Github Collaborator Recommendation

Shan Zhong, Bill Wen, Albert Jian

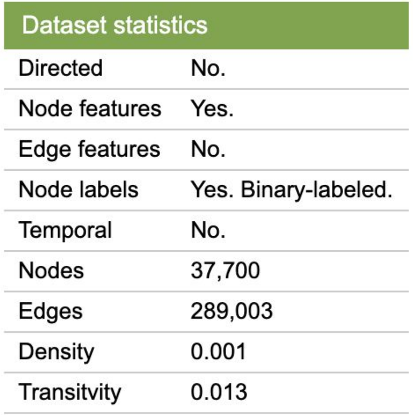
Math 168 | Spring 2020 | Professor Heather

Introduction

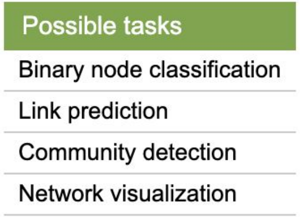
- This project aims to predict the probability of a link to recommend the highest probability links to a Github user.
- Exploratory data analysis was conducted to further understand followers and following statistics
- Binary classification method (Random Forest, Support Vector Machine and Gradient Boosting) was adopted to classify good links and bad links.
- SVD was applied to reduce the dimension of the dataset.
- Similarity measures such as jaccard distance, cosine distance and preferential attachment were calculated for featurization.
- Ranking measures (page rank), centrality measures (katz centrality) and other metrics such as shortest path and common neighbors were also added for model training.
- Confusion matrix was plotted at the end to evaluate model performance.

Dataset Information

- Source: Stanford Large Network Dataset Collection
- A large social network of GitHub developers which was collected from the public API in June 2019.
- Nodes are developers who have starred at least 10 repositories and edges are mutual follower relationships between them. The vertex features are extracted based on the location, repositories starred, employer and e-mail address.

