Complex Networks Comparison using Graphlets

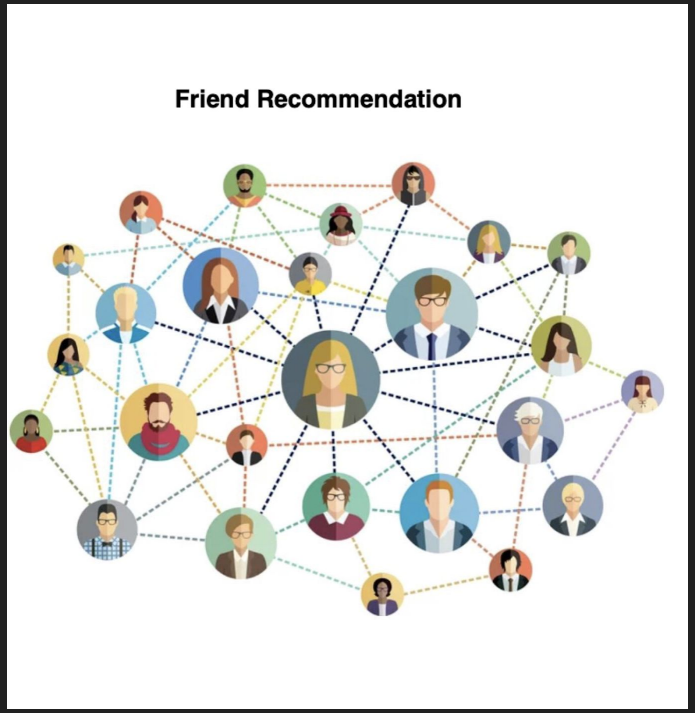
- Jiarong Li

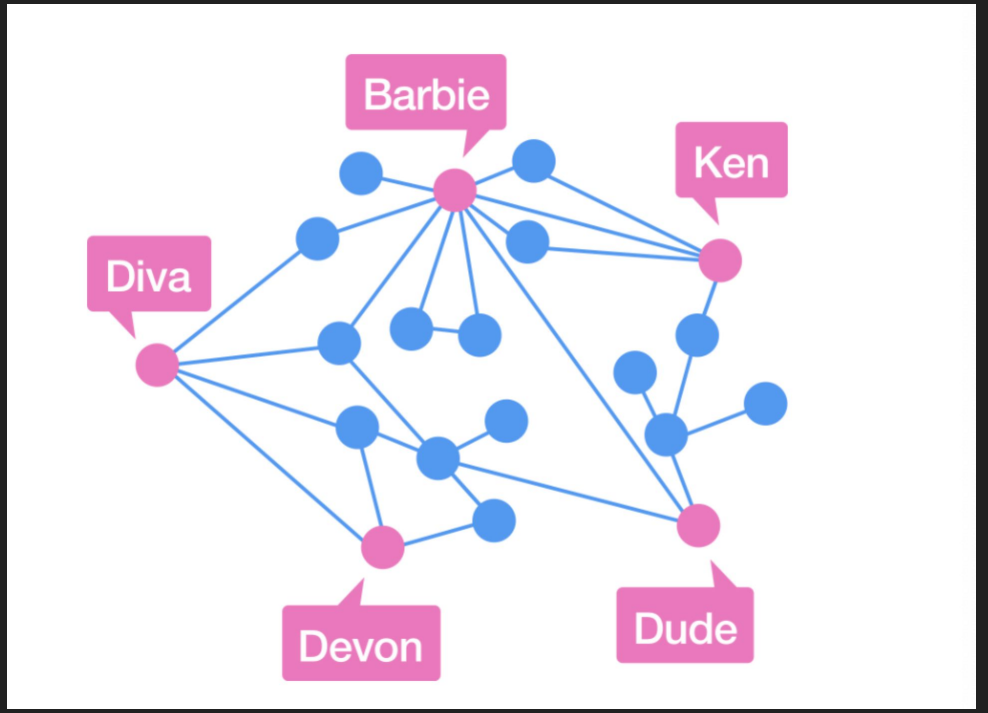
Networks

A very flexible and powerful way of modeling many real-world systems

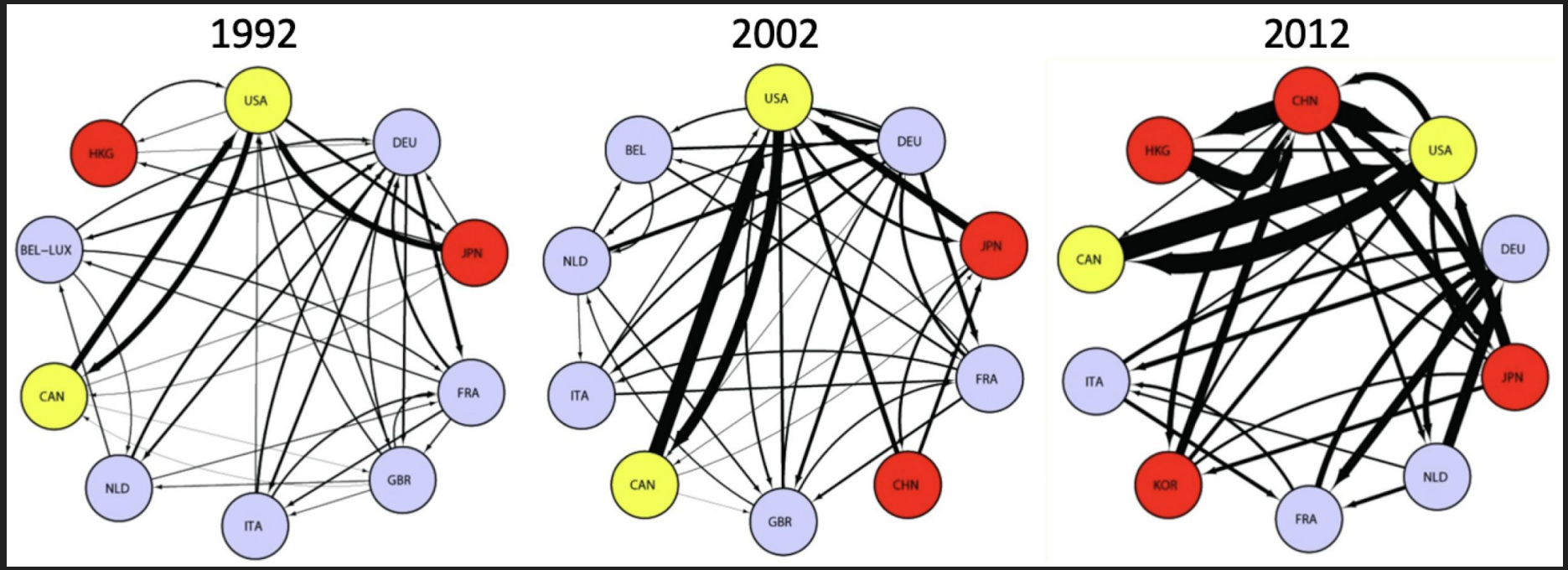
It represents the relationships among individuals or among different entities in some collection.

