Discrete Ricci Curvature and Ricci Flow for Graph Mining

Jie Gao

Computer Science Department
Stony Brook University
http://www.cs.sunysb.edu/~jgao

Feng Luo (Rutgers), David Gu (Stony Brook)
ATD PI Meeting, September 15th, 2017

Mobile Networks

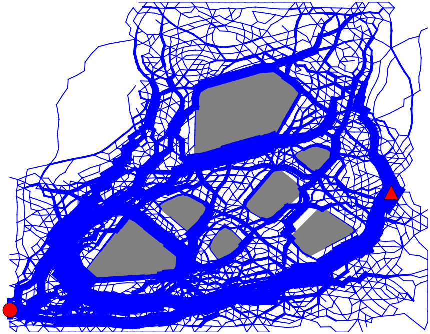
- Mobile devices roaming around.

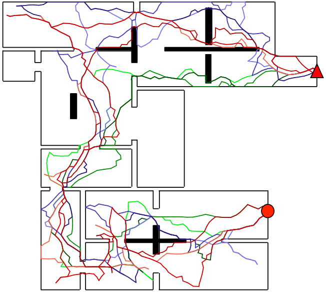
Mobile Networks

- Mobile devices roaming around.
- Vehicles trajectories, human motion behaviors.

Mobile Networks

- Mobile devices roaming around.
- Vehicles trajectories, human motion behaviors.





Social Networks

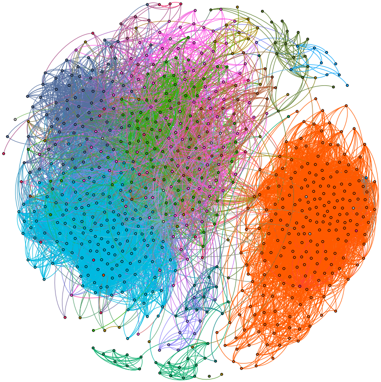
- Characterizing the social interactions.

Social Networks

- Characterizing the social interactions.
- Small world phenomena, power law degree distribution, community structures.

Social Networks

- Characterizing the social interactions.
- Small world phenomena, power law degree distribution, community structures.

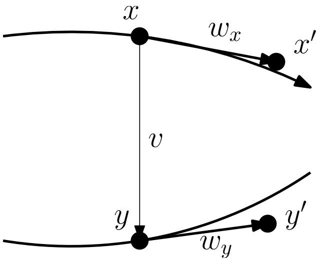


Outline

- Discrete graph curvatures;
- Applications on graph mining and anomaly detection.

Sectional Curvature in Geometry

Consider a tangent vector $v = xy$ and another tangent vector w_x at x. Transport w_x along v to be a tangent vector w_y at y.
If $|x'y'| < |xy|$, then sectional curvature is positive.



Ricci curvature: averaging over all directions w.

Discrete Ricci Curvature

Take the analog: for an edge xy, consider the distances from x**'s neighbors** to y**'s neighbors** and compare it with the length of xy.

Discrete Ricci Curvature

Take the analog: for an edge xy, consider the distances from x**'s neighbors** to y**'s neighbors** and compare it with the length of xy.

- Issue: how to match x's neighbors to y's neighbors?

Discrete Ricci Curvature

Take the analog: for an edge xy, consider the distances from x**'s neighbors** to y**'s neighbors** and compare it with the length of xy.

- Issue: how to match x's neighbors to y's neighbors?
- Assign uniform distribution μ_1, μ_2 on x' and y's neighbors.
- Use optimal transportation distance (earth-mover distance) from μ_1 to μ_2: the matching that minimizes the total transport distance.