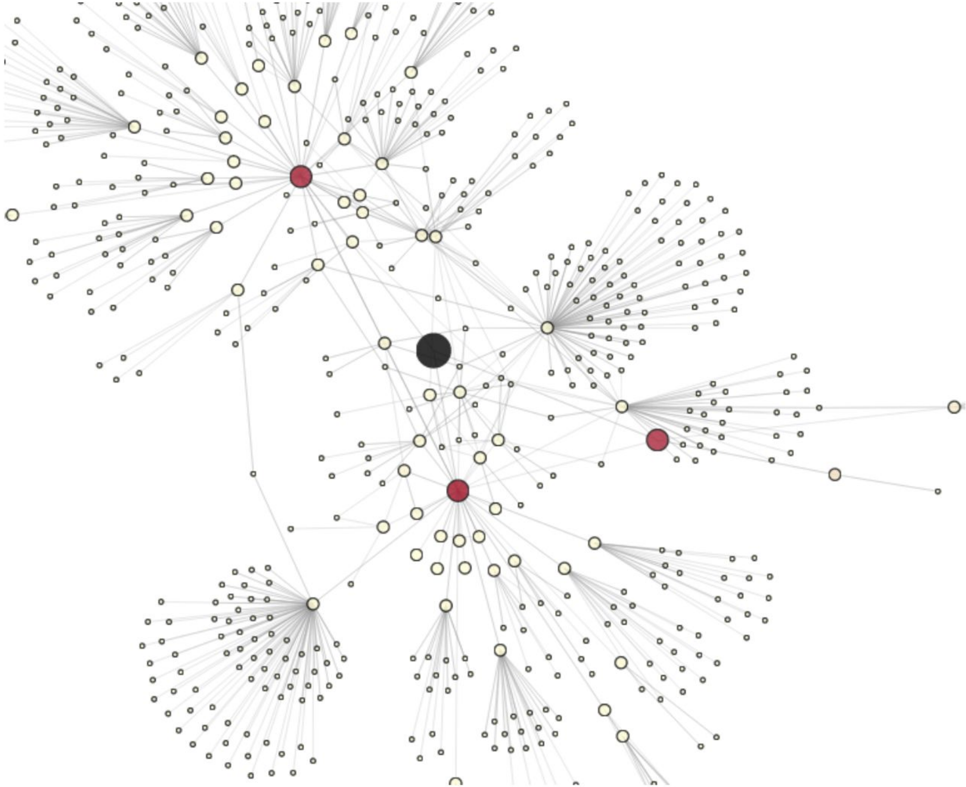


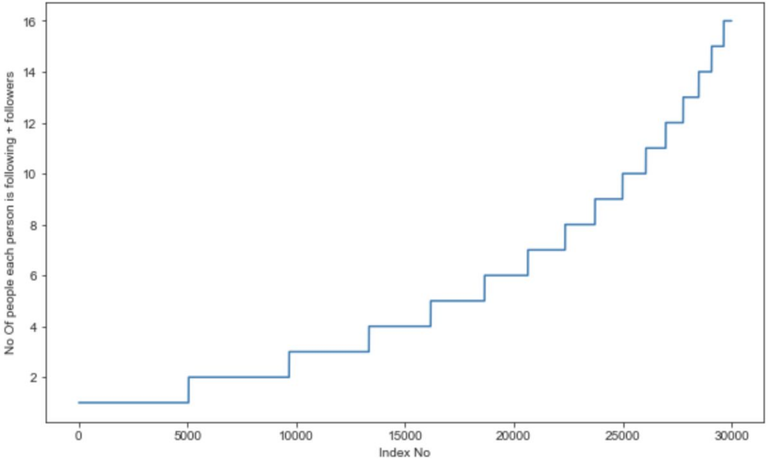
Dataset statistics	
Directed	No.
Node features	Yes.
Edge features	No.
Node labels	Yes. Binary-labeled.
Temporal	No.
Nodes	37,700
Edges	289,003
Density	0.001
Transitvity	0.013



Possible tasks
Binary node classification
Link prediction
Community detection
Network visualization

Exploratory Data Analysis





- The largest number of followers and followees is 9458 and the minimum number of followers and followees is 1.
- 5045 people have minimum number of followers and followees. Number of people having followers and following less than 10 is 2499.

Featurizations

Jaccard Distance:

- Assume that u1,u2,u3,u4,u5,u6 are connected,such that u1 followers-{u3,u4,u5} and u2 followers-{u3,u4,u6}.
- We could find how dissimilar these two sets are using Jaccard distance.
- The Jaccard similarity coefficient compares members for two sets to see which members are shared and which are distinct. It’s a measure of similarity for the two sets of data, with a range from 0% to 100%. The higher the percentage, the more similar the two populations. Although it’s easy to interpret, it is extremely sensitive to small samples sizes and may give erroneous results, especially with very small samples or data sets with missing observations.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

Featurizations

Cosine Distance:

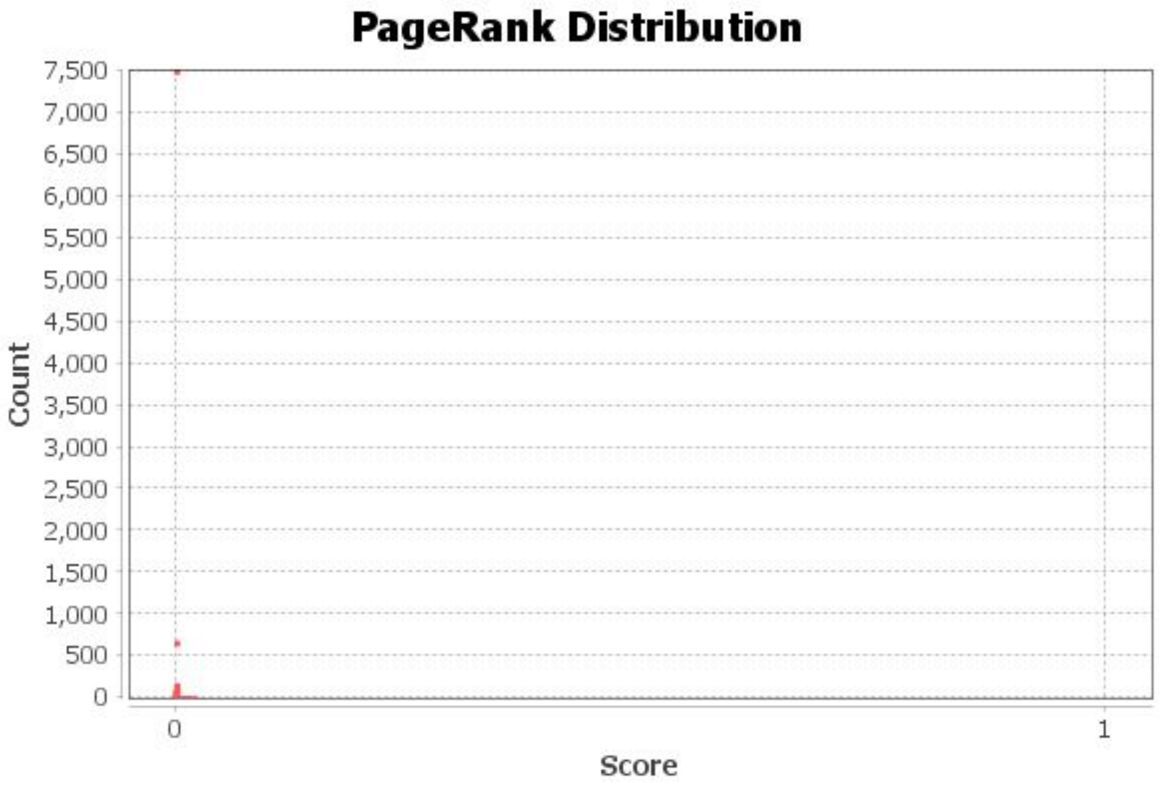
- Like Jaccard distance we could also calculate another distance which is an extension to cosine similarity.