Social Networks

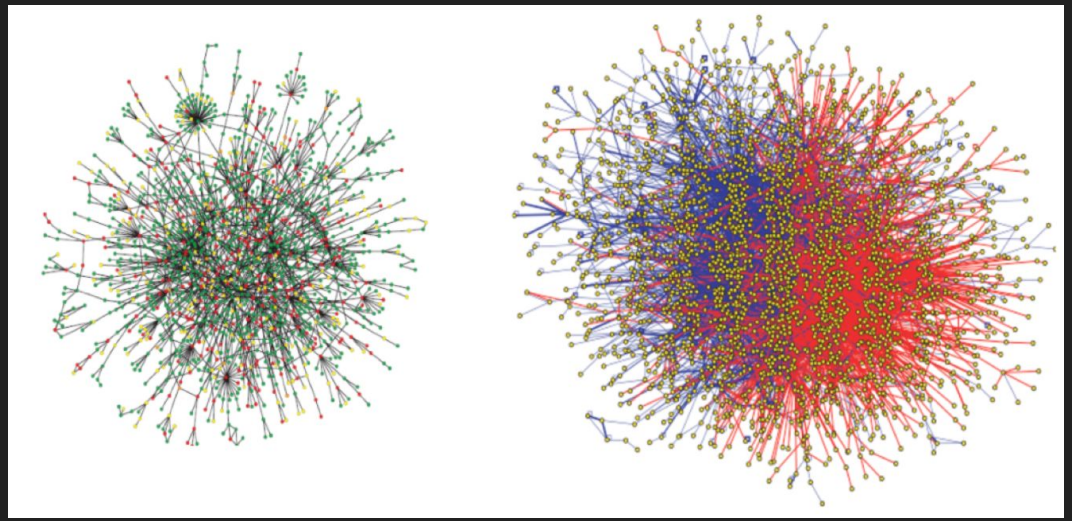




World Trade Networks



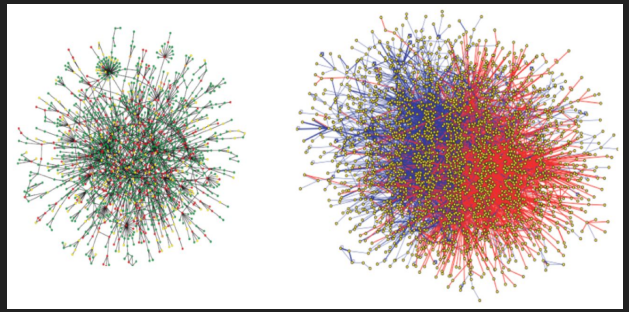
PPI Networks Reconstruction



Yeast protein Interactome

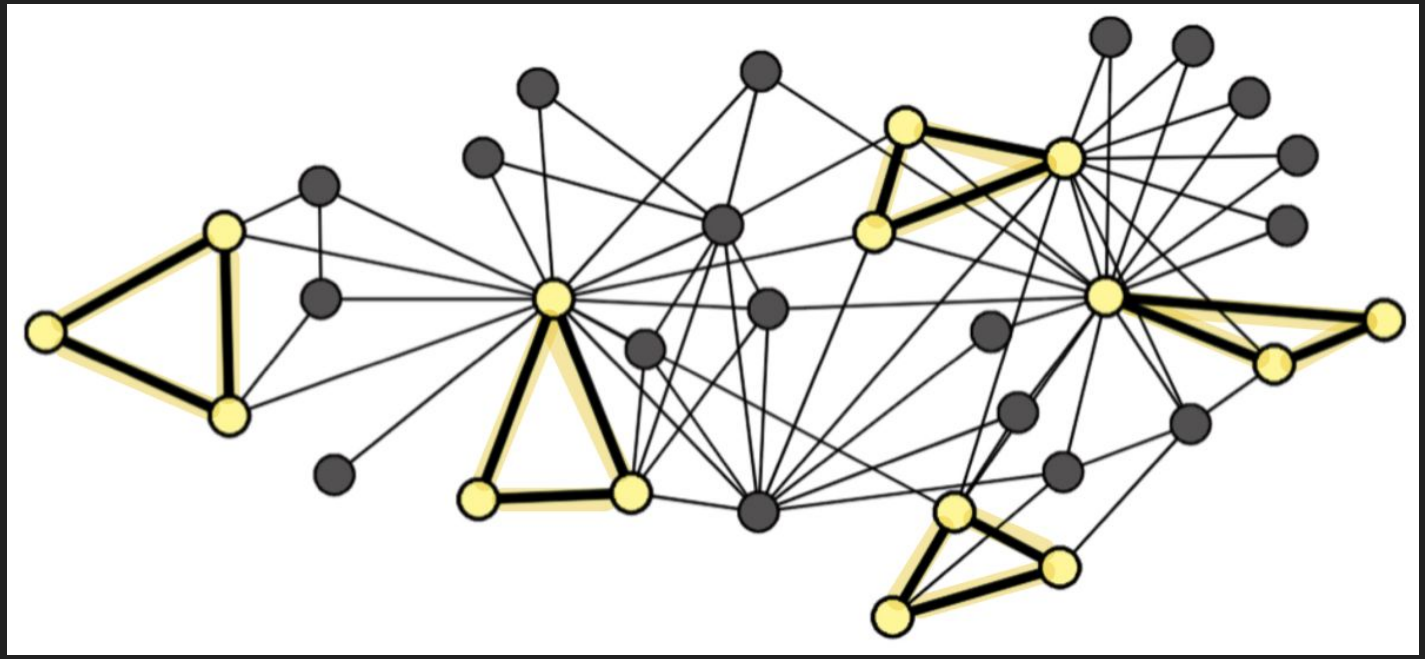
Human Interactome

How do we compare networks?

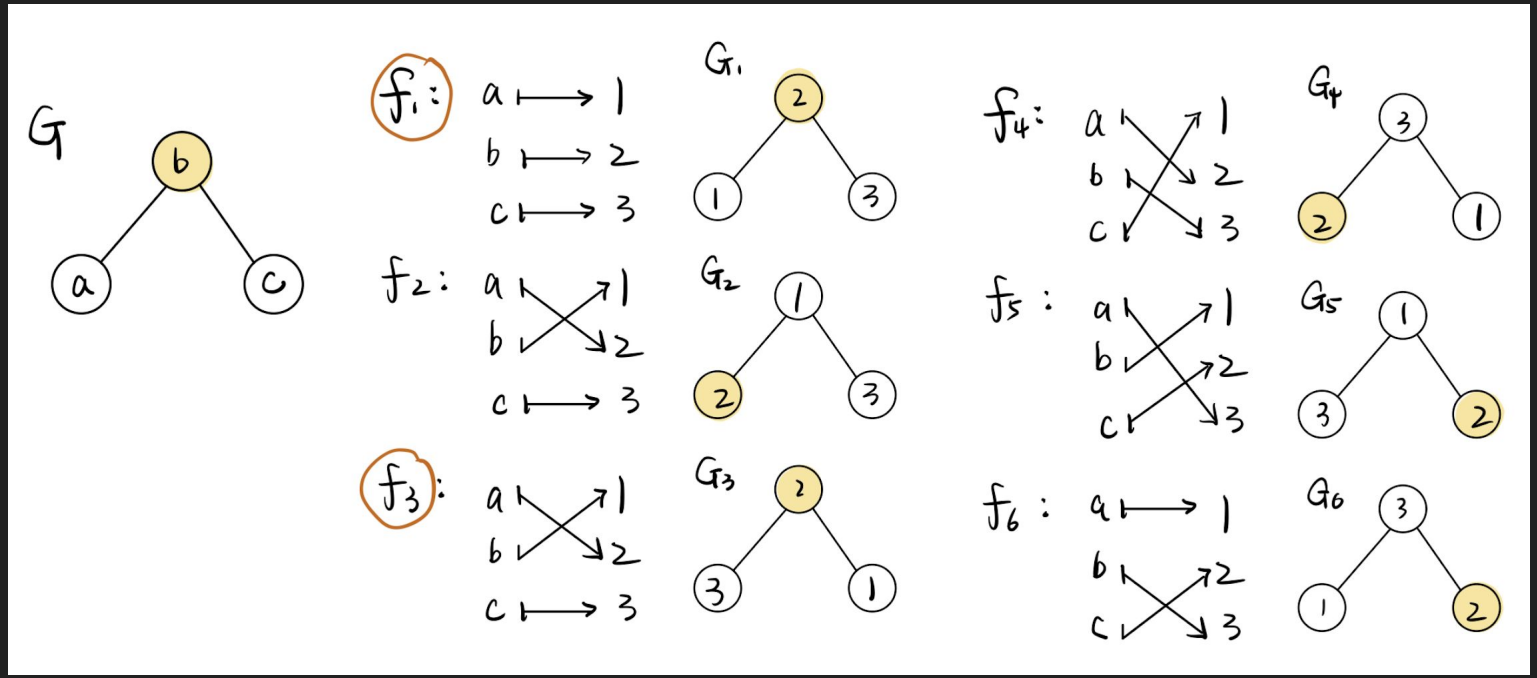


Substructures of a huge network: subgraphs

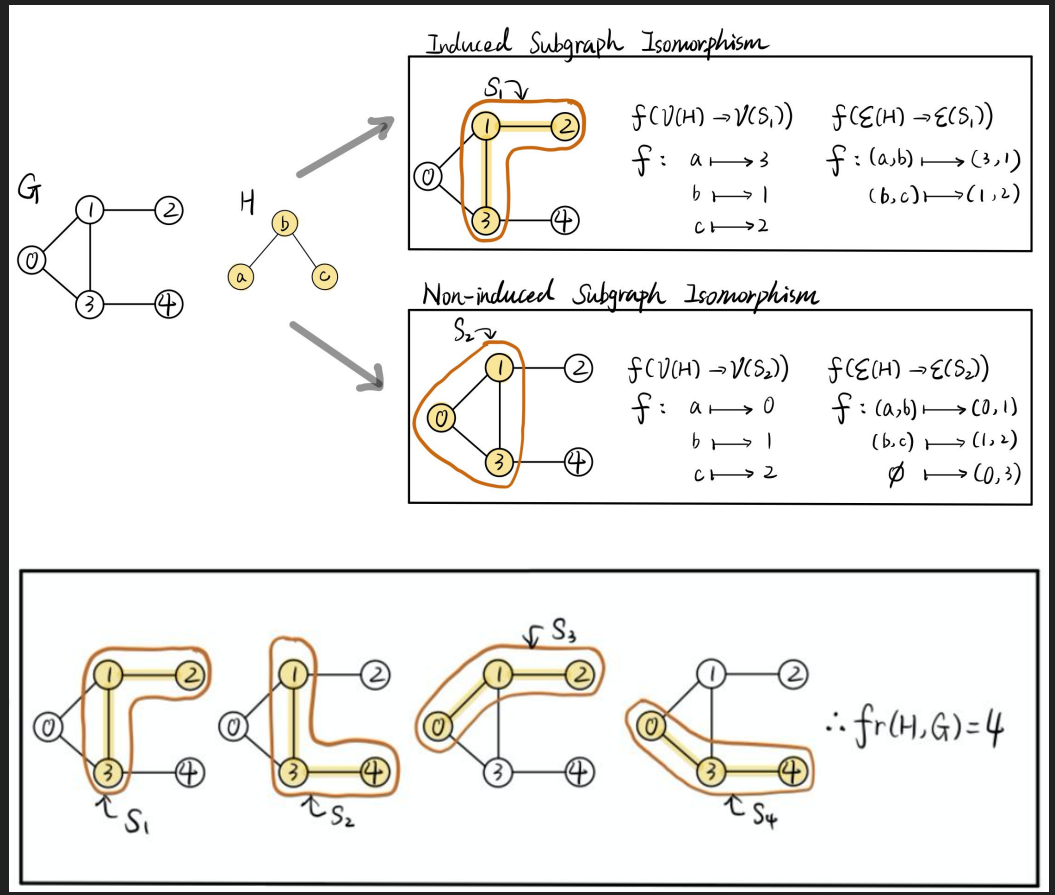
They give us power to characterize and discriminate networks



