

Evolution of Protein-Protein Interaction Networks from influential seeding

Computational Systems Biology | Feb - May 2021

RA Keerthan (CH17B078)

J Mahesh (CH17B049)

Agenda

01. Introduction

02. Related works

03. Our contributions

04. Implementation and Results

05. Conclusion

Introduction

The study of evolution of protein-protein interaction networks (PPIN) has been a key research area in network biology owing to its importance in understanding several biological processes and mechanisms.

One of the widely recognized approaches to approximate the evolution of PPIN is the duplicate-diverge model that was proposed by [3].

However, [3] did not discuss the choice of the right seed network from which the network will evolve. To study the impact of the choice of seed network, [1] proposed to choose two cliques of maximum sizes present in the original PPIN data.

In this work, we present our implementation of the duplicate-diverge model proposed by [3] by choosing the seed network proposed by [1] and study the effects of varying the seed along with proposing a PCA based evaluation technique.

Related Works

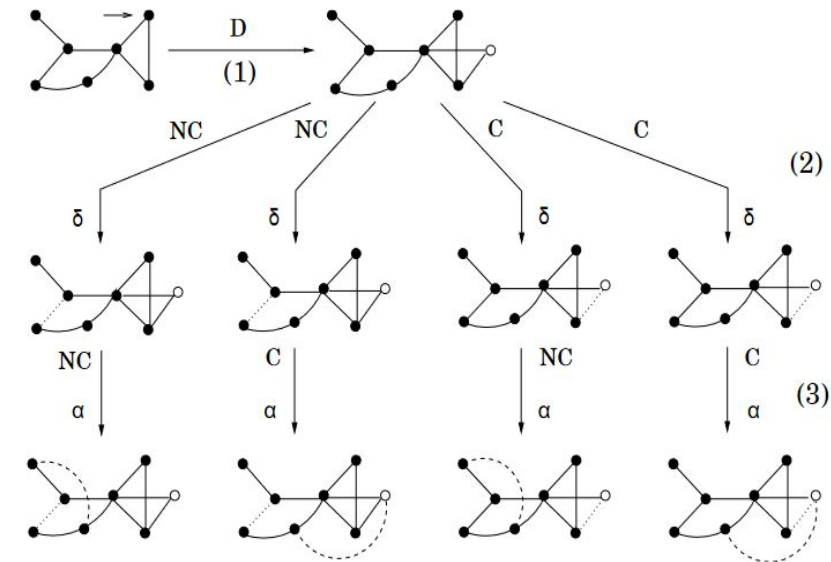
Duplicate-Diverge Algorithm

The Duplicate-Diverge algorithm for growth of a network model, as proposed by [3], is as follows:

- Select a node at random and duplicate it preserving its connectivity.
- With probability δ , remove an edge emanating from the newly formed node.
- With probability α , form an edge between newly formed node and rest of the nodes.

In this algorithm, there are two parameters:

- δ , the *new node edge removal probability*
- α , the *new node edge addition probability*.



Seeding Technique

- [1] proposed that the choice of the network seed graph is an important factor for determining the performance of the model.
- A seed graph is the graph from which network evolution takes place; it is the core of the PPIN. Seed graphs are much smaller in size in comparison to the networks in study.
- As stated by [1], the duplicate and diverge model does not produce large cliques. Therefore, [1] proposed to set the **seed graph to the top two largest cliques** present in the PPIN. We refer to this technique as *clique based seeding* approach.
- The large amount of singletons produced by DD model is addressed by *deleting any singleton nodes* if they form after any step.

Drawbacks of clique based seeding

- Time complexity associated with finding cliques scales exponentially with the number of nodes present in the network. The time complexity of finding all maximal cliques in a network with n nodes is $O(3^{n/3})$ [8].
- The cliques may not accurately represent the natural structure of a network before growth. Cliques are formed rarely and larger cliques take many iterations of growth before formation, hence are not suited to be seeds.
- Clique-based seeds are usually identical for networks of similar size. An alternative seeding technique could take into account more differences between the network structures.

