Social Network link Prediction

Study graph theory

Problem Statement: Predict missing edges/links from graph-based data

Data overview:

1. Given Directed graph
2. Train data
   1. Source node = u1 to u2, u1
   2. destination node – u2
   3. All edges originated from source u1 and corresponding destination
   4. 1.86 Million Users/Vertices
   5. 9.43 million edges
   6. Ui,uj pairs
   7. One snapshot of data at a specific time t.
   8. No meta info provided

Terms:

1. Nodes – dots/vertex – each Individual User
2. Edges – connection between dots – can be friend or follower
3. Undirected Graph – set of nodes and set of edges
4. Directed Edges
5. Path – path in a graph is a sequence of edges
   1. U4 🡪u2-🡪u3 = valid path with length of 2, 2 edges on this path
6. Undirected edges
7. Directed graph has directed edges

U1 -🡪 U2 = u1 following u2 – called as directed edges

Map to a Supervised Classification Problem:

1. Presence of edge as 1
2. Absence of edge as 0
3. Common vertices between u1 and u2
4. U1 – u3,u4,u5
5. U2 – u3,u4,u6
6. So u1 might want to follow u6, or u2 might follow u5, or u1 might follow u2
7. If u1 follow u2, there high change that u2 might follow u1 – back follow feature

Featurization

1.number of common vertices being followed by both u1 and u2

2.is there an edge between u1 u2, binary feature

Challenge: how to featurize

Business Constraints and Metrics:

1.Predict probability scores, to recommend highest probability links to user

2.Suggest connections most likely to be correct – high precision, high recall, F1 Score,

Precision @ top k

EDA Basic Stats:

Networkx library – Studying graphs and networks

Data size large

Reading graph using readedgelist

Each node represented as integer

In degree of a vertex – vertex coming from u2 to u1 source

Out degree – from u1 source to u2 destination

EDA follower and following stats:

* + - 1. Number of followers for each person – equal to in degree
      2. Boxplot number of followers – outliers

Percentile

90% of users have less than 12 followers

99% have 40 or fewer followers

Histogram – pdf

Sharp spike – very few have 40 followers

Number of people each person is following

Out degree analysis

EDA Binary classification task:

Ui,uj – 1 – edge, 0 – not edge

|  |  |  |
| --- | --- | --- |
| Source | Destination | Yi |
| U1 | U5 | 1 |
| U1 | U6 | 1 |
|  |  |  |

No not edge data/0 labelled data points in train data present

Add following computation

|  |  |  |
| --- | --- | --- |
| Source | Destination | Yi |
| U1 | U2 | 0 |
| U1 | U3 | 0 |
|  |  |  |

Create random subset of pairs of vertices from 9.43 million pairs of vertices using random sample, to make the dataset balanced

Create All the possible edges that are possible but are not present

Generate links from graph which are not in graph and whose shortest path is greater than 2.

As less than 2 might potentially connect later

Dataset constructed

EDA: Train and test split considering temporal nature of data

Temporally changing data

In real world, time-based split is the right way to split data

Currently we do not have timestamp provided, so we do simple random split.

Cold start problem:

Number of users present in test data, but not in train data

Feature Engineering on graphs:

**Jaccard & Cosine Similarities**

1. Jaccard:
   1. Construct set of all users who follow u1
   2. Construct set of all users who follow u2
   3. Jaccard distance = j = |x n y|/|x u y|
   4. Larger the j distance, higher the probability of edges
   5. Jaccard for followers, jaccard for followees
2. Cosine:
   1. Cosine coefficient can be computed between sets
   2. Cosine distance between two vectors
3. Pagerank:
   1. Pagerank is an algorithm used by google search to rank webpages
   2. If page B has larger links, if lot of quality pages on internet are linking to B then B must be importance page. So page B has top rank
   3. Pagerank algo outputs probability distribution to represent likelihood person randomly clicking will arrive at a particular page
   4. Given a directed graph, pagerank algo will give a score to determine how important a page is
   5. Alpha – pagerank hyperparameter
   6. Mean pagerank value for imputation
   7. Pagerank for source, and destination – two features
4. Shortest path:
   1. Length of path from ui to uj
   2. Ui following u1 and u1 following uj
5. Connected components:
   1. Weakly connected comp – subset of vertices if ignore directions of edges in directed graph,
   2. Strongly connected comp – subset of vertices that can. Go from any vertex to any vertex
   3. Help us detect communities
   4. Binary feature – same weakly connected component or not
6. Adar Index:
   1. To predict links in social media
   2. Adar index between two vertices X and y
7. Kartz centrality:
   1. Relative degree of influence of node within a social network
   2. Similar to pagerank or eigen vector centrality
   3. Kc for each of the nodes
8. Does the person follow back?
9. HITS algorithm
   1. Popular algorithm for search
10. SVD:
    1. Matrix fact can be used for feature engineering
    2. Use SVD to come up with features on directed graph
    3. Adjacency matrix concept in graph theory – binary matrix of 1 and 0
    4. SVD on Adj matrix
    5. 1.78 Million X 1.78 Million is a sparse matrix
    6. Decompose matrix 1.78 columns into 6 columns by applying SVD
    7. A = u Sigma VT
    8. U – 1.78 M X 6
    9. Sigma – 6 X 6
    10. VT = 6 X 1.78 M
    11. 4 6 dimensional vectors = 24 features
11. Weight Features:
12. Modeling:
    1. Nonlinear model might work well
    2. Random Forest classifier work well for nonlinear data,
    3. RF doesn’t need interaction features
    4. GBDT and SVM work well on this data
    5. Standard hyper parameter tuning – depth, sample split, estimators
    6. F1 score train – 96.5, test – 92.4
    7. Confusion matrix
    8. Precision class 0, class 1
    9. Recall for class 1 is dropping – could add more features to improve
    10. Followers back is the most important feature