Graph Isomorphism

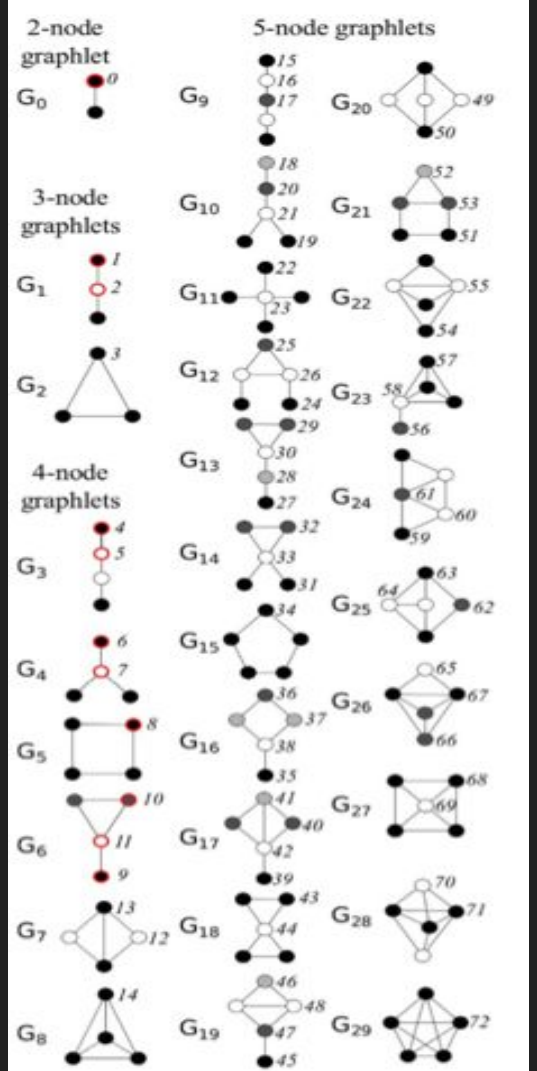


Subgraph Counting



Which subgraphs do we use to measure a huge network?

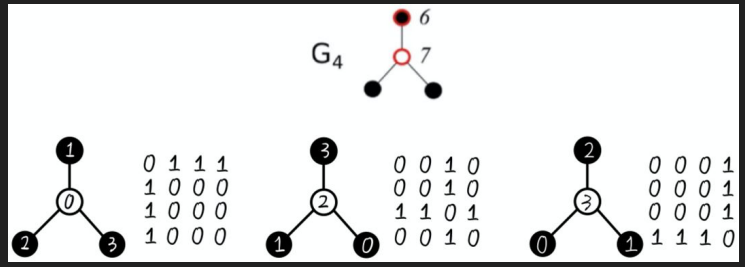
- Graphlets and Orbits

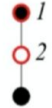
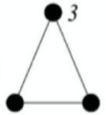
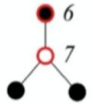
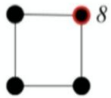
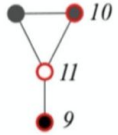
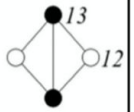
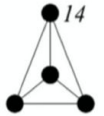


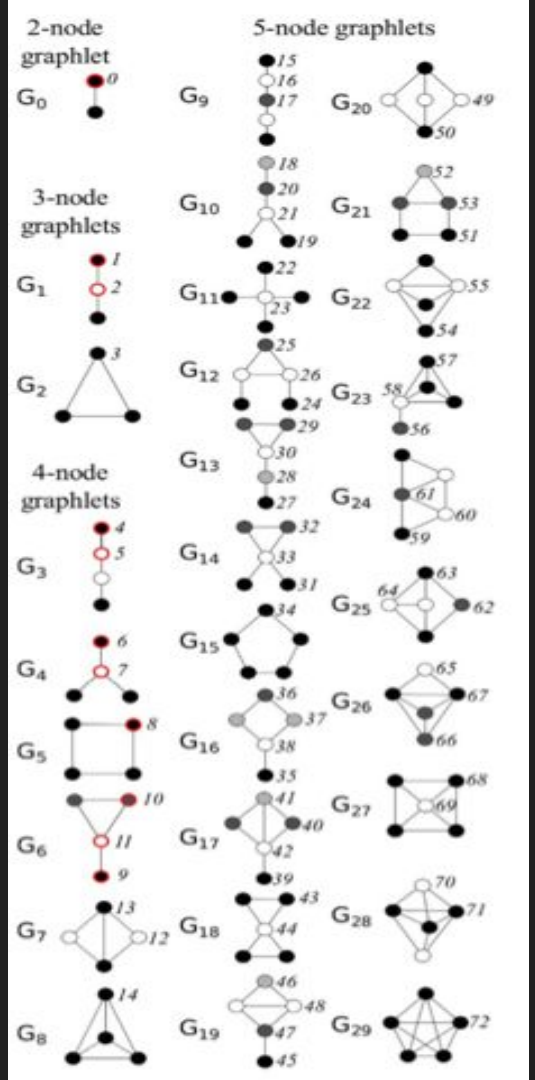
What are graphlets?

- Small induced non-isomorphic subgraphs of a large network that that differentiate nodes according to their subgraph positions
- There are 30 graphlets of up to 5 nodes (G0...G29)

Matrix ID for each Graphlet



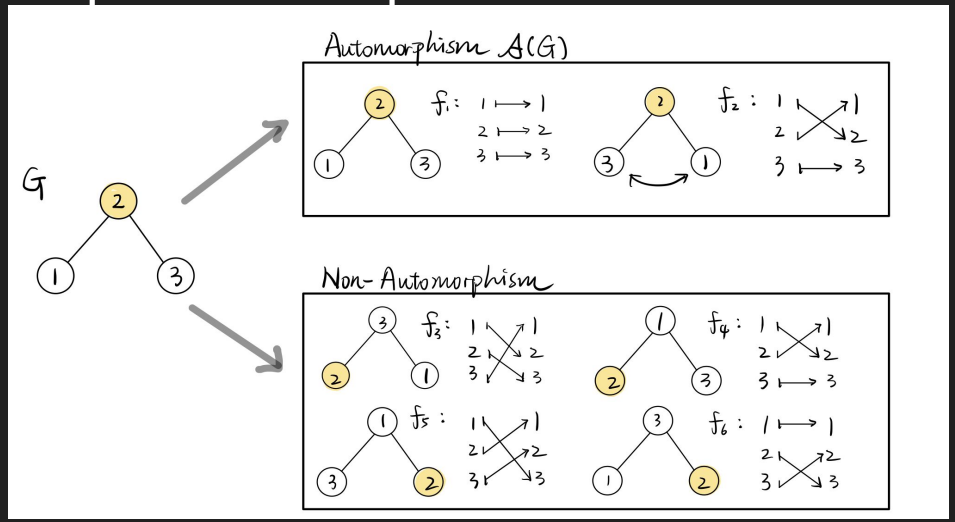
Graphlets	G_0 	G_1 	G_2 			
Adjacency Matrix	0 1 1 0	0 1 1 1 0 0 1 0 0	0 1 1 1 0 1 1 1 0			
Graphlets	G_3 	G_4 	G_5 	G_6 	G_7 	G_8 
Adjacency Matrix	0 1 1 0 1 0 0 1 1 0 0 0 0 1 0 0	0 1 1 1 1 0 0 0 1 0 0 0 1 0 0 0	0 1 1 0 1 0 0 1 1 0 0 1 0 1 1 0	0 1 1 1 1 0 1 0 1 1 0 0 1 0 0 0	0 1 1 1 1 0 1 1 1 1 0 0 1 1 0 0	0 1 1 1 1 0 1 1 1 1 0 1 1 1 1 0



What are automorphism orbits?

