Finding Communities by Clustering a Graph into Overlapping Subgraphs

Jeffrey Baumes, Mark Goldberg, Mukkai Krishnamoorthy, Malik Magdon-Ismail, Nathan Preston

Rensselaer Polytechnic Institute, Troy, NY, USA
Outline

Introduction

Algorithms IS and RaRe

Experiments

Conclusions and future work
Outline

Introduction

Algorithms IS and RaRe

Experiments

Conclusions and future work
What is a cluster?

A *cluster* is a set of closely related objects, not as closely related to the rest

*Clustering* is the development of a collection of clusters

Traditionally, clustering is restricted to *partitioning* into non-overlapping clusters
Overlapping clusters

Partitioning is well researched; many algorithms and software packages exist (Kernighan-Lin, $k$-means, hierarchical, CHACO)

General clustering allows overlapping clusters
Such clustering is natural in social networks

Clustering in a general sense is not well-studied
A new definition of a cluster

Define a weight function (or density) \( W(C) \) for every subset of objects, then maximize \( W \) locally

A cluster is defined as a set of objects whose weight is larger than any set close to it; the cluster is said to be locally optimal

A set is "close" to another if it may be derived from the other by adding or removing one object
Weighting functions

\[ p_{\text{ex}} = \frac{\text{III}}{\text{III} \times \text{IIIIIIII} } \]

\[ \frac{1}{2}(\text{IIII} \times \text{IIII}) \]

\[ \frac{\text{IIII} + \text{III} }{\text{IIII} + \text{I} + \text{II} + \text{III} } \]

\[ \frac{p_{\text{in}}}{p_{\text{in}} + p_{\text{ex}} } \]

\[ \frac{\text{IIIIIIII} }{\frac{1}{6} (\text{IIII} \times \text{IIII} \times \text{IIII} ) } \]

\[ W_p = p_{\text{in}} \]

\[ W_e \]

\[ W_i \]

\[ W_\Delta \]
Outline

Introduction

Algorithms IS and RaRe

Experiments

Conclusions and future work
Iterative Scan (IS) algorithm

Begin with some cluster “seed”

Traverses nodes, adding or removing the node while the cluster weight improves

Works for any choice of weight metric $W$

---

procedure IS(seed, $G$, $W$)

$C \leftarrow$ seed; $w \leftarrow W(C)$;

increased $\leftarrow$ TRUE;

while increased do

for all $v \in V(G)$ do

if $v \in C$ then

$C' \leftarrow C \setminus \{v\}$;

else

$C' \leftarrow C \cup \{v\}$;

if $W(C') > W(C)$ then

$C \leftarrow C'$;

if $W(C) = w$ then

increased $\leftarrow$ FALSE;

else

$w \leftarrow W(C)$;

return $C$;
IS algorithm
Seed clusters

IS depends on having good seed clusters
These clusters should represent the entire collection of objects

One option: use an existing partitioning algorithm to create the seed clusters

Another option: create a new algorithm with a global view to create seed clusters (possibly overlapping)
Rank Removal (RaRe) algorithm

Idea: split the graph into components by removing a few key players (high-rank nodes)
Then, add the nodes back into whatever clusters they improve (possibly more than one)

\[
\text{procedure RaRe}(C, W) \\
global R \leftarrow \emptyset; \\
\{H_i\} \text{ are connected components in } C; \\
\text{for all } H_i \text{ do} \\
\quad \text{ClusterComponent}(H_i); \\
\text{Initial clusters } \{C_i\} \text{ are cluster cores}; \\
\text{for all } v \in R \text{ do} \\
\quad \text{for all Clusters } C_i \text{ do} \\
\quad \quad \text{Add } v \text{ to cluster } C_i \text{ if } v \text{ is adjacent to } C_i \text{ or } W(v \cup C_i) > W(C_i); \\
\]

\[
\text{procedure ClusterComponent}(H) \\
\text{if } |V(H)| > \max \text{ then} \\
\quad \{v_i\} \text{ are } t \text{ highest rank nodes in } H; \\
\quad R \leftarrow R \cup \{v_i\}; H \leftarrow H \setminus \{v_i\}; \\
\quad \{F_i\} \text{ are connected components in } H; \\
\quad \text{for all } F_i \text{ do} \\
\quad \quad \text{ClusterComponent}(F_i); \\
\text{else if } \min \leq |V(H)| \leq \max \text{ then} \\
\quad \text{mark } H \text{ as a cluster core}; \\
\]
RaRe algorithm
$k$-Neighborhood algorithm

A naïve overlapping clustering procedure

Selects random cluster centers

Cluster contains all nodes distance at most $k$ from the center

“Baseline” for comparison with IS, RaRe
Outline

Introduction

Algorithms IS and RaRe

Experiments

Conclusions and future work
Experiments

Runtime and quality analysis

Compare algorithms on both real-world and simulated inputs
Simulated input

Random

Preferential attachment

Group random
Runtime for simulated input

The graph shows the runtime (s) against the number of nodes. There are two curves: one for RaRe represented by circles and one for IS represented by crosses. The runtime increases significantly with the number of nodes, indicating a non-linear relationship.
Quality for simulated input

![Bar chart showing the average weight value for different methods: Random, Group random, Preferential Attachment with 2-N, RaRe, IS, 2-N + IS, RaRe + IS.](chart)
Real-world graphs

575,600 node subset of the CiteSeer database

E-mail communications among RPI community over two days

Web graph of Malik Magdon-Ismail’s website (www.cs.rpi.edu/~magdon)

Newsgroup posts on alt.conspiracy
Quality for real-world graphs
Outline

Introduction

Algorithms IS and RaRe

Experiments

Conclusions and future work
Conclusions

First overlapping clustering algorithms successfully developed and tested

Local optimality is an intuitive criterion for overlapping clusters

RaRe/2-N improved by IS was best overall
Future work

More robust testing with various weight metrics, and with known clusters

Develop new, more efficient algorithms for clustering

Detect communities that evolve over time