

ب) مسئله پیدا نمودن انجمن‌ها (Community Detection)

برای این منظور یک گراف بدون جهت (مانند گراف فیسبوک) به عنوان ورودی داده می‌شود. هدف پیدا کردن افرادی هستند که به احتمال زیاد با هم در یک گروه کاری یا دوستی قرار دارند. مسئله پیدا کردن بهترین گروه‌ها بر اساس یک معیار یک مسئله NP-Complete است. برای این منظور یک الگوریتم تقریبی پیدا نمودن انجمن‌ها انتخاب شده است که باید پیاده‌سازی شود. این الگوریتم به نام الگوریتم لیوین [۱] معروف است. این الگوریتم را به صورت کامل پیاده‌سازی کنید. این الگوریتم دو فاز دارد و در هر فاز چند تکرار داریم. در ابتدا هر نود را در یک گروه جداگانه قرار می‌دهد و سپس بر اساس یک معیار تصمیم می‌گیرد که گروه یک نود را به گروه یکی از نوه‌ای مجاور آن نود تغییر دهد. خلاصه‌ای از متن الگوریتم در زیر آمده است ولی برای پیاده‌سازی بهتر است مقاله اصلی را بخوانید.

- “Fast unfolding of communities in large networks” (2008)
- Vincent Blondel, Jean-Loup Guillaume, Renaud Lambiotte, Etienne Lefebvre
- Optimize modularity: $[-1, 1]$, the density of links inside communities compared to links between communities

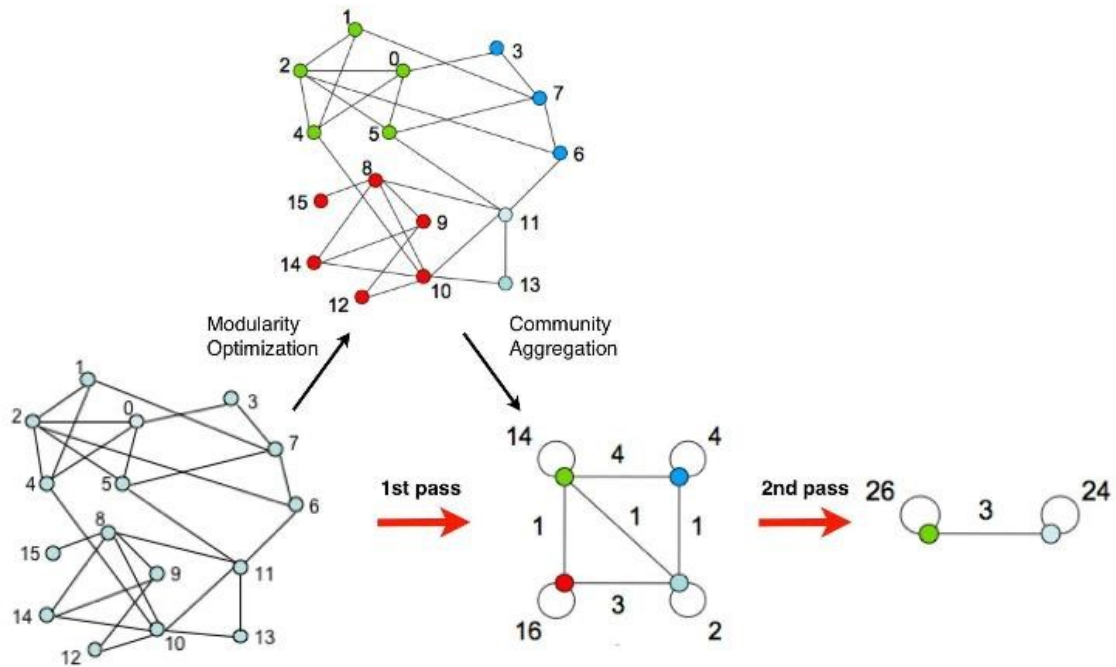
$$M = \frac{1}{2W} \sum_{ij} \left[w_{ij} - \frac{k_i k_j}{2W} \right] \delta(c_i, c_j)$$

- w_{ij} is the edge weight between nodes i and j
- k_i is the sum of the weights of links that connect to node i
- W is the sum of the weights of all links in the network
- c_i is the community of node i
- δ is a simple delta function (1 if $c_i == c_j$ else 0)

- Start with a weighted network of N nodes, initially assigning each node to a different community
- For each node i:
 - Evaluate change in modularity if we placed node i in the community of each of its neighbors
 - $$\Delta M = \left[\frac{\Sigma_{in} + 2k_{i,in}}{2W} - \left(\frac{\Sigma_{tot} + k_i}{2W} \right)^2 \right] - \left[\frac{\Sigma_{in}}{2W} - \left(\frac{\Sigma_{tot}}{2W} \right)^2 - \left(\frac{k_i}{2W} \right)^2 \right]$$
 - Σ_{in} is the sum of the weights of links inside community C
 - Σ_{tot} is the sum of the weights of links to all nodes in C
 - k_i is the sum of the weights of links that connect to node i
 - $k_{i,in}$ is the sum of the weights of links from node i to nodes in C
 - W is the sum of the weights of all links in the network
 - Move node i to the community with biggest modularity gain
 - If no positive gain is found, stay in original community
- Repeat until no further improvement can be achieved

Louvain Step 2: Community Aggregation

- Construct a new network whose nodes are the communities found in step 1
- Weight of a link between two nodes is the sum of the weights of all the links between the nodes in the corresponding communities
- Links between nodes of the same community lead to weighted self-loops
- Once step 2 is completed, a single pass of the algorithm is completed
- Repeat steps 1 and 2, decreasing the number of communities with each pass, until maximum modularity is achieved



[1] Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment. 2008 Oct 9;2008(10):P10008.