Link Prediction In Social Network Analysis

Jihoon Kim
UCSD Division of Biomedical Informatics
j5kim@ucsd.edu

Hari Damineni
UCSD Computer Science & Engineering
hdaminen@cs.ucsd.edu

André Christoffer Andersen
NTNU Industrial Economics
andrecan@stud.ntnu.no

May 31, 2011

Abstract

Social network can be represented by a graph where nodes are the people and the edges are the relations that the nodes or the people share in between them. Relations can belong to different classes. For example, relations in between two people oculd friends, siblings, colleagues etc. In this paper, we build link prediction models using binomial logistic regression to predict the both existence and the link type amongst terrorists using the People In Terror (PIT) data.

1 Introduction

Links in social networks can be analyzed to predict the presence of other links using binary classification techniques. Similarly, predicting the different types of links (sibling, colleague, friend etc.) is in general a multiclass classification task, but for our assignment it will be binary. In this article, we build two separate models using logistic regression for the aforementioned classification tasks. For each model, the PIT dataset, which contains user's descriptive features and their link type, is used to construct a final dataset which is a weighted combination of individual, pair-wise and structural features derived from PIT dataset. The logistic regression models, built on these final datasets using 10-fold cross-validation, achieves an accuracy of 0.981 and and precision of 0.341 for the link prediction task while it achieves an accuracy of 0.776 and and precision of 0.867 on the colleagueship task. Henceforth, in this article, we will use the word agent to represent a terrorist for reasons of subtlety.

2 Data Description

Data used in this article is the PIT dataset from `http://cseweb.ucsd.edu/~elkan/291/PITdata.csv`. Dataset contains 851 links in between 244 agents. Each of the 851 rows contains 1227 features, which are divided into two identifiers (1 to 244) for the linked agents, 612 binary descriptive features of each linked agent in the order of their identifiers, and the type of linkage (belong to same organization or other relation). In the dataset, there are two classes of links present. One class relates the agents as belonging to the same organization or colleague and the other class links the agents as sharing some other relation (such as met before, worked with each other before, etc.). Among the 851 links, there are 461(54.17%) links which belong to the *colleague* class and and 390(45.83%) links which belong to the *other* class.

3 Data Preparation

A naive use of the existing dataset which contains only descriptive features of the agents can lead to models with severe drawbacks and flaws. For example, using a linear classification models on PIT dataset with only the concatenated descriptive features of two agents would yield a model where the link existence ranking for all other agents with a single agent is determined only by the features of other agent. Such a model does not detect and predict link existence based on the interaction patterns of the relevant agents. [Vert and Jacob, 2008] explains the problem in more detail. The objective of the data preparation is to derive useful features out of the examples in PIT dataset and combine the derived features to construct a richer dataset. We here refer to a weighted combination of the features illustrated in figure 1.