Finding influential nodes

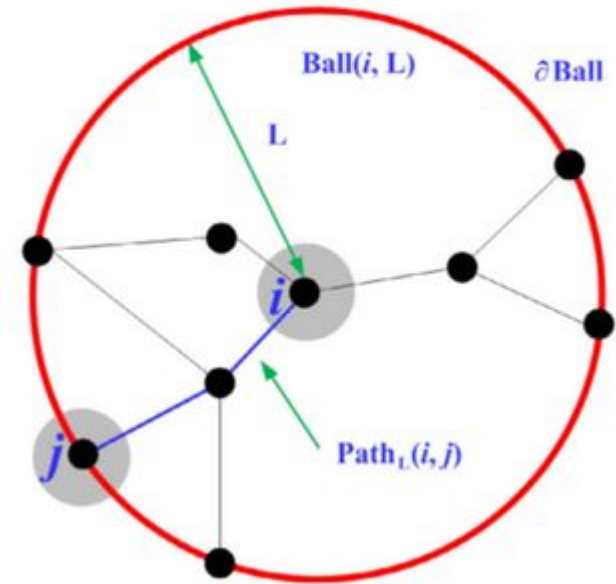
We adopt the approach presented in [2] to identify the *minimal set* of influential nodes whose removal will cause the whole network to collapse.

The basic idea is to remove the nodes in decreasing order of *Collective Influence*, denoted as **CI**, until the network collapses. **CI** of node i is formulated as a function of degree of node i ($k(i)$) as follows:

$$\mathbf{CI}_L(i) = (k(i) - 1) \sum_{j \in \text{Ball}(i, L)} (k(j) - 1)$$

The event of collapse of a network is identified when the largest eigenvalue of the *coupling matrix* is equal to unity. This concept of network collapse is based on percolation theory [2,7].

Finding influential nodes in a network with n nodes has the time complexity of $O(n \log n)$ [7].



Our contributions

1. Influential seeding

The first task is to determine N_{seed} number of influential nodes as the seed graph while maintaining the inter-node connectivity, if any.

We set the size of the seed N_{seed} to be approximately 1.5% of the total number of nodes in the network. This choice is arbitrary. It ensures we have a reasonable, but not too large, amount of nodes in the seed graph.

Then, we find the minimum set of influential nodes of the network. We take at most N_{seed} of these nodes, and form the seed by considering the subgraph formed out of these nodes.

Once we are set with the seed graph, we will follow the Duplicate-Diverge model to simulate the evolution of PPIN.

Our contributions

2. PCA based distribution analysis

[4] proposed a PCA based method to identify important centrality measures by constructing a *network-centrality* matrix, a matrix with number of rows and columns equal to number of nodes and centrality measures used respectively.

We slightly modified their technique by using the distribution of values along principal components for comparison of networks. This idea was inspired by the ability of principal components to capture intricate relationships between the centrality measures present in the network-centrality matrix.

In this work, we chose *Degree centrality*, *Betweenness centrality* and *Closeness centrality* as the base centrality measures while constructing the network-centrality matrix. We project the network centrality matrices of the original, clique-based and influential-based networks to the space spanned by the top two principal components of the original network. This will result in three *reduced-centrality* matrices. Two-sample KS Test is performed on these reduced-centrality matrices.