Discrete Ricci Curvature

Definition (Ollivier, Lin & Yau)

Let (X, d) be a metric space and let m_1, m_2 be two probability measures on X. For any two distinct points $x, y \in X$, the (Ollivier-) Ricci curvature along xy is defined as

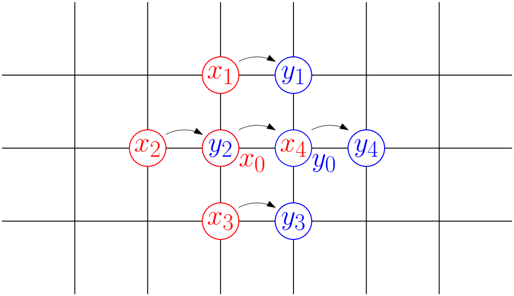
$$\kappa(x, y) := 1 - \frac{W_1(m_x, m_y)}{d(x, y)},$$

where m_x (m_y) is a probability distribution defined on x (y) and its neighbors, $W_1(\mu_1, \mu_2)$ is the L_1 **optimal transportation distance** between two probability measure μ_1 and μ_2 on X:

$$W_1(\mu_1, \mu_2) := \inf_{\psi \in \Pi(\mu_1, \mu_2)} \int_{(u,v)} d(u, v) d\psi(u, v)$$

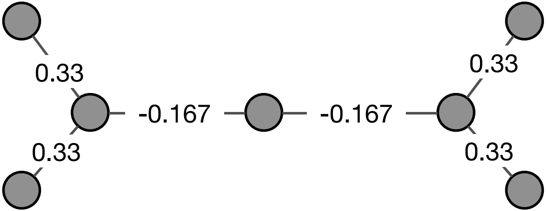
Examples

Zero curvature: 2D grid.



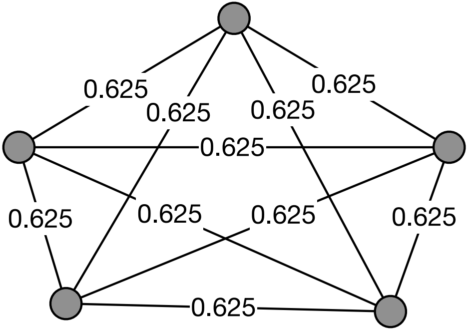
Examples

Negative curvature: tree: $\kappa(x, y) = 1/d_x + 1/d_y - 1$, d_x is degree of x.



Examples

Positive curvature: complete graph.



Data Sets

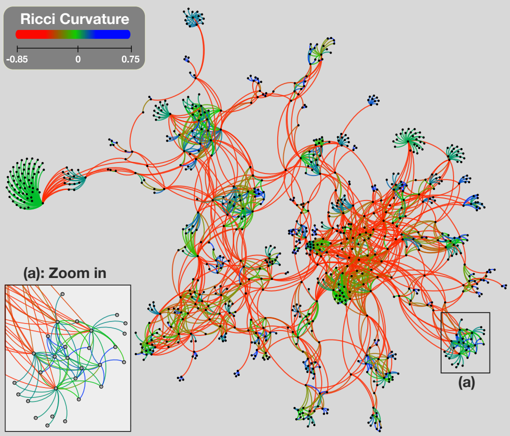
Data Set	Node	Edge	MaxDeg	AvgDeg	Diam	Mean SPL
RouteView	6474	12572	1458	3.88	9	3.71
Gnutella	6301	20777	97	6.59	9	4.64
PGP	10680	24340	205	4.56	24	7.48
Power grid	4941	6594	19	2.67	46	18.99

RocketFuel data set: router level network.

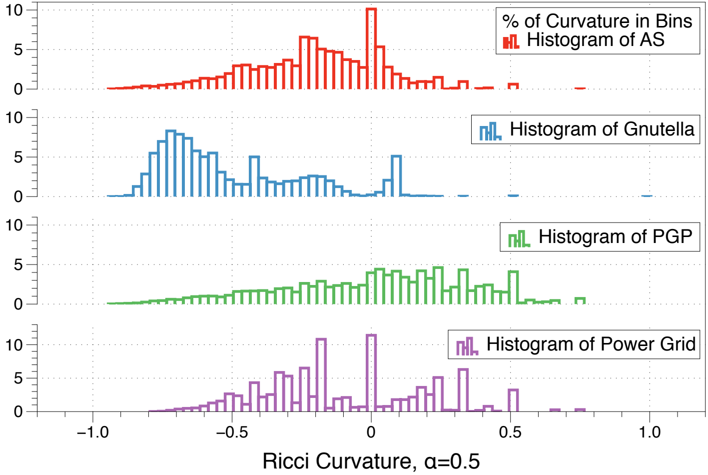
RochetFuel	Node	Edge	MaxDeg	AvgDeg	Diam	Mean SPL
AS: 7018	10145	14166	78	2.04	11	6.95
AS: 3257	843	1156	90	2.23	14	5.27
AS: 3967	895	2071	75	4.63	13	5.94
AS: 1221	2998	3789	106	2.53	12	5.53