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}},$$

PageRank:

- Imagine whole internet as directed graph where each web page is node.
- Each page is given a rank depending on number and quality/importance of links/edges.
- It gives you a probability value for each of the web page that represents the likelihood of a random person clicking on a page to arrive at another page.

$$PR(p) = \frac{1-d}{N} + d \sum_{i} \frac{PR(i)}{NumLinks(i)}$$


PageRank Distribution
Count
7,500
7,000
6,500
6,000
5,500
5,000
4,500
4,000
3,500
3,000
2,500
2,000
1,500
1,000
500
0
0
1
Score

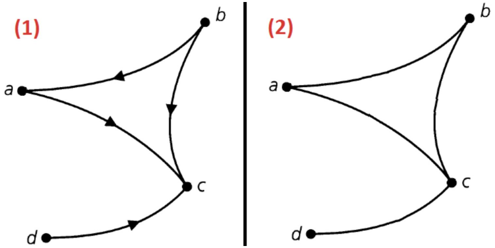
Featurizations

Shortest Path:

- Getting Shortest path between two nodes, if nodes have an edge i.e, trivially connected then we are removing that edge and calculating the shortest path.
- Else assign -1

Connected Component:

- A digraph is strongly connected if for every pair of distinct vertices u and v there exists a directed path from u to v)
- A digraph is weakly connected if for every pair of distinct vertices u and v there exists an undirected path (potentially running opposite the direction on an edge) from u to v.



Featurizations

Katz Centrality:

- Introduced by Leo Katz in 1953 and is used to measure the relative degree of influence of an actor (or node) within a social network.
- Katz centrality similarly like page rank that measures influence by taking into account the total number of walks between a pair of nodes

$$x_i = \alpha \sum_j A_{ij} x_j + \beta$$

SVD features:

- We computed adjacency matrix
- We then decomposed it using SVD of components 6 which results in two matrices left singular and right singular matrix.
- Now both the matrices could be used as features containing vector of 6-dimensional

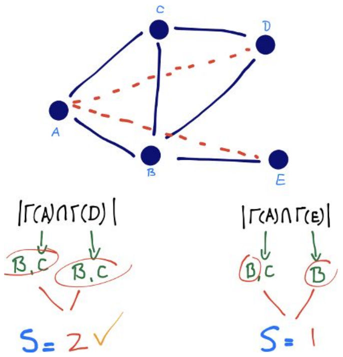
Featurizations

Common Neighbors:

- Newman [7] has computed this quantity in the context of collaboration networks, verifying a positive correlation between the number of common neighbors of x and y at time t, and the probability that x and y will collaborate at some time after t.

$$\text{Score}(x,y) = |\underbrace{\Gamma(x)}_{\text{Neighbors of } x} \cap \Gamma(y)|$$

list comparison : O(V . VlogV)



Featurizations

Weight feature:

- Taken from a research paper - Graph-based Features for Supervised Link Prediction by William Cukierski, Benjamin Hamner, Bo Yang
- Intuitively, consider one million people following a celebrity on a social network then chances are most of them never met each other or the celebrity. On the other hand, if a user has 30 contacts in his/her social network, the chances are higher that many of them know each other.
- In order to determine the similarity of nodes, an edge weight value was calculated between nodes. Edge weight decreases as the neighbor count goes up.
- Since the graph is directed,weighted in and weighted out are differently calculated:

$$w_i^{in} = \frac{1}{\sqrt{1 + |\Gamma_{in}(v_i)|}}$$

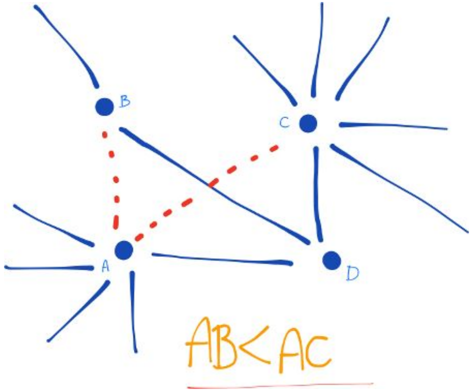
$$w_i^{out} = \frac{1}{\sqrt{1 + |\Gamma_{out}(v_i)|}}.$$

Featurizations

Preferential Attachment

- One well-known concept in social networks is that users with many friends tend to create more connections in the future. This is due to the fact that in some social networks, like in finance, the rich get richer. We estimate how ”rich” our two vertices are by calculating the multiplication between the number of friends (|Γ(x)|) or followers each vertex has. It may be noted that the similarity index does not require any node neighbor information; therefore, this similarity index has the lowest computational complexity.

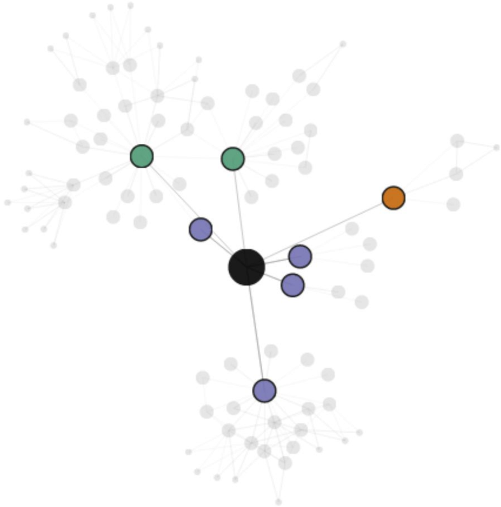
$$\mathrm{Score}(x,y) = |\Gamma(x)| \cdot |\Gamma(y)|$$



The link between A and C is more probable than the link between A and B as C have many more neighbors than B