Implementation and Results

Parameter Estimation

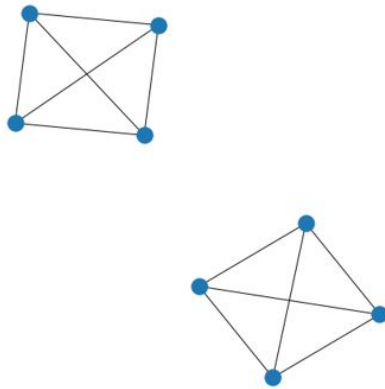
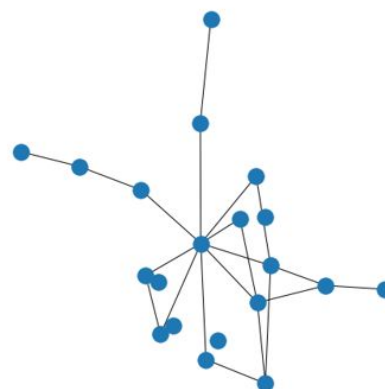
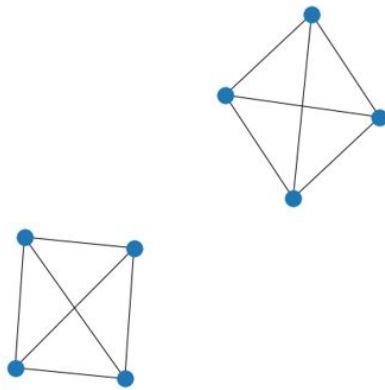
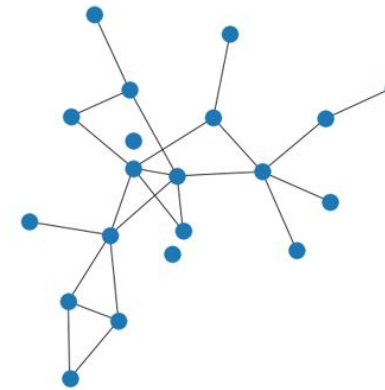
- All computation that is presented in this work were implemented in Jupyter Notebook using Python 3.7 and the networkx library.
- We used yeast PPIN [CCSB-Y2H [5]] and worm interactome [WI-2007 [6]] dataset for testing the model. The CCSB-Y2H dataset consisted of 1280 proteins and 1810 interactions (inclusive of self-loops). WI-2007 dataset consisted of about 1498 proteins and 1817 interactions (inclusive of self-loops).
- Parameter Estimation was carried out by performing a grid search to minimize the objective function shown on the right.

$$L(G_m, G_o) = (\epsilon + |E_m - E_o|) \times \frac{\left(\frac{1}{Z_{deg}+1} + Z_{deg} + 1\right)}{2}$$

Where,

- G_m, G_o stand for the model-generated graph and original network respectively.
- E_m, E_o stand for the number of edges in their respective graphs.
- ϵ is a non-negative number to make objective function non-zero. Our chosen value for ϵ was 1.
- Z_{deg} is the KL Divergence between the degree centrality distributions of the original graph and the network model.

Seed graphs

	Clique-based seed	Influential seed
CCSB-Y2H dataset		
WI-2007 dataset		

Number of nodes in seed graph		
	Clique-based seed	Influential seed
CCSB-Y2H	8	20
WI-2007	8	20

