Social Network Analysis: Motifs

Social Network Analysis for Computer Scientists — Course Project Paper

Athanasios Agrafiotis
LIACS, Leiden University
s2029413@umail.leidenuniv.nl

1. INTRODUCTION

The paper is examines the graphs theory that is introduce the motif and clustering. The graphs represents network that show how does their connections can be used for solving problem and extract information that is important. Data from facebook , messenger and computer networks can be visualized as a graph. Motifs represent a high order properties than the usual edge that connect each entity of the network. A motif has already introduced as an edge between two node, as a triangle that three number of entities are connected to each formulate a small cycle and has also introduced as a clique.

The graph theory is separated as a directed graph that each connection represents a directed connection of one way. In contrast the undirected graph that each connection does lead to both sides of connections. The weighted graph is a representation of the graph with connection that has a value. Weighted graphs have been used as properties of the network that usually the value is the counting number that each entity had common with the other entity. In a reall world problem the weigthed graphs are a social network that each user(entity) has exchange message with all the other users(entities).

The clustering is the aspect that graph theory have so a lot of interest in the last researches that entities are connected with each other forming a group. As first approach to visualize and examine cluster was introduced the k-means that an algorithm separate the graph into groups on the average distance between the central nodes. Local graph clustering is the idea of finding a cluster without actually explore the whole graph.

This paper has as a main scope to continue analyzing data of the network such as counting triangle. The triangles are an important characteristic of a cluster that has been measured with the conductance. Conductance is an evaluation of the cluster that smaller value results to an more important cluster. The conductance also is highly correlated with the entity, such entities (nodes) does not lead to no dangling (dead end) and their structural roles in the network is centers of star, members of cliques and peripheral nodes. Roles are based on the similarity of ties between subset nodes.

This project have concentrated on clustering problem , the algorithms that can solve such issues and the time consuming required to solve such a problem. Overall the motifs have been used as the factor that can estimate the cluster importance in the graph. The most important phases is that the procedure starts from a random entity the surrounding entities are evaluated with the selected. At last the graph has as outcome to discover a community of the graph that it is important, because of the community connections with the whole graph.

The rest of the paper is organized as follows. In Section **??**. In Section **??** introduces some previous work and how it is related to our work. In Section **??** how algorithms work and how we will use them for our experiment and which are the factors that we need to consider. In Section is our datasets, In Section **??** the evaluation of our experiment and finally in Section **??** conclusion and future work. Cited papers are referenced in the Section 8.

2. RELATED WORK

The motif have solved with different research methods so far. Chiba and Nishizeki [2] paper explains that is a simple algorithm for triangle enumeration. The algorithm computes the intersection of adjacent nodes. The algorithm is a heuristic search that start at random from a node with high degree and continue visiting the connected nodes counting the number of triangles.

Yin [4] propose the MAPPR a local motif algorithm that is an imporvement of Aproxmated Personalized Pagerank method. The way that MAPPR works is that counts the number of motifs in a whole graph and next it generates a new graph. The new graph contains edges that are weighted based on the motif enumeration and improve motif conductance.

Shang [3] creates communities picking a seed node and move to neighborhood nodes base on the motif degree and measure the importance of the community using an extension of the modularity function.

The most of the researches are consider motifs as important properties and creates local clusters around them. For the measure that have been used is the conductance and modularity.

In recent approach the local clustering is introduced directly from motifs and no more with edges, the paper [1] propose an algorithm that can computer the clusters six times faster and three times better than the state of the art for the tri-

This paper is the result of a student course project, and is based on methods and techniques suggested in [**?**, **?**, **?**, **?**].
SNACS '17 Social Network Analysis for Computer Scientists, Master CS, Leiden University (`liacs.leidenuniv.nl/~takesfw/SNACS`).

angle motif. The overall cluster start from a random node and evaluate the cluster using a Hyper graph model H_μ that can calculate the motif conductance from it.
modularity starts with the assumption that each node represents a community and in each iteration the neighbors of these node is added to the community by recomputing modularity. A_{ij} are the weights between node i and node j. $k_i \sum_j, A_{i,j}$ is the sum of weights of the edges attached to vertex i. The community c_i represents one community with node i, the function δ is 1 the weights are equal otherwise is 0.

$$Q = \frac{1}{2m}\sum_{ij}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j) \tag{1}$$