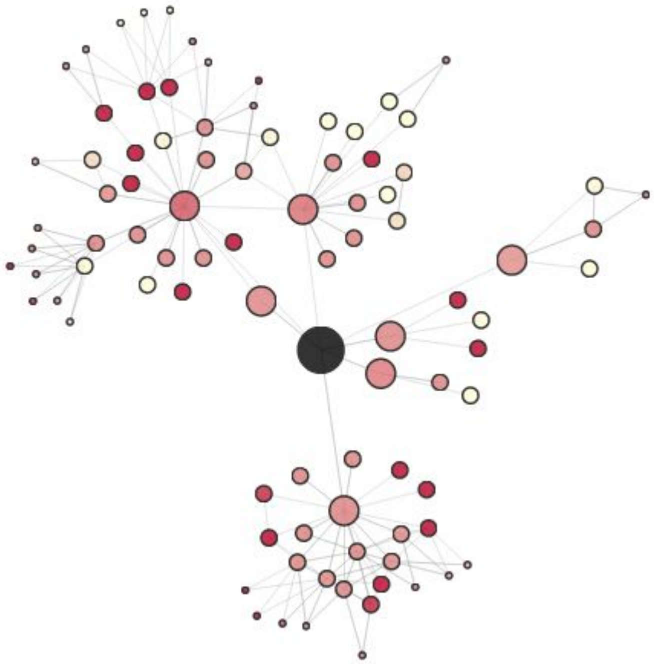
Feature Understanding

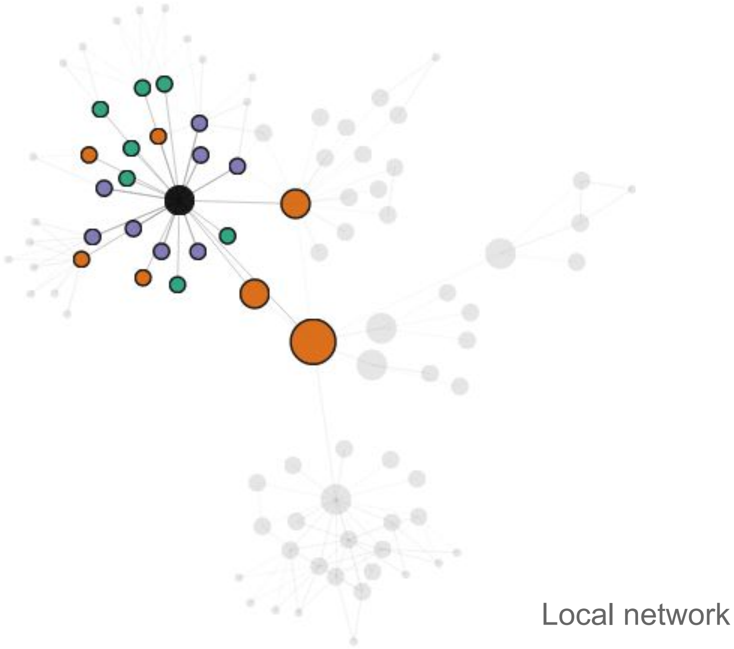
- If we look at the following vs. follower relationships of the central node, we can see that, as expected (because edges that represented both following and follower were double-weighted in my PageRank calculation), the darkest red nodes are those that are friends with the central node, while those in a following-only or follower-only relationship have a lower score.



Feature Understanding

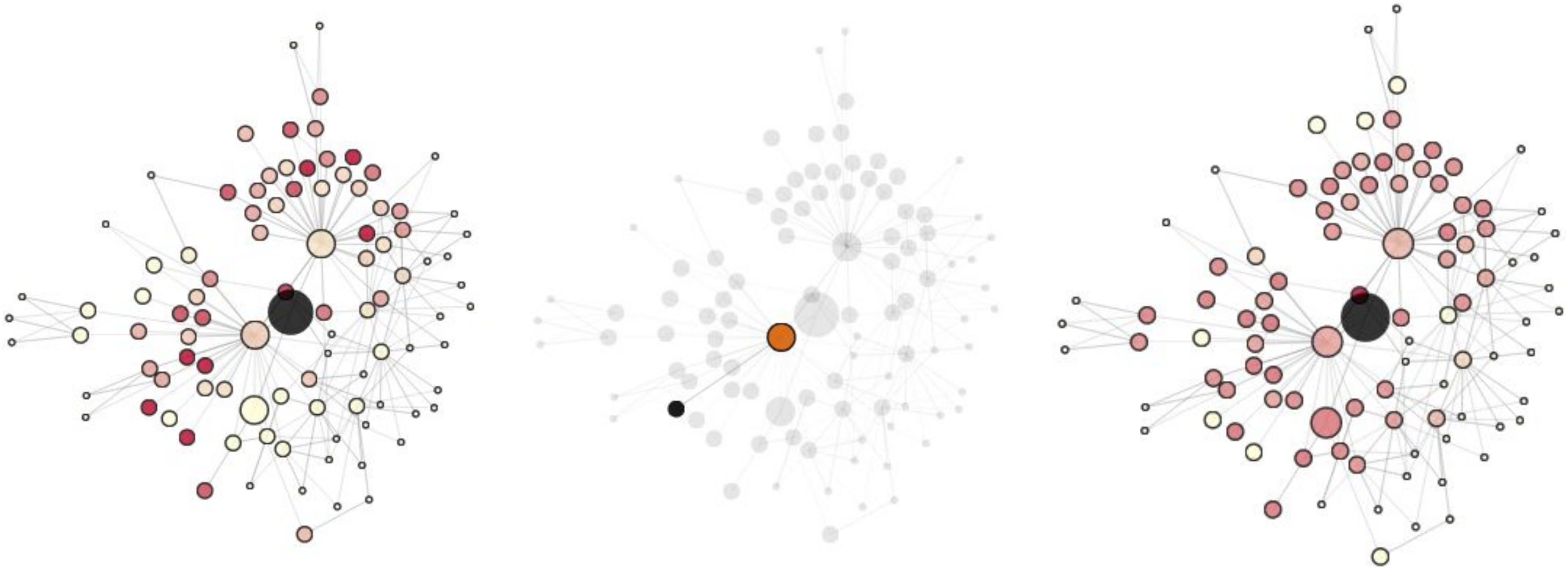
- How does the propagation score compare to personalized PageRank?