Network statistics comparison

CCSB-Y2H dataset

Optimal params	Clique-based seed	Influential-seed
α	0.001	0.001
δ	0.790	0.822

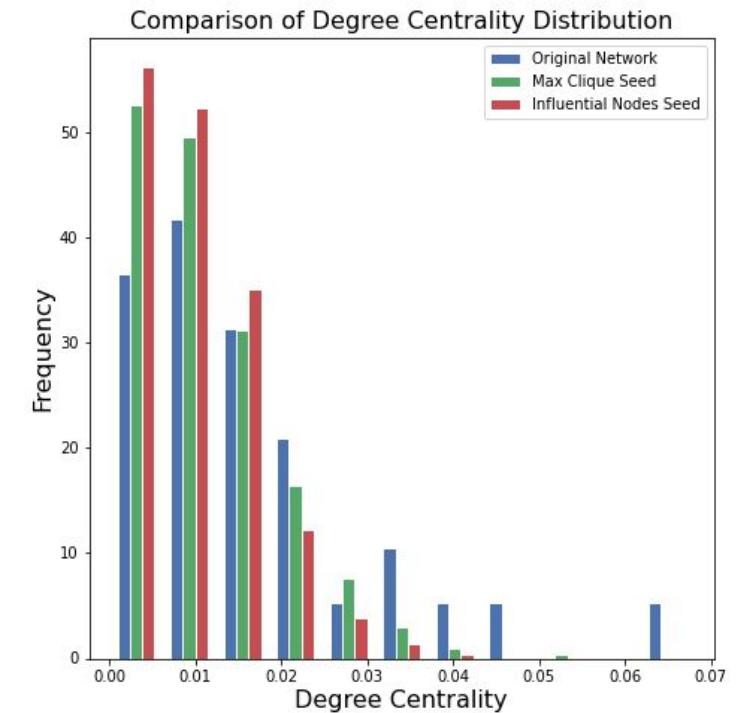
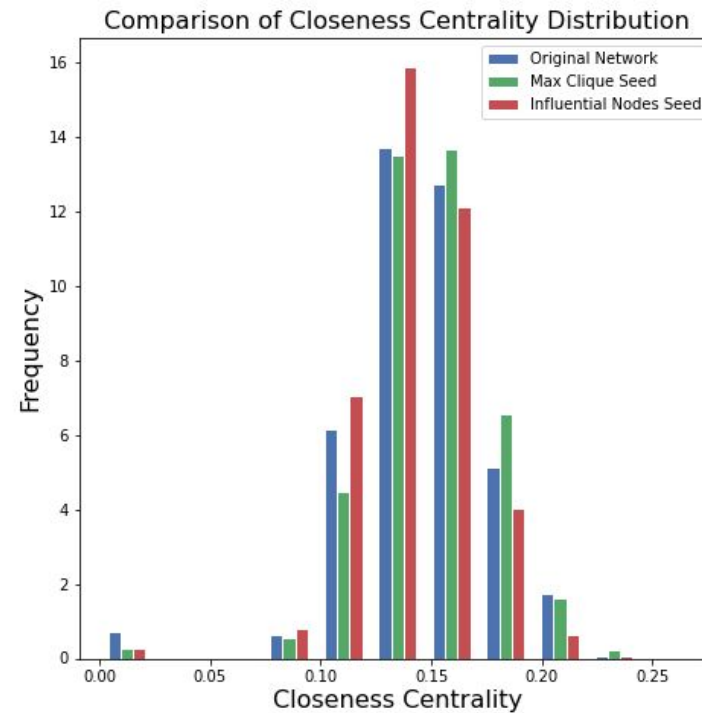
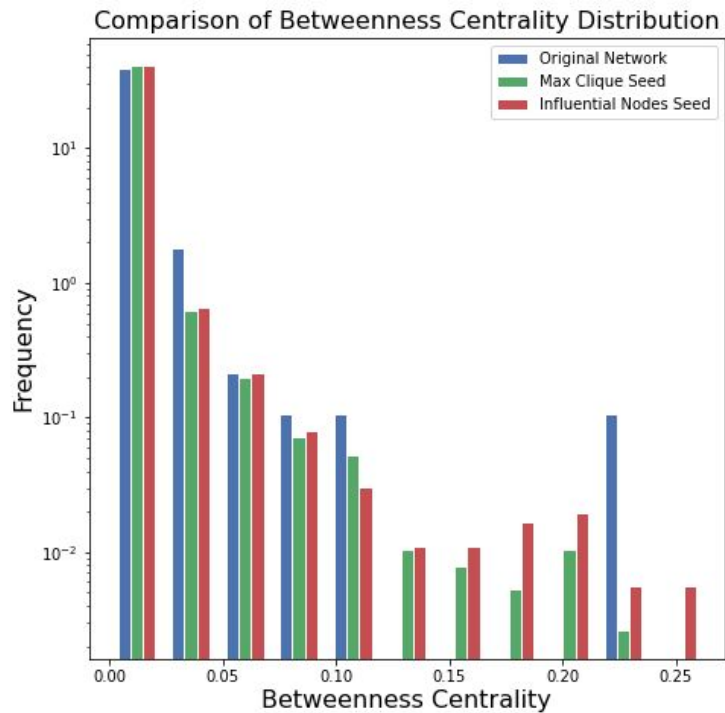
	original network	clique-based seed	influential-seed
Avg. edges	1810	1848	1769.4
Avg. Avg. Degree	2.828	2.888	2.765