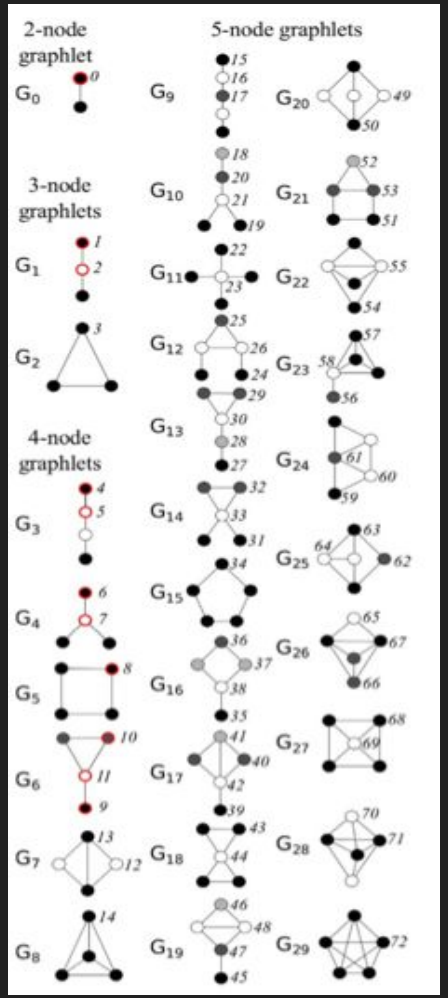
- Symmetry groups of nodes within graphlets
- A bijection of nodes that preserves node adjacency
- There are 73 automorphism orbits of graphlets of up to 5 nodes (red circles)

Graph Automorphism

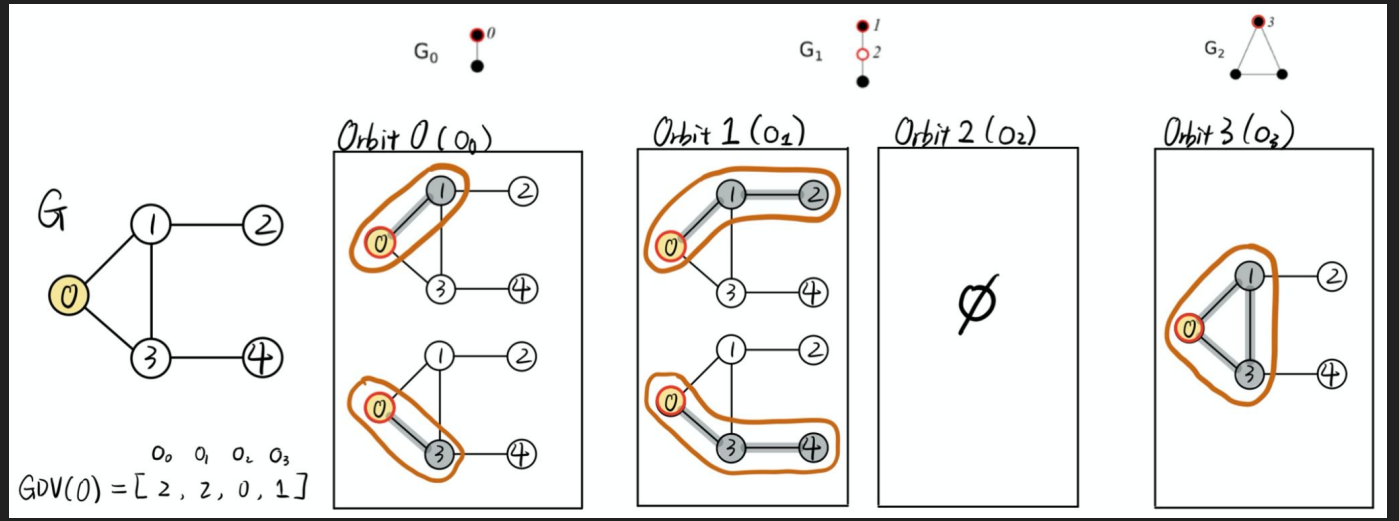


Quantifying networks’ structural information

- Graphlets Degree Distribution (GDD)



Example of GDD calculation



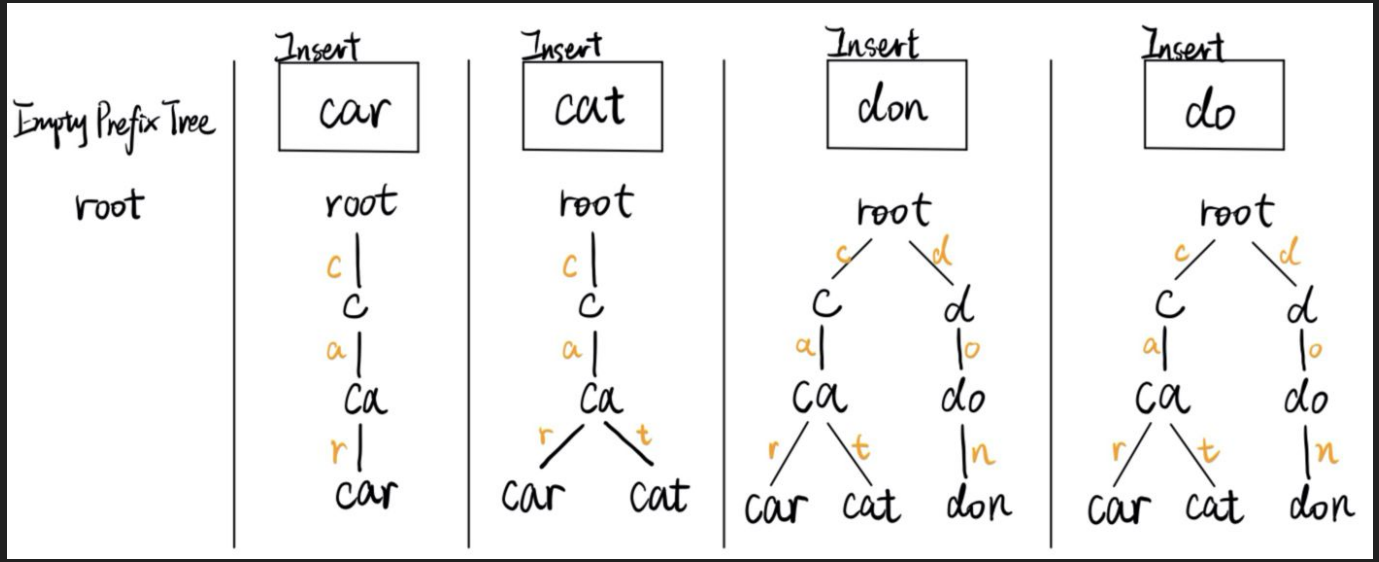
$$Fr_G = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} O_0 & O_1 & O_2 & O_3 \end{array} \\ \begin{bmatrix} 2 & 2 & 0 & 1 \\ 3 & 2 & 1 & 1 \\ 3 & 2 & 1 & 1 \\ 1 & 2 & 0 & 0 \\ 1 & 2 & 0 & 0 \end{bmatrix} \end{array} \quad \Rightarrow \quad GDD_G = \begin{array}{c} \\ O_0 \\ O_1 \\ O_2 \\ O_3 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 2 & 1 & 2 \\ 0 & 5 & 0 \\ 2 & 0 & 0 \\ 3 & 0 & 0 \end{bmatrix} \end{array}$$

How do we count graphlets efficiently?

Trie (Prefix Tree)

A tree data structure used for locating specific keys within a set.

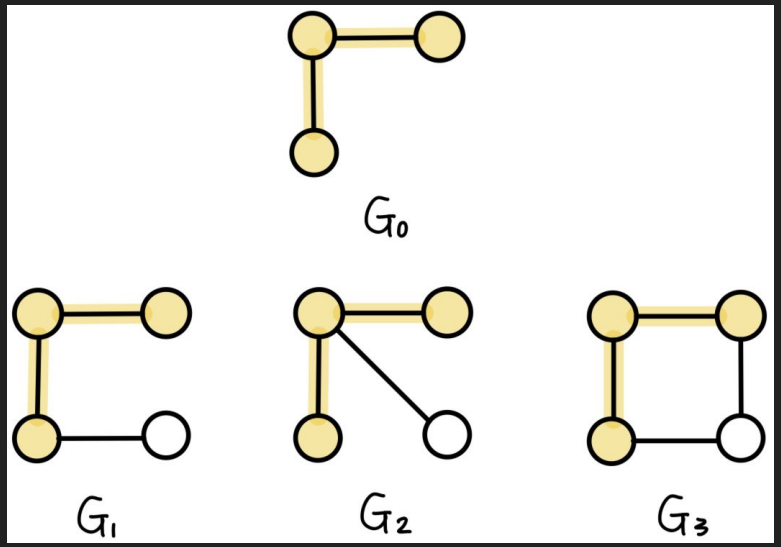
E.g. Looking for a set of substrings in a given long string



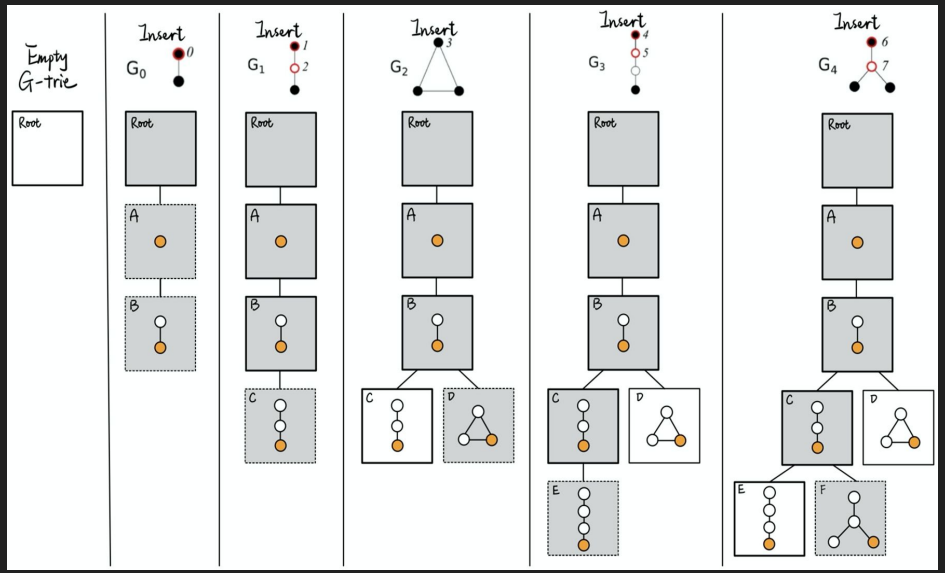
G-Trie (Prefix Tree for Graphs)

ca

cat car



G-Tries - A data-structure to store and enumerate subgraphs



Algorithm 1 Populate a g-trie T with subgraphs $G \in GSet$

```
1:  procedure CreateGTrie(GSet)
2:      T ← EmptyGTrie
3:      for all G ∈ GSet do
4:          Insert(T.root, G, 0)
5:      return T
6:  procedure Insert(V, G, depth)
7:      if k < numberRows(G.MatrixID) then
8:          for all C ∈ V.children do
9:              if ShareCommonTopology(C, G, depth) then
10:                 Insert(C, G, depth + 1)
11:                 return
12:         NewChild ← N.addChild(G, depth)
13:         Insert(NewChild, G, depth + 1)
```

Starts by looking for occurrences of the smaller common subgraph to save searching time.