Feature Understanding

- Here, each node is colored according to its Jaccard similarity with the source:

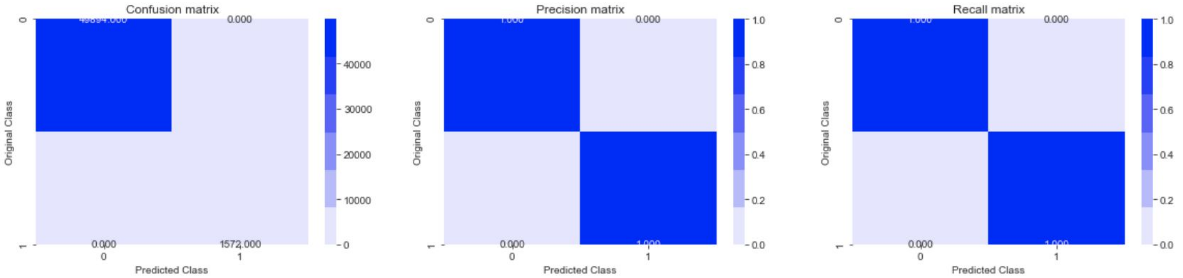


Regularized version of Jaccard similarity

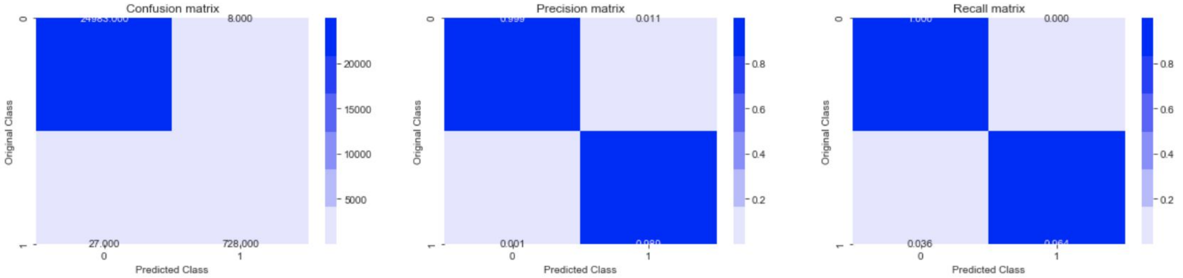
Results

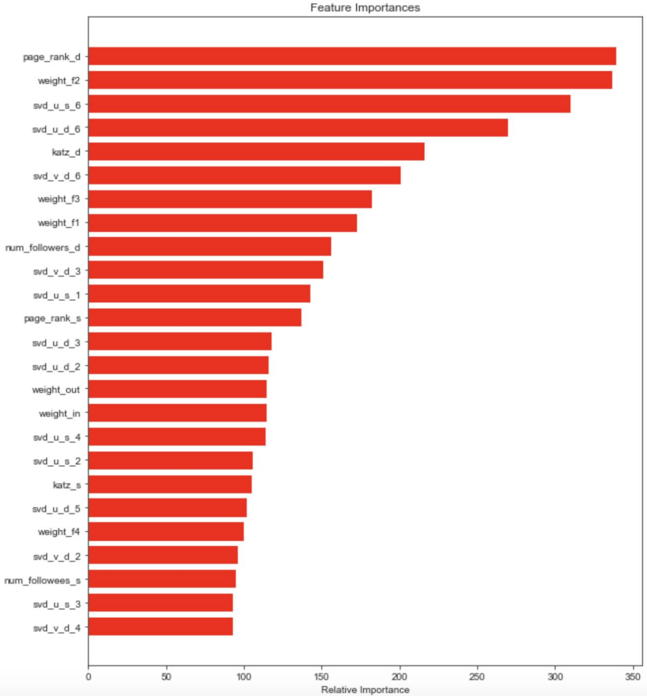
```
+---------------------------------+-----------------------------------------+----------+----------+
|              Model              | Hyperparameters(max_depth,n_estimators) | Train F1 | Test F1  |
+---------------------------------+-----------------------------------------+----------+----------+
|                RF               |                 (14,121)                |   0.96   |   0.92   |
| GBDT After Feature Engineering  |                 (10,200)                |   0.99   |   0.99   |
+---------------------------------+-----------------------------------------+----------+----------+
```

Train confusion_matrix



Test confusion_matrix





Thank you!

W. Cukierski, B. Hamner, and B. Yang.
Graph-based features for supervised link prediction.
In *The 2011 International Joint Conference on Neural Networks*, pages 1237–1244, 2011.

Benedek Rozemberczki, Carl Allen, and Rik Sarkar.
Multi-scale attributed node embedding.
arXiv preprint arXiv:1909.13021, 2019.