3. NOTATION

We consider a graph $G = V_{(1..n-1)}, E$ that V are the number of nodes and E are the number of edges. The total number number of node $n = |V|$ and total number of edges $m = |E|$. Each node has a number of nodes that are connected to each other with edges $N(u) = u : v, u \in E$ that is the notion of neighbor and a node u is connected with node v with an edge. The G' is a subgraph of the original graph so that $G' = (V', E')$ and $V' \subset V$ and the edges is $E' \subset E$. The edges are of subgraph is presented as $E' = E \cap (V'xV')G'$ induced in G' by V' and $\bar{V'} = \frac{V}{V'}$ be the complement of a set $V' \subset V$. Let a motif μ in a graph G consists of building all occurences of μ as subgraph of G. The number of edges of a node u are notated as $d(u)$ and the diameter as Δ the weighted degree as $d_w(u)$ and the weighted diameter as Δ_w. The number of motifs are $d_\mu(u)$ and $d(u) = \sum_{u \in V'} d(u)$ and $d_w(u) = \sum_{u \in V'} d_w(u)$ and $d_\mu(u) = \sum_{u \in V'} d_\mu(u)$.
The local graph clustering of a graph $G = (V, E)$ that a node $u \in V$ are taken as input and belong to a community $C \in V$. The quality of the cluster is measured with modularity or conductance. The conductance measures $\phi(C) = \frac{|E'|}{min(d(C), C(\hat{C}))}$ where $E' = E \cap (Cx\hat{C})$ the set of edges that have the cluster C. For the motif enumeration is used $\phi(C) = \frac{|M'|}{min(d(C), C(\hat{C}))}$ where M' all motifs that contained one node in C and one node in $\hat{C}$ if a motif is an edges $\phi(C) = \phi_\mu(C)$.

4. PAGERANK

The original problem is that we rank nodes based on their properties. Page rank has been used initially used as a link analysis algorithm that assigns a numberical weighting to each element of a hyperlinked set of documents. The first step is to structure forms a huge graph $G(V, E)$ that each of the nodes have edges conneted to each other. The page rank algorithm has as purpose to rank the nodes with the most number of connections or any other property higher than the other nodes. The assumption is that more important nodes are likely to receive more links from other nodes. As already have mentioned edges can be undirected and bidirected known as indegree,outdegree. Since there is a graph it is possible to calculate the pagerank of a given node u associated with every node. The formulation of pagerank is

$$PR(u) = (1-d) + d(\frac{PR(T_1)}{C(T_1)} + \frac{PR(T_2)}{C(T_2)} + \frac{PR(T_n)}{C(T_n)}) \tag{2}$$

The pagerank of a node u is equal to the dampling factor that it's value is between (0,1) summed with dampling factor again multiplied with all other nodes $(T_1, T_2, .., T_n)$ probability $PR(T_{1..n})$ that links to node u. Divided with number of outdegree edges of a given node T_i formulated as $C(T_i)$. At the begining every single node has equal probability pagerank value to be visited $\frac{1}{n}$. In each iteration the value converges and formulated as:

$$PR_{t+1}(P_i) = \sum_{P_j}\frac{PR_t(P_j)}{C(P_j)} \tag{3}$$

The pagerank intializes the values of each node at the first iteration as $1n$ and in the next iteration sum the values of the current value $PR(u) = \frac{\frac{1}{n}}{outdegreeedges} + \frac{\frac{1}{n}}{outdegreeedges}$ the terms is the neighbor node (v, d), so next it calculates the pagerank of node v that is the next node with the same way. The sum of pageranks values must be equal to one. In the second iteration it makes use of the nodes pagerank value that point to u divided by their total outdegree number. No matter the number of edges the pagerank will give higher rank to the nodes that are be pointed from nodes with high centrality.
The pagerank can be computed using a matrix representation of the probability that each node has to be conndected with all nodes as:

$$v = \begin{bmatrix} \frac{1}{n} \\ \frac{1}{n} \\ \frac{1}{n} \\ \frac{1}{n} \end{bmatrix} \tag{4}$$

$$v_n = Hv_n = H(Hv) = H^n v \tag{5}$$

after the a number of iterations converges to a state. The H is the transition matrix and v is the final page rank values. The pagerank is defined by the probability that a random walk starts on a node and follow the edges to visit nodes. One of the issues using pagerank are the nodes that do not have outgoing edges. And independent cluster nodes that are subgraphs unconnected to each other. So the last equation is the pagerank values that is equal to the damping factor that is usually to 0.15 multiplied with the transition matrix and in the second term the multiplication of damping factor with the and Identity equal to transition with all values equal to 1. The first term calculates the ranking with probability $1 - d$ while the second term is the probability 0.15 of jumping to another cluster or to avoid nodes without outgoing edges.

$$v = (1-d)H + dB \tag{6}$$

4.1 Approximation of Page Rank

The approximation page rank is similar to the original page rank. The algorithms starts from a node u. Next it makes use of the breadth first search to visit all the nodes in the subgraph S' around the selected node. Next it calculates the influence of the current node in the subgraph. As influence takes in consideration the number of motifs. Approximation of pagerank uses a sample that depends in the selected node u and number of l layers that the subgraph has.

the u is the node that picked for the next call, r is the set of recursive calls and p is the current approximation.