WI-2007 dataset

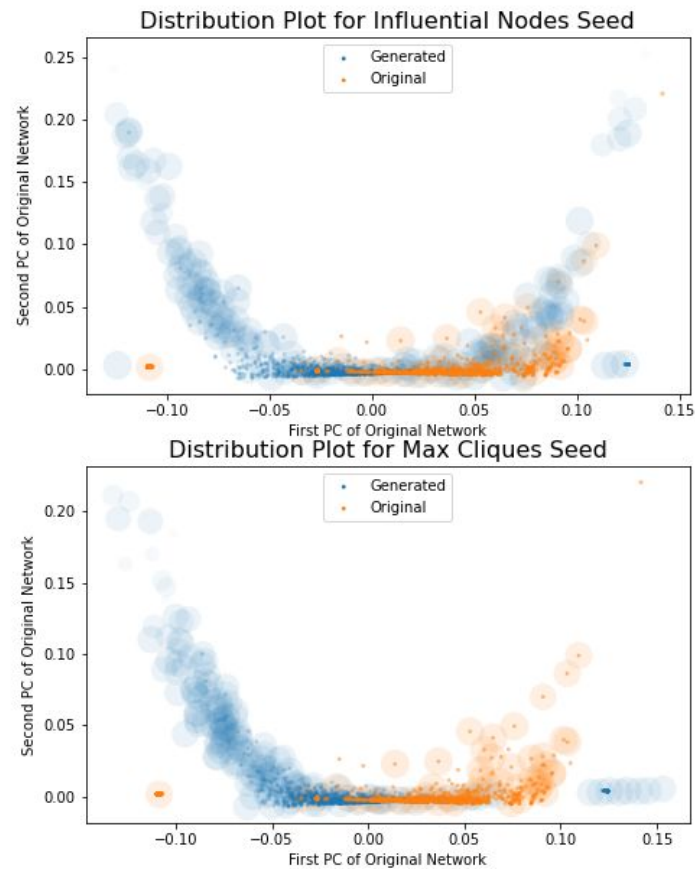
Optimal params	Clique-based seed	Influential-seed
α	0.001	0.001
δ	0.895	0.895

	original network	clique-based seed	influential-seed
Avg. edges	1817	1781.4	1821.4
Avg. Avg. Degree	2.426	2.378	2.432

