)

Garcia-Gasulla, D. “Link Prediction in Large Directed Graphs”, Ph. D. thesis (2015)

Martinčić-Ipsić, S & Močibob, E., Link Prediction on Twitter, PLOS , ONEhttps://doi.org/10.1371/journal.pone.0181079 July 18, 2017

Hasan, M & Chaoji, V et. al., Link Prediction Using Supervised Learning (2006), https://www.researchgate.net/publication/277289291

Garcia-Gasulla, D. & Cortes, U., Link Prediction in Very Large Directed Graphs: Exploiting Hierarchical Properties in Parallel, (2014), https://www.researchgate.net/publication/261587508