Algorithm 1 Approximate Personalize PageRank

Ensure: `ApproximatePageRank(u,a,e)`
Ensure: $p = 0$
Ensure: $r = x_u$
while $max_{u \in V} \frac{r(u)}{d(u)} e$ **do**
  `Choose any vertex u where` $\frac{r(u)}{d(u)} e$
  `Apply` $push_u$`at vertex u updating p and r.`
end while
Ensure: `Return p, which satisfies` $p = apr(a, x_u, r)$ `with` $max_{u \in V} \frac{r(u)}{d(u)} < e$
Ensure: $push_u(p, r)$:
Ensure: `Let p'=p and r'=r, except for the following changes:`
$p'(u) = p(u) + ar(u)$
$r'(u) = (1 - a)r(u)/2$
for $eachvsuchthat(u, v) \in E$ **do**
  $r'(u) = r(u) + (1 - a)r(u)/(2d(u))$
end for
Ensure: $Return(p', r')$

4.2 Motif-based Approximate Personalized PageRank (MAPPR)

It is an adaptation of Approximate Personalized PageRank that make use of the motifs. The algorithm makes use of the approxumate PPR vector on a weighted graph based on motif.

Algorithm 2 Approximate Weighted PageRank

Ensure: `Input:Undirected edge-weighted graph` $G_w = (V_w, E_w, W)$ `seed node u teleporation parameter a, tolerance e`
Ensure: `Output:an aproximate weighted Personalize Page Rank vector p`
Ensure: $p(u) \leftarrow 0forallverticesu$
Ensure: $r(u) \leftarrow 1andr(u) \leftarrow 0$ `for all vertices u except v`
Ensure: $d_w(u) \leftarrow \sum_{e \in E_w : u \in e} W(u)$
while $r(u)/d_w(u) \geq e$ `for some node u` $\in V_w$ **do**
  `/* push operation */`
  $p \leftarrow r(u) - \frac{e}{2} d_w(u) : p(u) \leftarrow p(u) + (1 - a)p; r(u) \leftarrow \frac{e}{2} d_w(u)$
  for $eachx : (u, x) \in E_w$ **do**
    $r'(x) \leftarrow r(x) + \frac{W(u,x)}{d_w(u)} * ap$
  end for
end while
Ensure: $Return(p)$

The motif-APPR constructs a weighted graph, where $W_{i,j}$ is the number of instances of M containing nodes i and j. Next computes the approximate PPR vector with weights. An at last makes use of the sweep procedure to output the minimal conductance.

5. DATASETS

6. EXPERIMENTS

				F1-scone		Precision		Recall		Motif conductance	
Network	|V|	|E|	#comms.(sizes)	edge	triangle	edge	triangle	edge	triangle	edge	triangle
COM-ORKUT											
COM-FRIENDSTER											
COM-AMAZON											
EMAIL-EU											
WIKI-CATS											

Table 1: motif based APPR (MAPPR) where motif is the triangle.

plot(Motif Conductance over iterations)
plot(runing time of algorithm over time)
plot(cluster size colorize the subgraph)
plot(motif distribution)

7. CONCLUSION

8. OUR PAPERS

Crawling Online Social Graphs [?], Sampling from Large Graphs [?] and Walking in Facebook: A Case Study of Unbiased Sampling of OSNs [?].

9. REFERENCES

[1] A. Chhabra, M. F. Faraj, and C. Schulz. Local motif clustering via (hyper)graph partitioning, 2022.

[2] N. Chiba and T. Nishizeki. Arboricity and subgraph listing algorithms. *SIAM Journal on Computing*, 14(1):210–223, 1985.

[3] R. Shang, W. Zhang, J. Zhang, J. Feng, and L. Jiao. Local community detection based on higher-order structure and edge information. *Physica A: Statistical Mechanics and its Applications*, 587:126513, 2022.

[4] H. Yin, A. R. Benson, J. Leskovec, and D. F. Gleich. Local higher-order graph clustering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD 17, page 555–564, New York, NY, USA, 2017. Association for Computing Machinery.