Comparison of centrality distributions - Y2H data

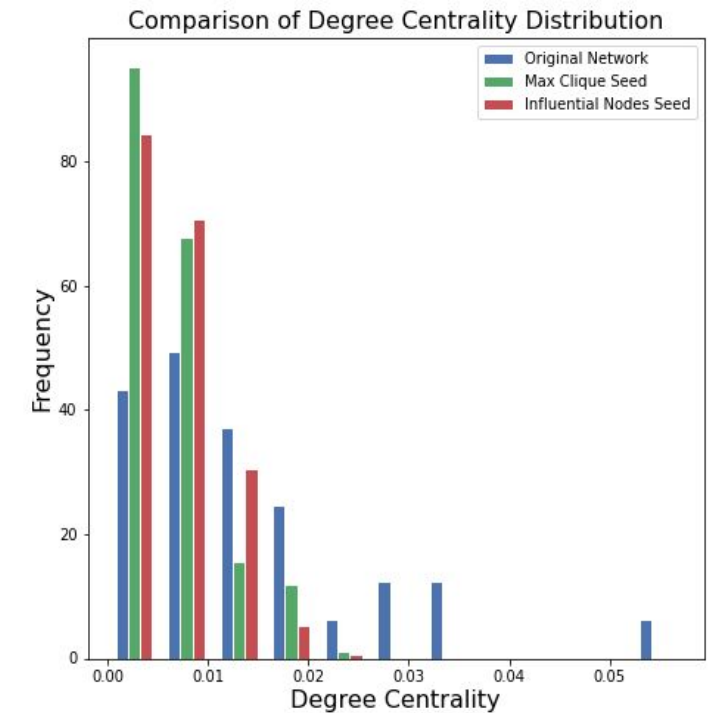
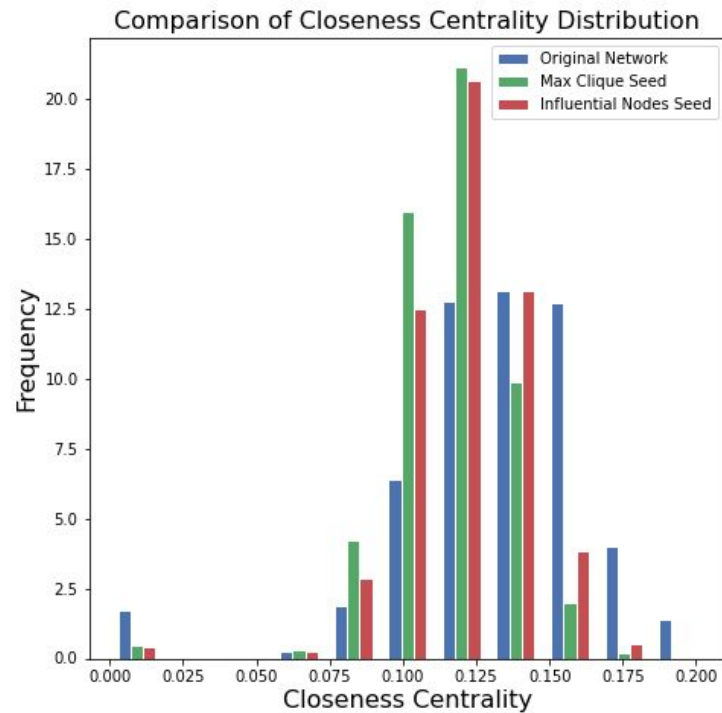
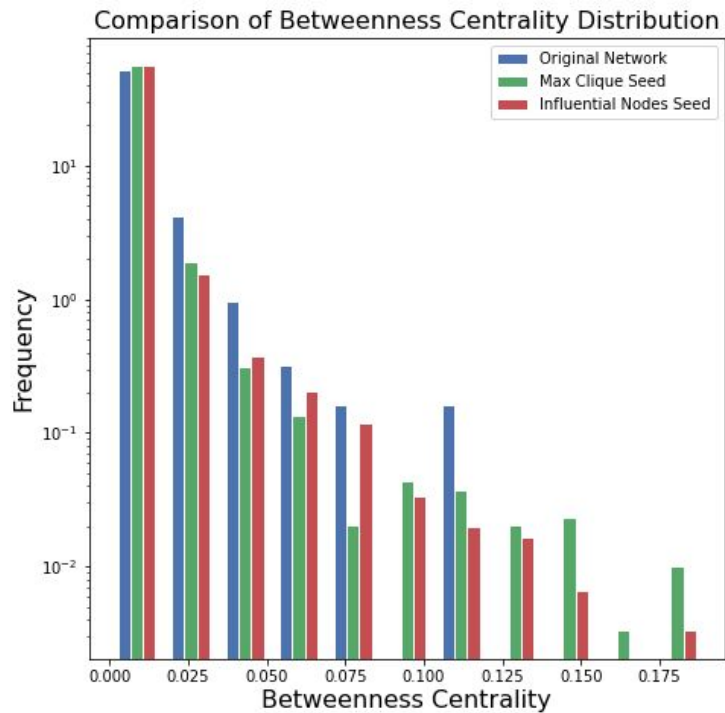