Curvature Distribution

Negatively curved edges are like “backbones”, maintaining the connectivity of clusters, in which edges are mostly positively curved.

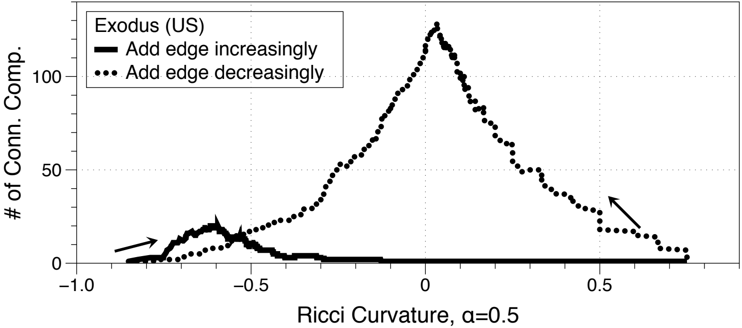


Curvature Distribution



Network Connectivity

Negatively curved edges are well connected. Adding edges with increasing/decreasing curvature: few/many connected components.



Applications: Graph Mining

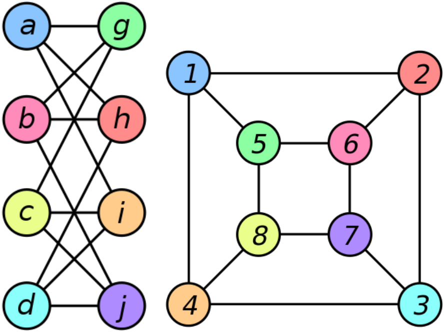
- Comparing two graphs;
- Graph classification;
- Anomaly detection.

Graph Isomorphism

Given a pair of graphs G_1, G_2, find a one-to-one correspondence of the vertices in G_1 to vertices in G_2 such that (u, v) is an edge in G_1 if and only if their corresponding nodes $f(u), f(v)$ are connected in G_2.

Graph Isomorphism

Given a pair of graphs G_1, G_2, find a one-to-one correspondence of the vertices in G_1 to vertices in G_2 such that (u, v) is an edge in G_1 if and only if their corresponding nodes $f(u), f(v)$ are connected in G_2.



Graph Isomorphism

One of the most fundamental problems in theoretical computer science.

- In NP.

Graph Isomorphism

One of the most fundamental problems in theoretical computer science.

- In NP.
- Nov 2015/Jan 2017, László Babai claimed quasi-polynomial time algorithm: $O(\exp(\log^{O(1)} n))$.

Graph Isomorphism

One of the most fundamental problems in theoretical computer science.

- In NP.
- Nov 2015/Jan 2017, László Babai claimed quasi-polynomial time algorithm: $O(\exp(\log^{O(1)} n))$.
- Many practical algorithms: e.g., NAUTY.

Graph Isomorphism

One of the most fundamental problems in theoretical computer science.

- In NP.
- Nov 2015/Jan 2017, László Babai claimed quasi-polynomial time algorithm: $O(\exp(\log^{O(1)} n))$.
- Many practical algorithms: e.g., NAUTY.
- Subgraph isomorphism is NP-complete.

Graph Isomorphism

One of the most fundamental problems in theoretical computer science.

- In NP.
- Nov 2015/Jan 2017, László Babai claimed quasi-polynomial time algorithm: $O(\exp(\log^{O(1)} n))$.
- Many practical algorithms: e.g., NAUTY.
- Subgraph isomorphism is NP-complete.
- Approximate graph isomorphism: find the best correspondence between vertices in G_1 and G_2.