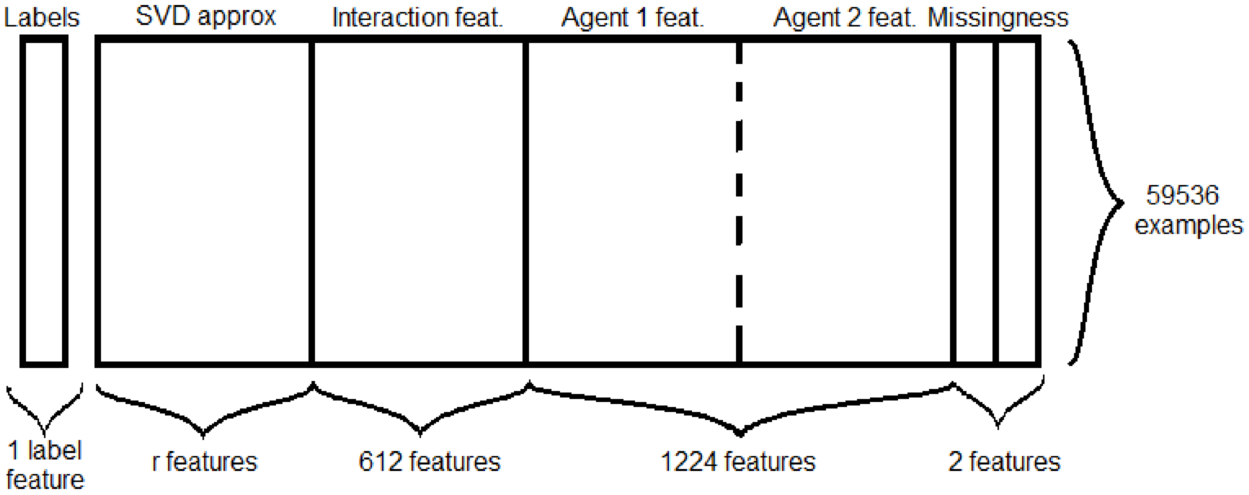

Figure 1: Features set

Structural Features. Structural features, which contain the social network information, are extracted using a low rank approximation of the adjacency matrix A taking inspiration from the method described in Section 5.2 of [Doi, 2010]. The adjacency matrix is decomposed by SVD in to matrix U, S and V such that $A = USV^T$. This can be reduced from a three-way decomposition to a two-way decomposition by setting $L = U\sqrt{S}$ and $R^T = \sqrt{S}V^T$ such that $A = USV^T = U\sqrt{S}\sqrt{S}V^T = LR^T$. The SVD operation orders the rows and columns of S by the values of its diagonal. This gives a natural interpretation where the first rows and columns are more important, i.e., contributes more, than the last once. We select the rank of the approximation by only keeping the first r columns of S. This naturally reduces L and R to also have only r columns. For the agent pair i and j we construct the structural features by applying componentwise multiplication of collumn i of L and j of R. This is transposed and repeated for all pairs, yeilding a 59536 by r feature set for link prediciton and 851 by r for colleague prediction. To avoid information leakage we construct this features set with only training data.

Descriptive Features. PIT dataset consists of 612 binary features for each of the 244 users. These features are descriptive features of the specific agent, but the description of these features is not available. Intuitively, the presence of a link may be partly driven by the features of both the agents. We represent this component of the final featureset by concatenating the 612 features of each agent for both agents. The resultant component is a featureset of 1224 features.

Pair-Wise Interaction Features. The discussed feature set represents a necessary, but by no means sufficient featureset. The presence of links and their types are also driven by interactions which only exist between two agents. We model the interaction between features as an element wise product of the descriptive features. As the descriptive features are binary, the product reduces to a bitwise `AND` operation which illuminates what we are achieving. If two agents have a feature in common then this will be allowed to contribute to the prediction through the `AND` operation. This featureset contains 612 features as a result of 612 pair-wise products of descriptive features.

Missing Features. The descriptive features matrix of the users is considerably sparse where some of the agents have significantly more features values with zeroes. The value of zero (0) could either mean that the value is actually nagative or that the value is missing. As the accompanying documentation of the PIT dataset does not provide any guidance on this issue, we proceeded to do a qualitative analysis of the feature data. From figure 2 we see that the majority of the agents have four or less posive features, i.e., feature value of one (1). This leads to a simple heuristic where we flag an agent if it has more than four active features. The flagging is implemented as a binary feature that is used once per agent.

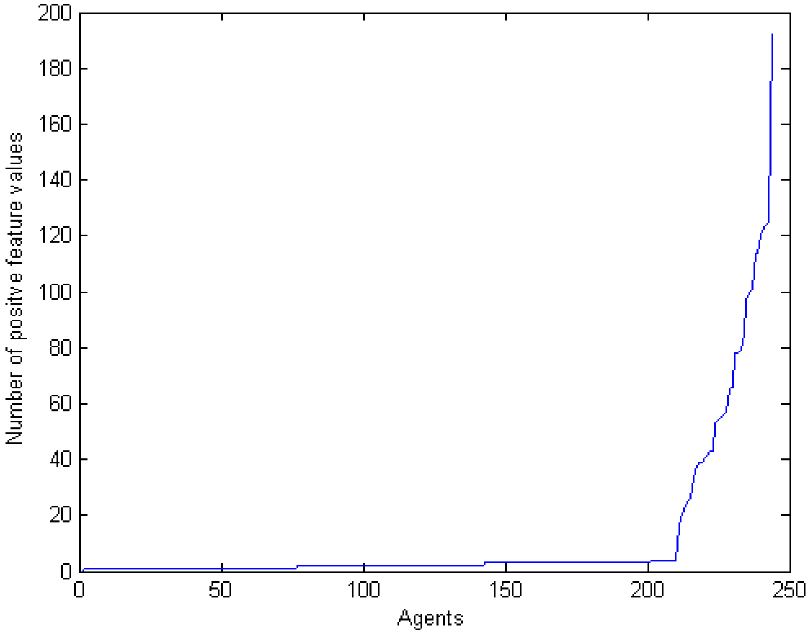

Figure 2: Number of features per agent sorted ascendingly.