Comparison of reduced-centrality distribution - Y2H data

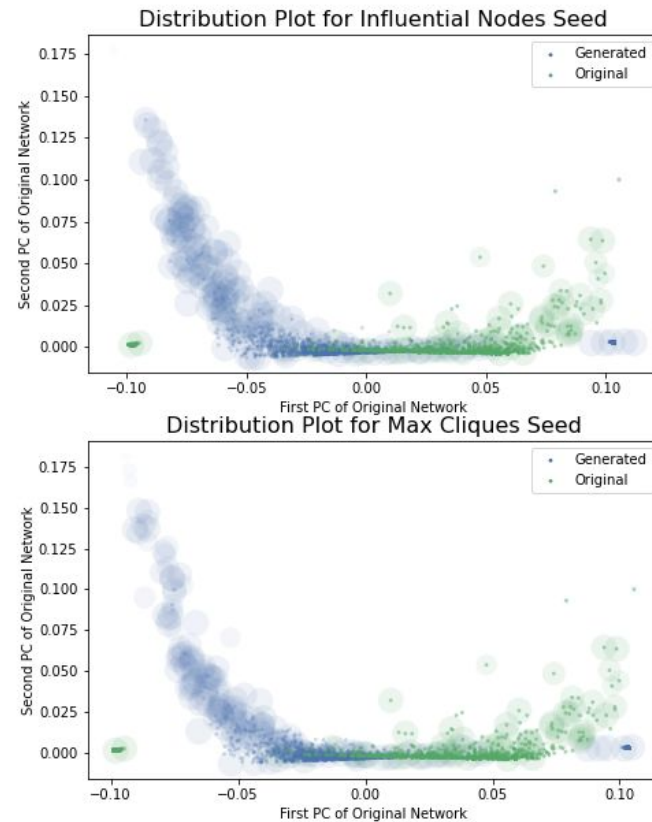


	Clique vs Original	Influential vs Original
KS Test Statistic	0.431	0.378

Comparison of centrality distributions - WI data



Comparison of centrality distributions - WI data



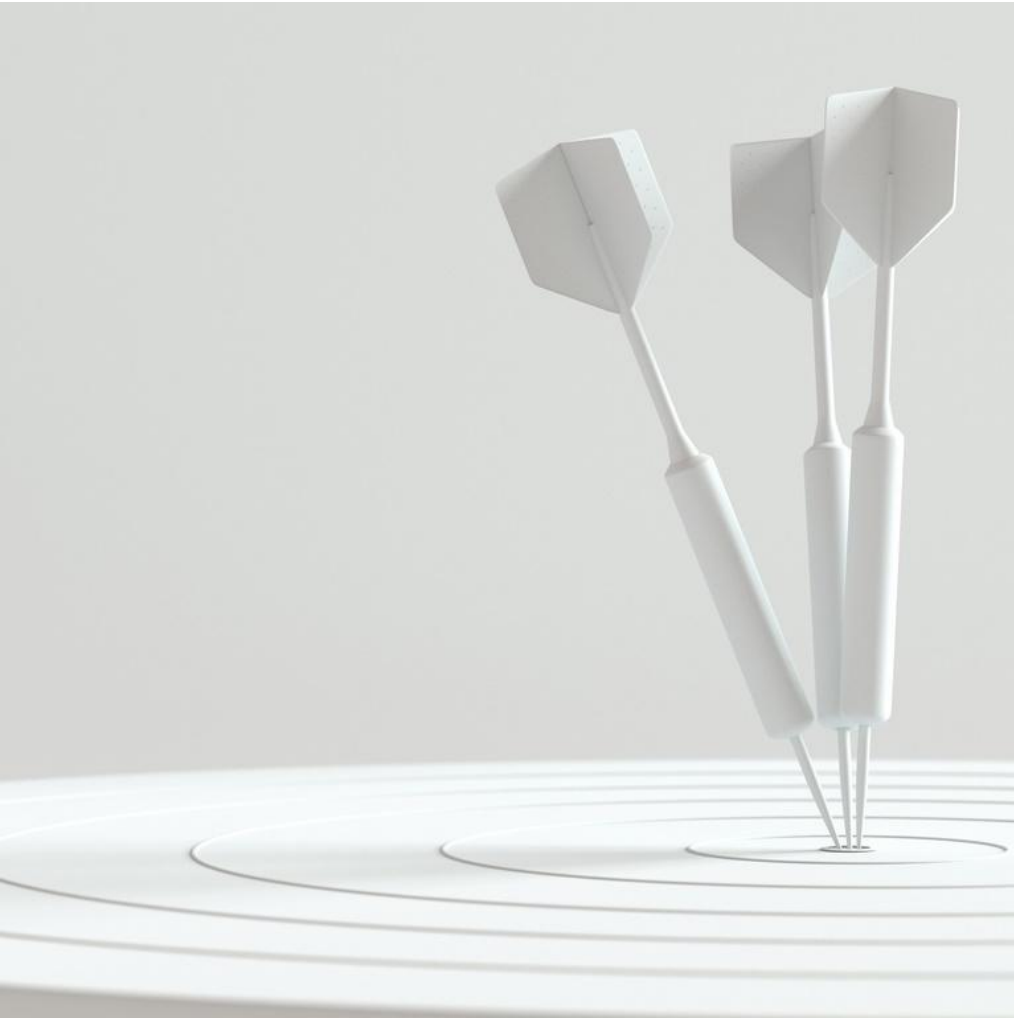
	Clique vs Original	Influential vs Original
KS Test Statistic	0.449	0.446