Graphlets-tries - g-tries for graphlets

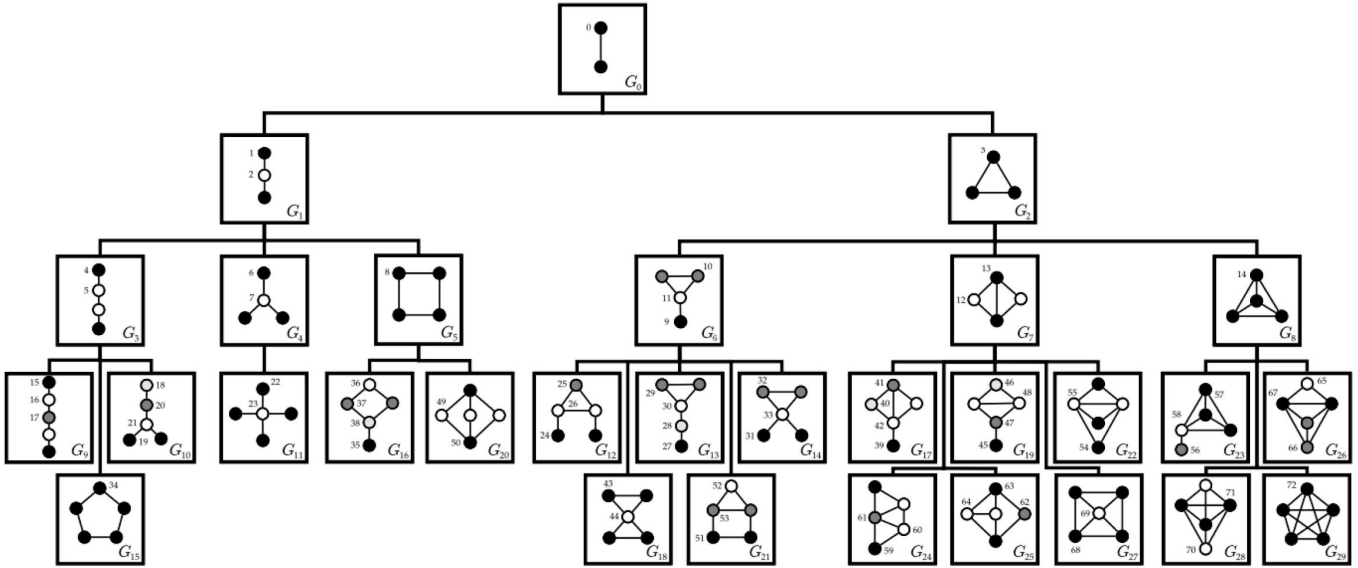
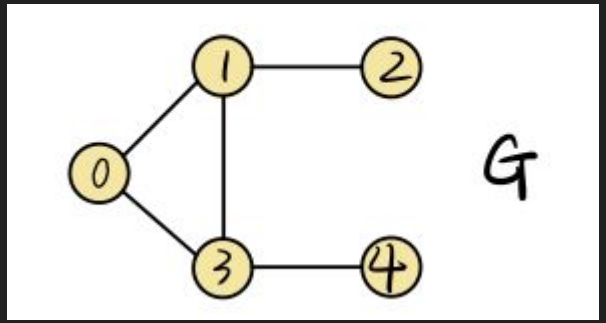


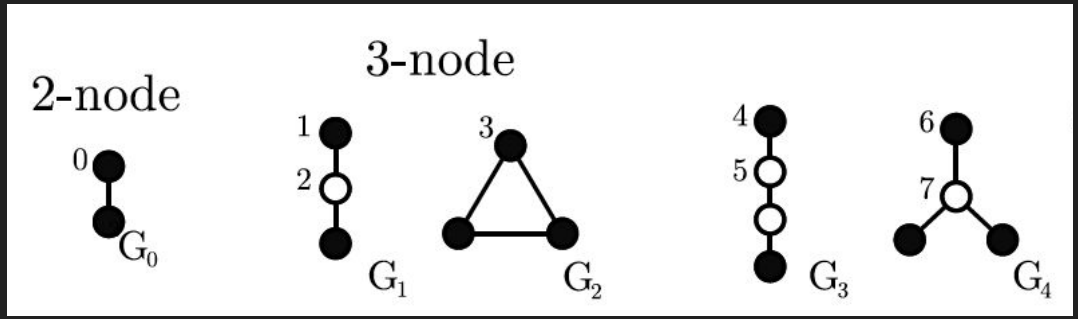
Figure 3.2: A graphlet-trie containing all 2, 3, 4 and 5-node undirected graphlets $G_0, ..., G_{29}$ as they were presented in [8]. In a g-trie the common topologies between graph(let)s of different sizes become evident.

Graphlet Census using Graphlet-trie (GT-Scanner)

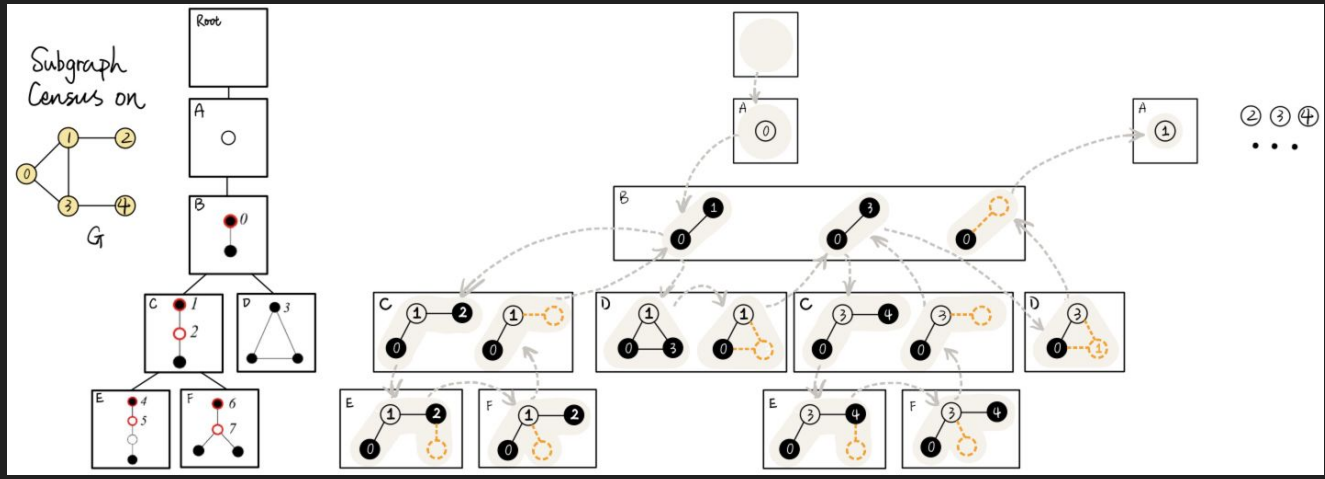
Let’s count the frequencies of the first 5 graphlets in graph G.

We will use graphlet-trie as the data structure to perform frequencies counting...



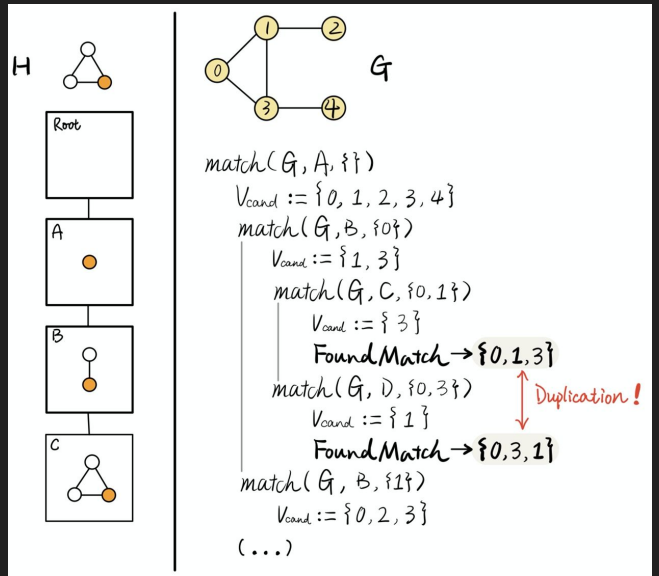


Graphlet Census using Graphlet-trie (GT-Scanner)



GT-Scanner's Symmetry Breaking Condition:

Vertices that appear later in the same orbit are only valid if they have a bigger index than the previous vertices of the same orbit.