4 Logistic Regression

In order to predict links between agents we implement a logistic regression model with regularization. Gradient descent is used to approximate the learning parameters. The model follows the setup of [Elkan, 2011] page 10. The following is the per-example update rule and prediction formula of the model.

$$\beta := \beta + \lambda[(y-p)x - 2\mu]$$

$$p = \frac{1}{1+\exp\{-\beta^T x\}}$$

Here the learning parameters are the elements of vector β, the training example is the 1-augmented vector x, the label is y, the learning rate is λ and the regularization strength is μ. The flowing modifications to the standard model is implemented.

Learning Rate. During learning the model uses MSE e_k on training data to gauge convergence at epoch k. When the MSE decreases, the learning rate stays constant, but if the trend changes and MSE increases, i.e. overshoots the local minimum, then the learning rate λ_k is halved. Mathematically that is $\lambda_k = \lambda_{k-1}/2$ if $e_k > e_{k-1}$. This prevents the learning sequence from entering an indefinite oscillation around a local minimum.

Termination. Learning is terminated when MSE falls below a positive termination threshold $\gamma > 0$ or when the maximum allowed number of epochs $k_{\max}$ is reached. That is, learning is terminated when $|e_k - e_{k-1}| < \gamma$ or $k > k_{\max}$.

5 Design of Experiments

Cross-validation. In order to protect from bias and freak results, all performance measures are calculated by using the arithmetical mean of the results from 10-fold cross-validation. This means, for the link prediction case, that we are training on datasets with 825 edges and trying to predict the remaining 92 hidden edges in the test dataset.

Selecting meta-parameters. There are in total 5 potential meta-parameters to set. The initial learning rate λ_0 can, within reason, be arbitrarily set since the learning rate converges to zero as learning progresses. The maximum number of epochs $k_{\max}$ is selected by trail and error constrained by available computational resources. The threshold γ which limits the change in MSE $|e_k - e_{k-1}|$ is selected as to yield desired precision. This leaves regularization strength μ and SVD decomposition rank r. These are found by grid search, however, we will later see that we need not look at all elements of the Cartesian product between the potential values of μ and r.

Prediction threshold. In general, a logistic regression model transforms a linear regression model that outputs values from the entire real line to a regression model that outputs values between zero and one. This lets us treat the output as monotonically increasing binary values or probabilities. We can force the output to take a specific binary value by using a threshold $\sigma > 0$. That is, for an output vector p we can produce binary output by $\mathbb{I}(p \succ \sigma)$ where $\mathbb{I}(\cdot)$ is the indicator function and "$\succ$" is a component-wise vector inequality. This allows us to set up a confusion matrix for prediction. The threshold is chosen such that the number of positive edge predictions on training data is equal to the number of labeled edges in the training data. We avoid using tuning the threshold using test data since this may be overfitting. In a practice we wouldn't have access to the number of edges in training data.

6 Results and Analysis

For the link prediction tasks we found, by trail-and-error, that $\lambda_0 = 0.01$, $k_{\max} = 100$, and $\gamma = 10^{-5}$ were suitable for our needs. Similarly for colleagueship prediction $\lambda_0 = 0.01$, $k_{\max} = 300$, and $\gamma = 10^{-6}$ were tractable paramaters.

6.1 Predicting agent links

Grid search for non-trivial meta-parameters. From [Doi, 2010] we excpect low ranks to be fruitfull, thus promting us to select the following search grid for the SVD approximation rank and regularization strength $r \in \{5, 10, 15, 20, 25, 30\}, \mu \in \{10^{-t}\}_{t=2}^{11}$. Tabel 1 shows the results of the meta-parameters grid search. With the parameter settings $\mu = 10^{-10}$ and $r = 20$ we achived the lowest MSE of 0.0108 on test data.