Conclusion

- The existing clique-based method for choice of seed was observed to have some drawbacks.
- We proposed an alternate method of constructing the seed graph, using the Collective Influence algorithm, which is of lower time complexity and could potentially capture more information.
- We also proposed a model evaluation technique, using PCA-based dimensionality reduction on the network-centrality matrix.
- Results showed us that the proposed seeding method performs better than clique-based method on CCSB-Y2H dataset and equalled the performance on WI-2007 dataset.
- However, the current work relies on a grid-search for parameter estimation, which could take long time for large datasets. More optimized parameter estimation techniques for estimation of α and δ , like GA-based methods, could be a potential research area in the future.

References

- [1] Hormozdiari F, Berenbrink P, Pržulj N, Sahinalp SC (2007) *Not all scale-free networks are born equal: The role of the seed graph in PPI network evolution*. PLOS Computational Biology 3(7): e118. doi:10.1371/journal.pcbi.0030118
- [2] Peng Gang Sun, Yi Ning Quan, Qi Guang Miao, Juan Chi, *Identifying influential genes in protein–protein interaction networks*, Information Sciences, Volumes 454–455, 2018, Pages 229-241, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2018.04.078>.
- [3] Pastor-Satorras R, Smith E, Solé RV. *Evolving protein interaction networks through gene duplication*. J Theor Biol. 2003 May 21;222(2):199-210. doi: 10.1016/s0022-5193(03)00028-6. PMID: 12727455.
- [4] Ashtiani, M., Salehzadeh-Yazdi, A., Razaghi-Moghadam, Z. et al. *A systematic survey of centrality measures for protein-protein interaction networks*. BMC Systems Biology 12, 80 (2018). <https://doi.org/10.1186/s12918-018-0598-2>
- [5] CCSB-Y2H Dataset, Yeast Interactome Project, CCSB Interactome Database, Harvard Medical School.
- [6] WI-2007 Dataset, Worm Interactome Project, CCSB Interactome Database, Harvard Medical School.
- [7] Morone, F., Min, B., Bo, L. et al. *Collective Influence Algorithm to find influencers via optimal percolation in massively large social media*. Sci Rep 6, 30062 (2016). <https://doi.org/10.1038/srep30062>
- [8] Etsuji Tomita, Akira Tanaka, Haruhisa Takahashi, *The worst-case time complexity for generating all maximal cliques and computational experiments*, Theoretical Computer Science, Volume 363, Issue 1, 2006, <https://doi.org/10.1016/j.tcs.2006.06.015>.



Thank you

We are open for any questions that you may have.