Algorithm 2 Graphlets Census using g-trie T in network G

```
1:  procedure CENSUS(G, T)
2:      for all children c of T.root do
3:          MATCH(G, c, ∅)
4:  procedure MATCH(G, T, V_used)
5:      V_cand ← GETVALIDCANDIDATES(G, T, V_used)
6:      for all vertex v ∈ V_cand do
7:          if T.FOUNDMATCH(V_used ∪ {v}) then
8:              output (V_used ∪ {v})
9:          for all children c of T do
10:             MATCH(G, c, V_used ∪ {v})
11: function GETVALIDCANDIDATES(G, T, V_used)
12:     if V_used = ∅ then V_cand ← V(G)
13:     else
14:         V_conn ← vertices connected to the vertex being added
15:         m      ← vertex of V_conn with smallest neighborhood
16:         V_cand ← neighbors of m that respect both
17:                  connections to ancestors and
18:                  symmetry breaking condition
19:     return V_cand
```

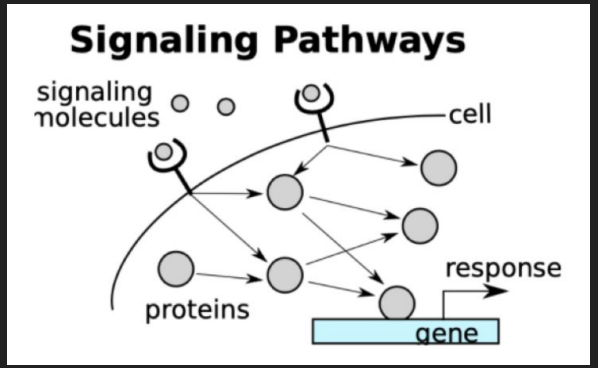
Experimental Comparisons

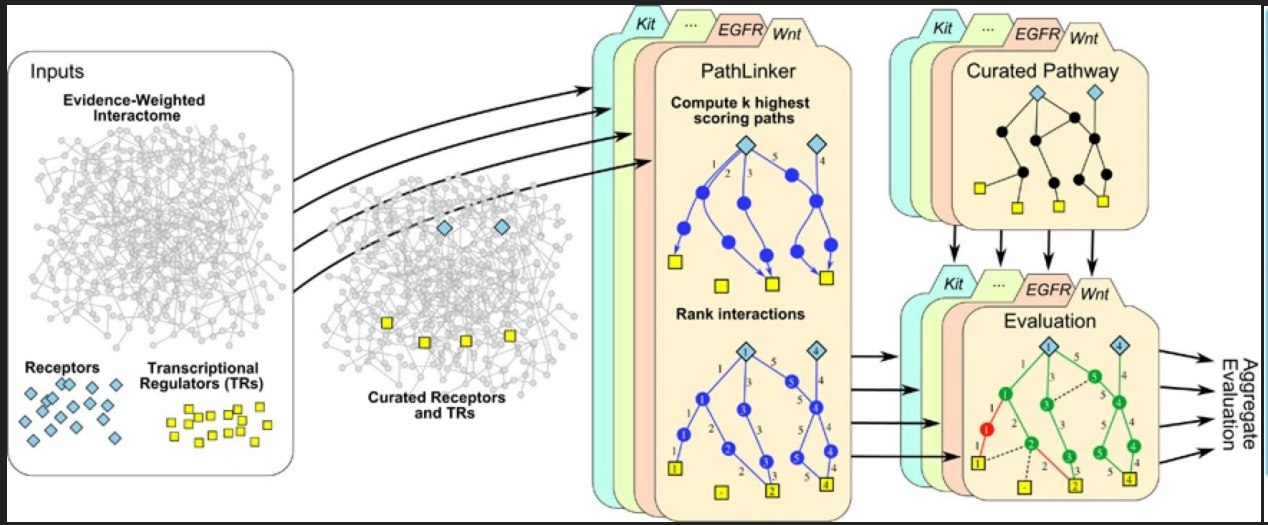
- Wnt Signaling Pathway
- Manually Curated: 106 vertices and 220 edges (gained from NetPath database)
- Reconstruct Wnt pathway from Interactome

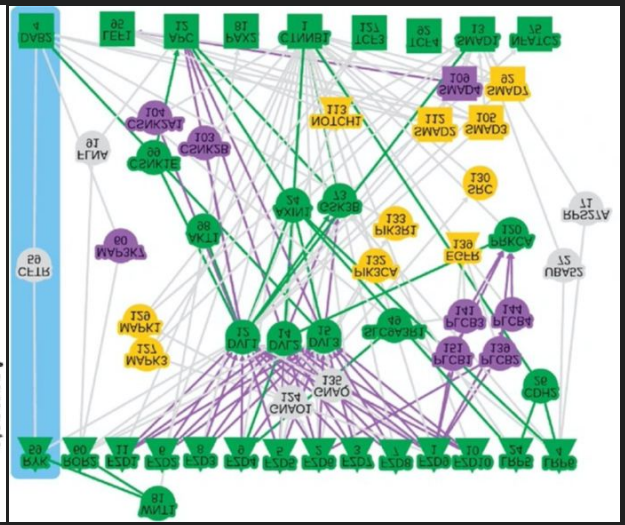
Curated vs PathLinker

Glance of PathLinker

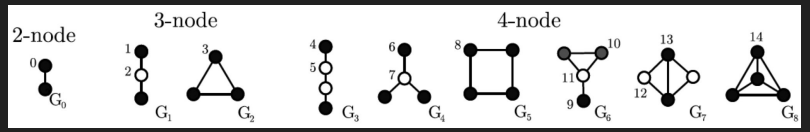
Given as input a set of sources and a set of targets for a particular curated pathway, PathLinker reconstructs the pathway by nding the k shortest paths from any source to any target within the interactome.

