Professor QUAN Thanh Tho
12 January 2020

MINF 2019 - Bordeaux Master Computer Science

Social Network community detection algorithm design report

Students:
NGUYEN Vu Anh Trung
VO Thi My Hanh
PHAM Duc Tho

Section 1: Experimental Results

Traditional Algorithm: Hierarchical clustering

Space: it requires $O(N^2)$ space for storing the distance matrix.

Time: $O(N^2Log(N))$ in most cases.

Disadvantages : HC is computationally expensive $O(N^2Log(N))$

hence is not recommended on huge datasets whereas k-means using linear time.

==========

Heuristic Algorithm : Walk-traps (Ants Random Walk)

Params of input Networks:

+N: number of vertices

+M: number of edges

WORST CASE: **Space:** $O(N^2)$ **Time:** $O(M \times N^2)$.

AVERAGE CASE: **Space:** $O(N^2)$ **Time:** $O(N^2Log(N))$ in real-life data cases.

Disadvantages : Random Walk give a satisfactory answer but not and absolutely correct all communities presents in the network.

Methods	Graph Size	ExecutionTime in seconds	Communities found
Conventional method: Hierarchical clustering	Nodes: 1000 Communities: 39	82 seconds	39 communities
Heuristic Method: Random walks	**Nodes: 1000 Edges: 168093 Average degree: 336**	**3 seconds**	**32 communities**
	Nodes: 5000 Edges: 82357 Average degree: 291	**Around 50 seconds**	**125 communities**

Section 2: The heuristic approach: Random walk algorithm

That is the idea behind the walktrap algorithm, introduced by Pascal Pons and Matthieu Latapy in 2005. It also inspired the infomap algorithm created in 2008 by Martin Rosvall.

In order to group the vertices into communities,
we will now introduce a distance r between the vertices that captures the community structure of the graph. This distance must be large if the two vertices are in different communities, and on the contrary if they are in the same community it must be small.
It will be computed from the information given by random walks in the graph.

Distance representation :
heap structure has been implemented to store the distances between communities.

Computation of the distance r
Once the two vectors $\mathbf{P_i^t}$ and $\mathbf{P_j^t}$ are computed, the distance $\mathbf{r_{ij}}$ can be computed in time O(n) using Equation (1).

Definition 1 *Let i and j be two vertices in the graph and*

$$r_{ij} = \sqrt{\sum_{k=1}^{n} \frac{(P_{ik}^t - P_{jk}^t)^2}{d(k)}} = \left\| D^{-\frac{1}{2}} P_{i\bullet}^t - D^{-\frac{1}{2}} P_{j\bullet}^t \right\| \qquad (1)$$

where $\|.\|$ is the Euclidean norm of $\mathbb{R}^n$.

Notice that given the probability vectors $\mathbf{P^t}$ and $\mathbf{P^t}$,
the distance $r_{C_1C_2}$ is also computed in time O(n)
The probability vectors can be computed once and stored in memory
(which uses $\mathbf{O(n^2)}$ memory space) or they can be dynamically computed
(which increases the time complexity) depending on the amount of available memory.
We propose an exact method and an approximated method to compute them.

Example Running guide:

Run program

./walktrap net/graph.net -o result1.txt -m2000
Argument:
OUTPUT RESULTS: -o output_file_name
MEMORY_LIMIT: -m2000

Reference :

Computing communities in large networks

using random walks

Pascal Pons and Matthieu Latapy

LIAFA – CNRS and University Paris 7 – Jussieu

Walk Trap project of LIP6 institute

https://www-complexnetworks.lip6.fr/~latapy/PP/walktrap.html

==========

Community detection in networks

using self-avoiding random walk

Guilherme de Guzzi Bagnato, José Ricardo Furlan Ronqui, Gonzalo Travieso

(Submitted on 28 Jul 2016 (v1), last revised 21 Jan 2018 (this version, v4))

https://arxiv.org/abs/1607.08597

==========

Multiple Local Community Detection

Alexandre Hollocou, Thomas Bonald, Marc Lelarge

HAL Id: hal-01625444

https://hal.archives-ouvertes.fr/hal-01625444