$\mu \setminus r$	5	10	15	20	25	30
10^{-4}	0.0129	0.0124	0.0122	0.0124	0.0127	0.0122
10^{-5}	0.0123	0.0123	0.0113	0.0139	0.0123	0.0121
10^{-6}	0.0123	0.0113	0.0118	0.0121	0.0114	0.0124
10^{-7}	0.0124	0.0123	0.0115	0.0114	0.0121	0.0119
10^{-8}	0.0117	0.0122	0.0118	0.0111	0.012	0.012
10^{-9}	0.0121	0.0114	0.0119	0.0112	0.0115	0.0136
10^{-10}	0.0116	0.0113	0.0117	**0.0108**	0.0120	0.0120
10^{-11}	0.0118	0.0115	0.0119	0.0109	0.012	0.0122
0	0.0121	0.0116	0.0122	0.0123	0.012	0.0123

Table 1: MSE on link test data for different regularization strengths μ and SVD rank r.

Threshold. With the best meta-parameters from the grid search we can proceed to find a threshold σ for prediction. As discussed a reasonable goal would be to predicts as many edges on training data as there are labeled edges in the training data. This is achieved with $\sigma = 0.1023$ which predicts the desired 825 edges on training data. This is illustreed in figure 3.

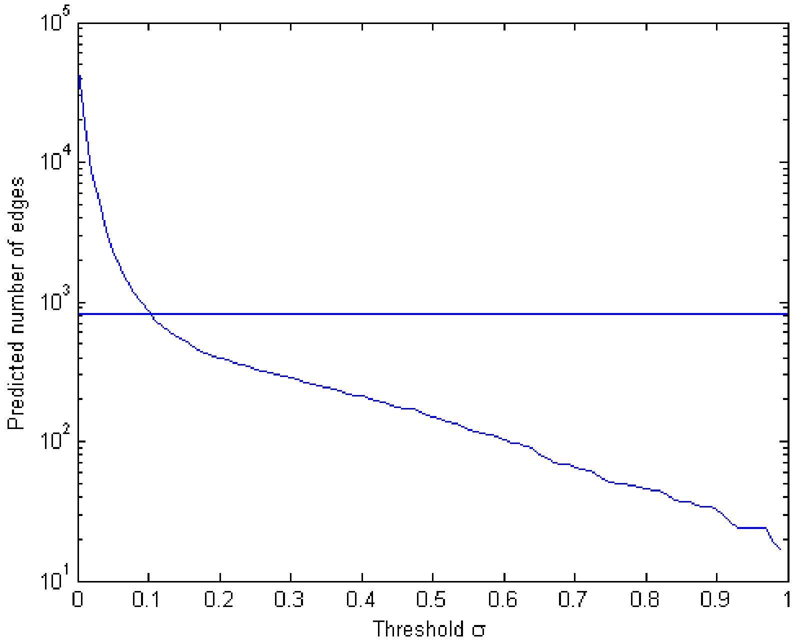

Figure 3: Search for link prediction threshold σ.

Performance. The confusion matrix in tabel 2 shows the results we obtained by implementing this model on test data. Precision is $0.341 = \frac{31}{31+60}$, accuracy is $0.981 = \frac{31+5808}{31+60+54+5808}$ and lift is $21.3 = \frac{31/(31+54)}{(31+60)/(31+60+54+5808)}$. Tabel 3 compairs the results with some trivial selection methods in order to evaluate performance. In order to achive the same precision with random sampling you would need to inspect $2030 = 0.341 \cdot (31 + 60 + 54 + 5808)$ edges.

	Outcome	
Prediction	tp = 31	fp = 60
	fn = 54	tn = 5808

Table 2: Confusion matrix of final link prediction results.

	Model	Select none	Select all	Select 92 random
Precision	0.341	0	1	0.0155
Accuracy	0.981	0.985	0.0155	0.970

Table 3: Performance of final link prediction model

6.2 Predicting Colleagueship

Prediction of colleagueship follows similarely from link prediction, except we here only look at examples with links present. We want to determine if a given link is a colleauge link or not. The same search grid for finding a suitable SVD aproximation rank is used, while the regularization strength has shifted to the following span.

$$\mu \in \{10^{-t}\}_{t=0}^{7}$$

The grid search results can bi found in table 4. We se that $k = 15$ and $\mu = 10^{-3}$ gave the lowest MSE of 0.1353 on training data.

Performance. Of the 766 examples in the training data there are 417 positive labels. If we pick a threshold of $\sigma = 0.6325$ the classifier predicts a satisfiably close 418 positive labels. Using the threshold on training data we get the confusion matrix in table 5 which we can use to calculate the performance of the classifier. Precision is $0.867 = \frac{39}{39+6}$, accuracy is $0.776 = \frac{31+35}{39+6+5+35}$ and lift is $1.674 = \frac{39/(39+5)}{(39+6)/(39+6+5+35)}$. Table 6 compairs the results with some trivial selection methods in order to evaluate performance.