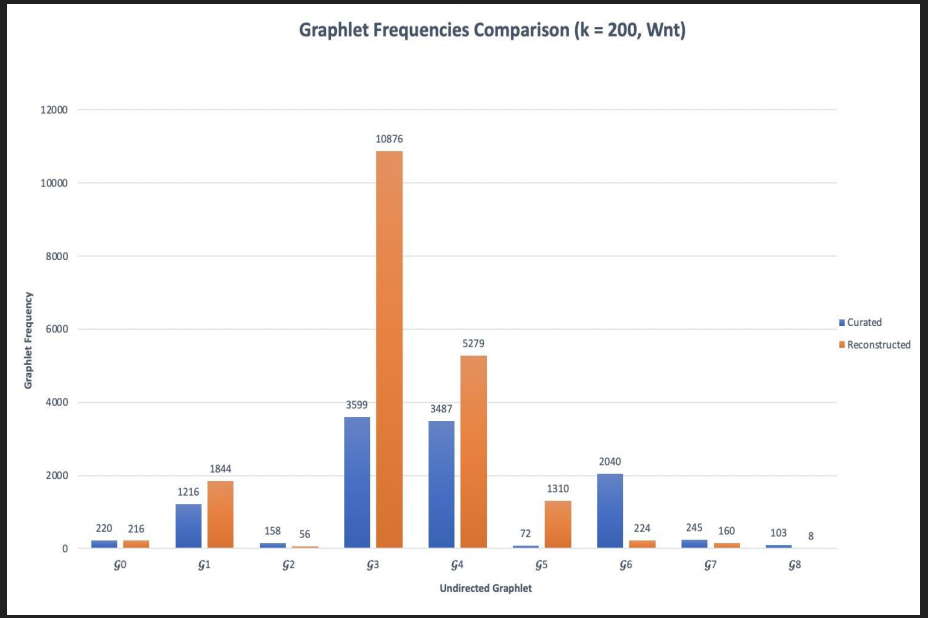


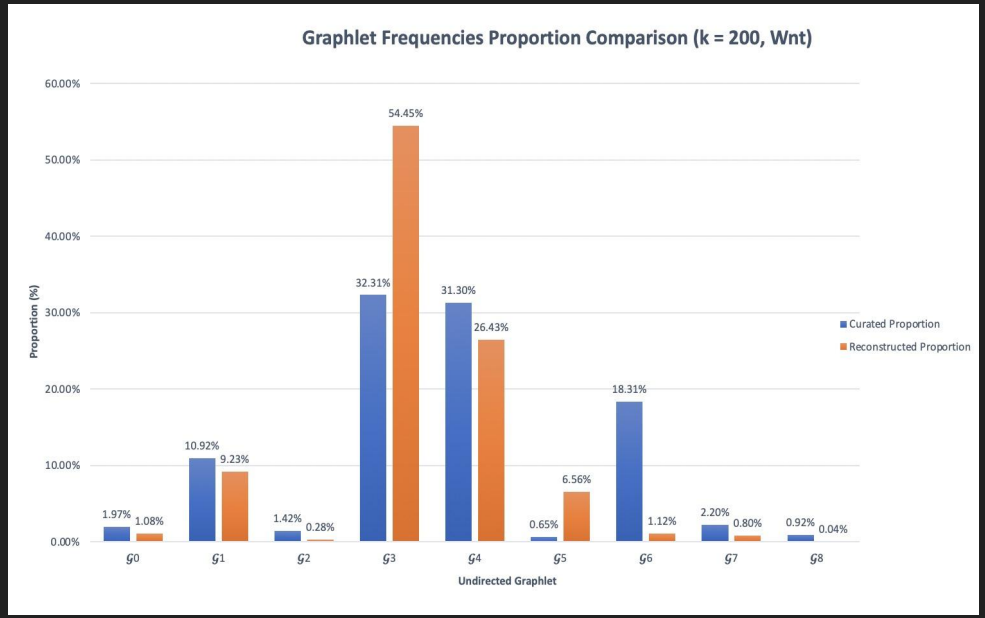




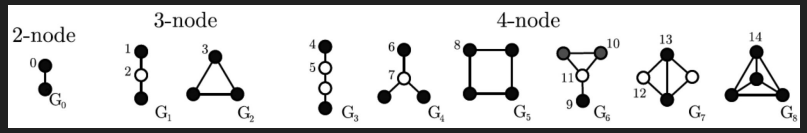
Curated Pathway vs PathLinker's Reconstruction (Wnt, k = 200)



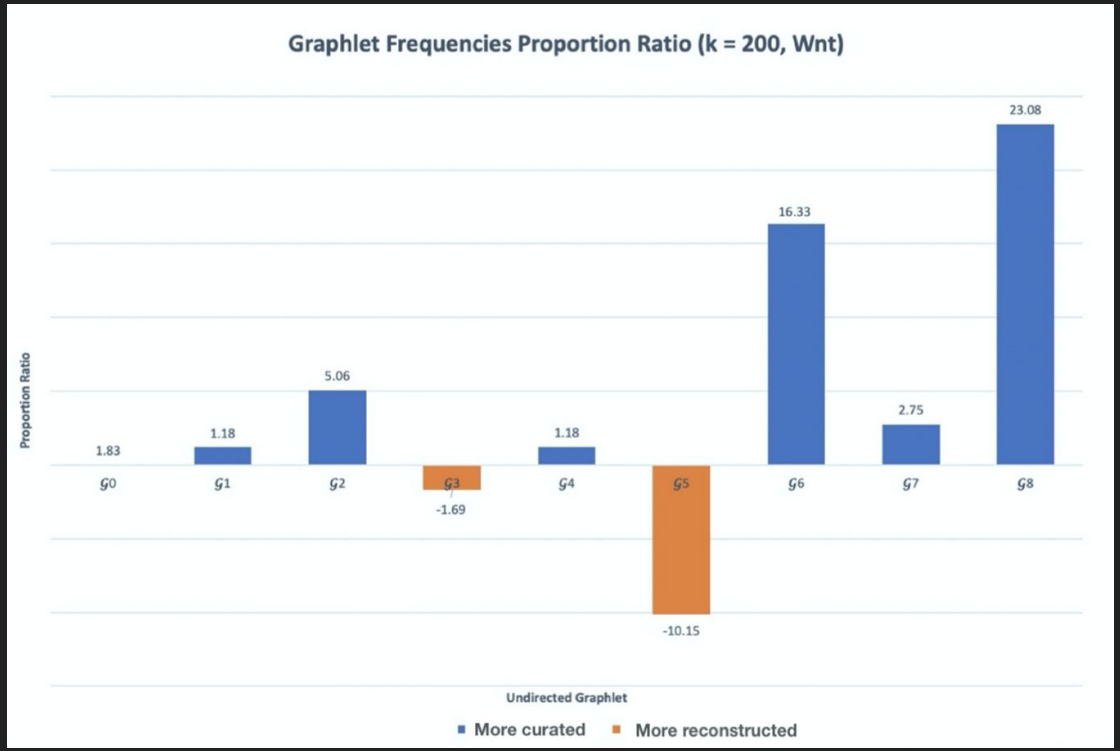




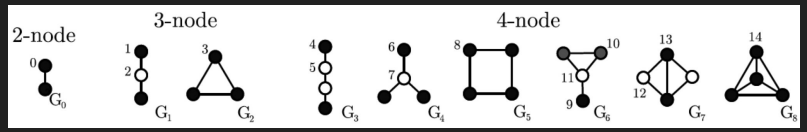
Curated Pathway vs PathLinker's Reconstruction (Wnt, k = 200)

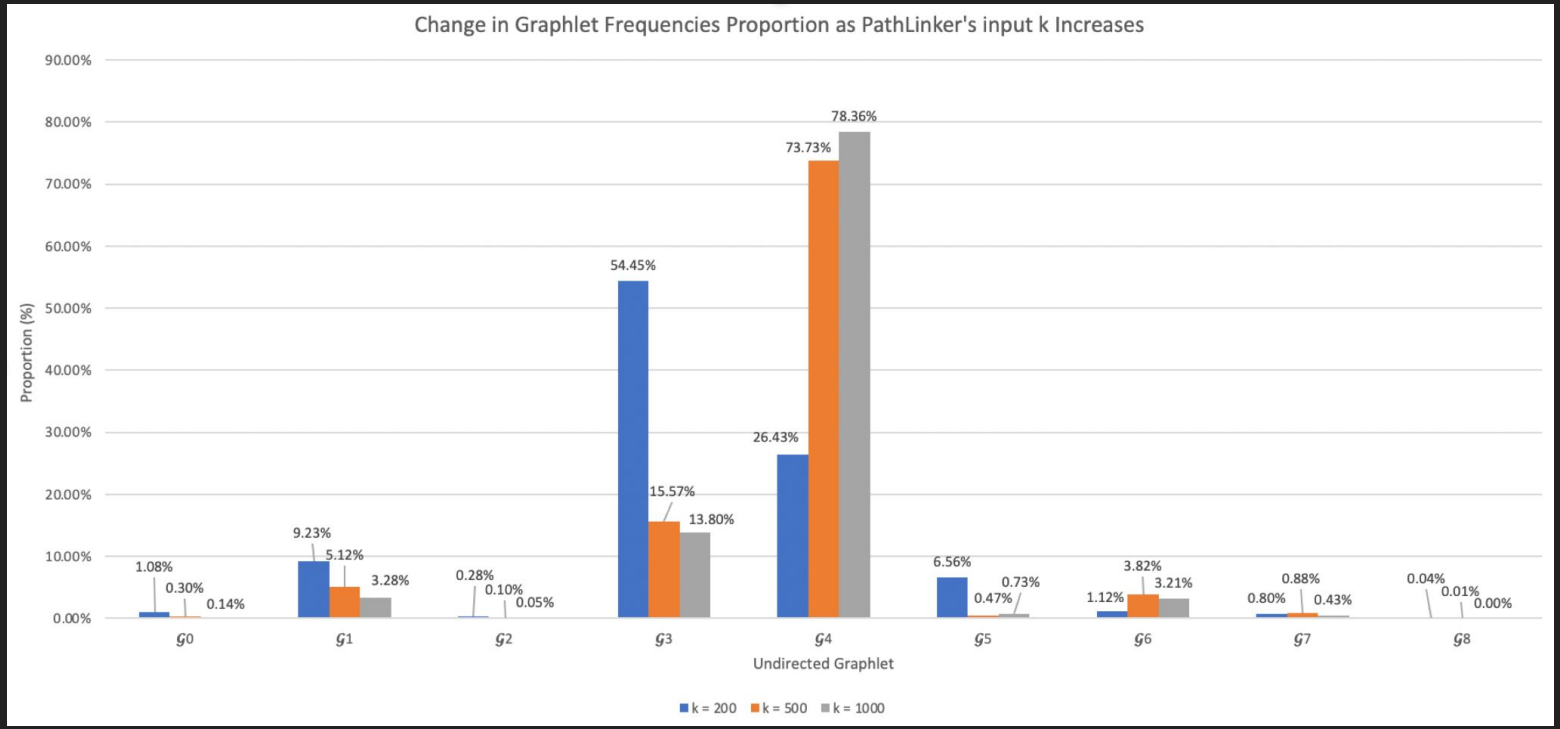


$$\frac{max(P_c(\mathcal{G}_n), P_r(\mathcal{G}_n))}{min(P_c(\mathcal{G}_n), P_r(\mathcal{G}_n))} \times sign(P_c(\mathcal{G}_n) - P_r(\mathcal{G}_n))$$



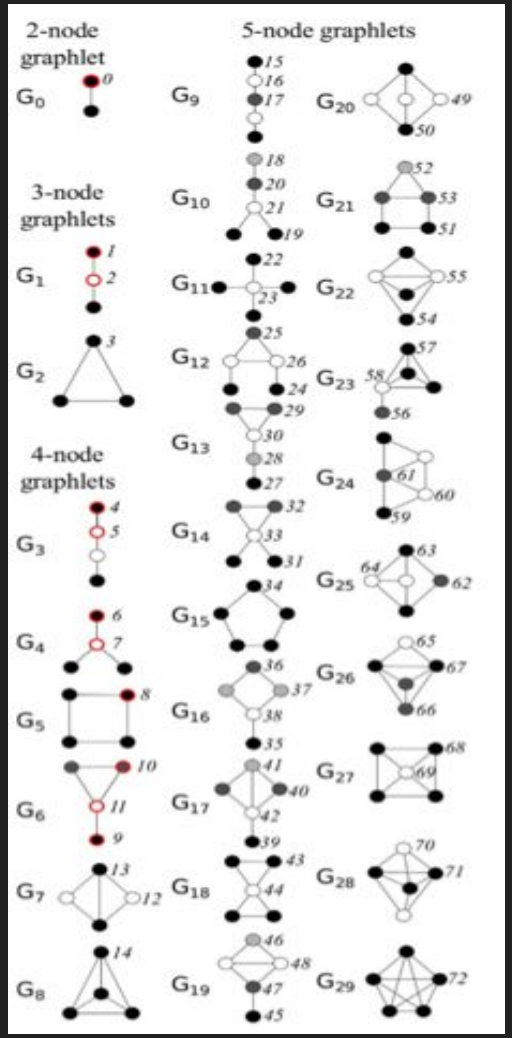
Comparisons among PathLinker's Reconstructions (Wnt, k = 200, 500, 1000)

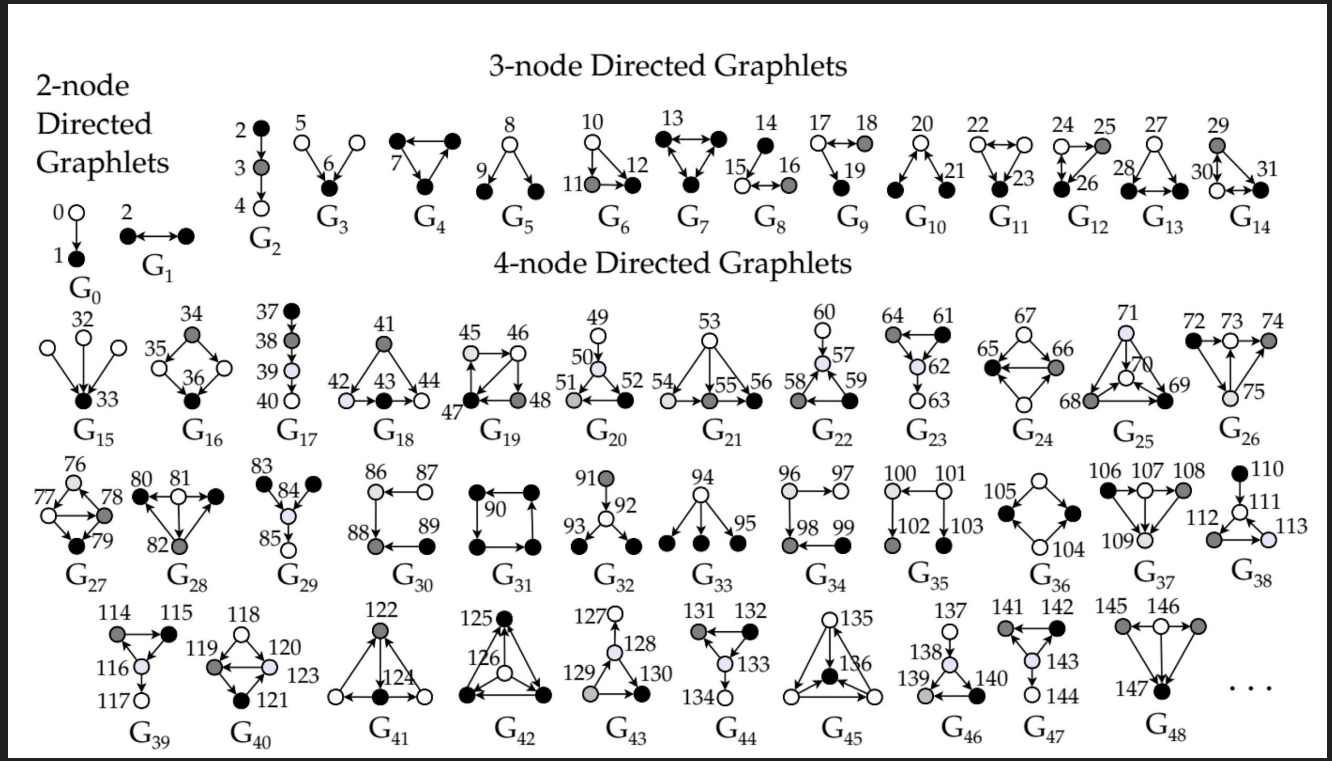




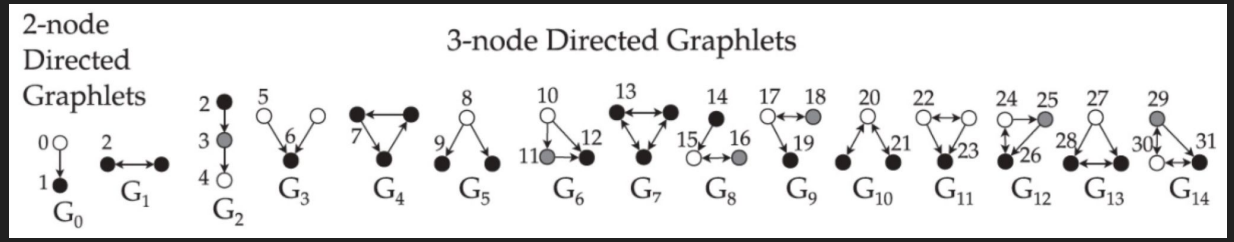
Some other fun facts about graphlets...

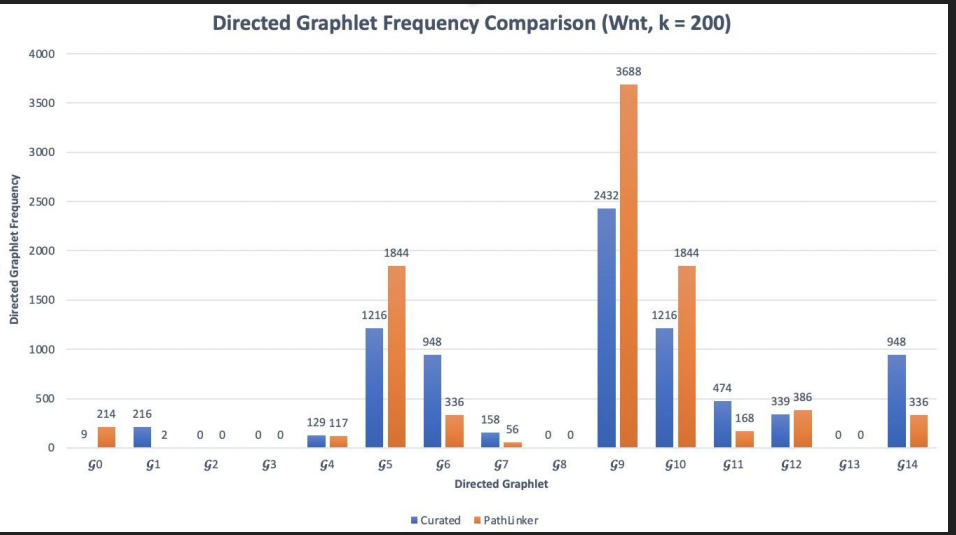
Undirected Graphlets vs Directed Graphlets

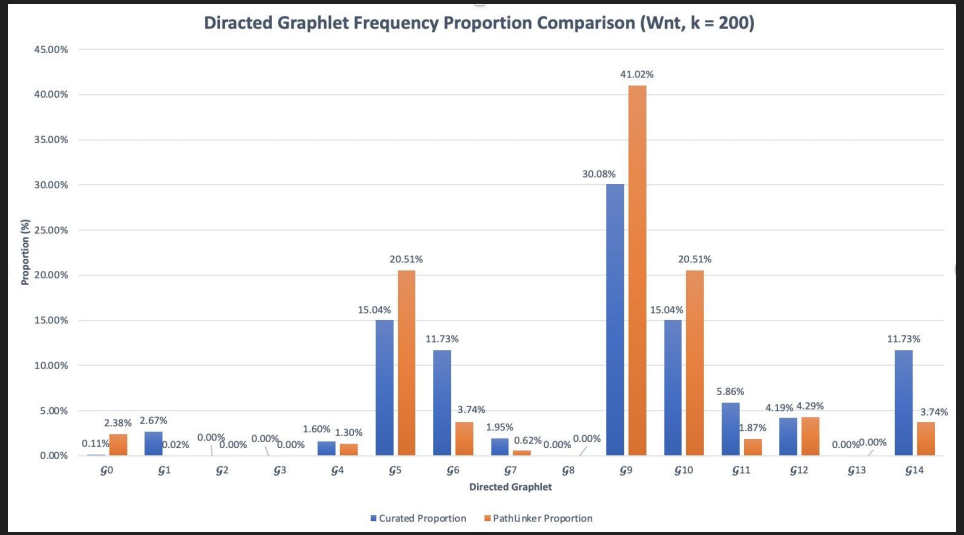




Curated vs PathLiker using Directed Graphlets (Wnt, k = 200)







Curated vs PathLiker using Directed Graphlets (Wnt, k = 200)