$\mu \backslash r$	5	10	15	20	25	30
1	0.2398	0.2395	0.2351	0.2372	0.2400	0.2394
10^{-1}	0.1999	0.1986	0.1894	0.1941	0.1925	0.1917
10^{-2}	0.1853	0.1535	0.1482	0.1400	0.1390	0.1484
10^{-3}	0.1646	0.1512	**0.1353**	0.1631	0.1373	0.1627
10^{-4}	0.1820	0.1518	0.1928	0.1635	0.1565	0.1823
10^{-5}	0.1828	0.1458	0.1541	0.1518	0.1832	0.1622
10^{-6}	0.1886	0.1538	0.1655	0.1529	0.1516	0.1435
10^{-7}	0.1790	0.1621	0.1579	0.1690	0.1584	0.1574

Table 4: MSE on colleagueship test data for different regularization strengths μ and SVD rank r.

	Outcome	
Prediction	tp = 39	fp = 6
	fn = 5	tn = 35

Table 5: Confusion matrix of final colleagueship prediction results.

	Model	Select none	Select all
Precision	0.867	0	1
Accuracy	0.776	0.482	0.518

Table 6: Performance of final colleagueship prediction model

7 Conclusion

In this assignment we have explored social network link prediction and link type prediction. We used an SVD approximation with rank r to represent the networks structural features, while incorporating node, or agent, features directly and through a transformation using component-wise multiplication. Due to not knowing if features meant zero or missing we used a qualitative method to add a missingness feature per agent. Grid search was used isolate reasonable meta-parameters for the modified logistic regression model we used. Finally thresholds were used to give concrete predictions on test data. The thresholds were designed to mimic the training datas sparsely.

References

[Vert and Jacob, 2008] Vert, J.-P. and Jacob, L. (2008). Machine learning for in silico virtual screening and chemical genomics: New strategies. Combinatorial Chemistry & High Throughput Screening.

[Doi, 2010] Doi, E., LOW-RANK DECOMPOSITION AND LOGISTIC REGRESSION METHODS FOR LINK PREDICTION IN TERRORIST NETWORKS, http://cseweb.ucsd.edu/˜elkan/291/Eric.Doi_report.pdf

[Elkan, 2011] Elkan, Charles, Maximum Likelihood, Logistic Regression, and Stochastic Gradient Training, http://cseweb.ucsd.edu/ elkan/250B/logreg.pdf

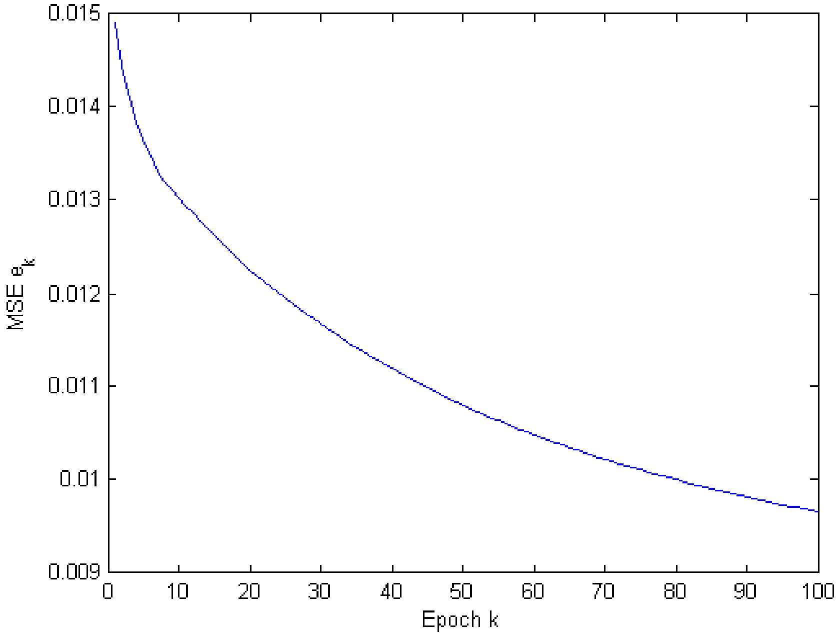

Figure 4: Learning error convergence of link prediction measured by MSE.