Applications of Graph Isomorphim

- Understanding network evolution.

Applications of Graph Isomorphim

- Understanding network evolution.
- Match two social networks on the same population – de-anonymization.

Applications of Graph Isomorphim

- Understanding network evolution.
- Match two social networks on the same population – de-anonymization.
- Anomaly?

Quantify the ‘Position’ of a Node

Select well positioned nodes as *landmarks* and define the position of a node wrt landmarks.

Quantify the ‘Position’ of a Node

Select well positioned nodes as *landmarks* and define the position of a node wrt landmarks.

Q: What distance to use?

Quantify the 'Position' of a Node

Select well positioned nodes as *landmarks* and define the position of a node wrt landmarks.

Q: What distance to use?

- Internet: Measure the delay;
- Social Networks: measure the tie strength; – Trouble: time-consuming, difficult.

Quantify the 'Position' of a Node

Select well positioned nodes as *landmarks* and define the position of a node wrt landmarks.

Q: What distance to use?

- Internet: Measure the delay;
- Social Networks: measure the tie strength; – Trouble: time-consuming, difficult.
- Count # hops to these landmarks – Trouble: small world property;

Quantify the 'Position' of a Node

Select well positioned nodes as *landmarks* and define the position of a node wrt landmarks.

Q: What distance to use?

- Internet: Measure the delay;
- Social Networks: measure the tie strength; – Trouble: time-consuming, difficult.
- Count # hops to these landmarks – Trouble: small world property;

Our Idea: Edge Weights Generated by Ricci flow

Given a graph G in which $d(x, y)$ is the weight of the edge xy and $\kappa(x, y)$ is the discrete Ricci curvature, we run
/* Ricci flow

$$d_{i+1}(x, y) = d_i(x, y) - \varepsilon \cdot \kappa_i(x, y) \cdot d_i(x, y)$$

/* Normalization

$$d_i(x, y) \leftarrow d_i(x, y) \cdot \frac{|E|}{\sum_{\overline{xy} \in E} d_i(x, y)}$$

Until convergence.

Our Idea: Edge Weights Generated by Ricci flow

Given a graph G in which $d(x, y)$ is the weight of the edge xy and $\kappa(x, y)$ is the discrete Ricci curvature, we run
/* Ricci flow

$$d_{i+1}(x, y) = d_i(x, y) - \varepsilon \cdot \kappa_i(x, y) \cdot d_i(x, y)$$

/* Normalization

$$d_i(x, y) \leftarrow d_i(x, y) \cdot \frac{|E|}{\sum_{\overline{xy} \in E} d_i(x, y)}$$

Until convergence.
Ricci Flow Metric: Shortest path metric with edge weights computed above.

[Ni, Lin, G, Gu under submission]

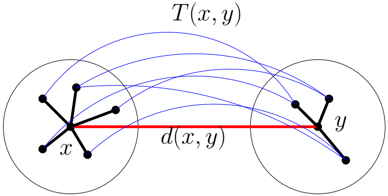
Ricci Flow Metric

$$(d(x,y) - \kappa(x,y) \cdot d(x,y)) \cdot N \approx d(x,y)$$

Ricci Flow Metric

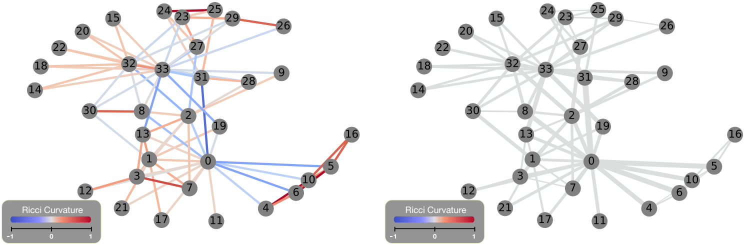
$$(d(x,y) - \kappa(x,y) \cdot d(x,y)) \cdot N \approx d(x,y)$$

$$\frac{T(x,y)}{d(x,y)} \approx \frac{1}{N}.$$

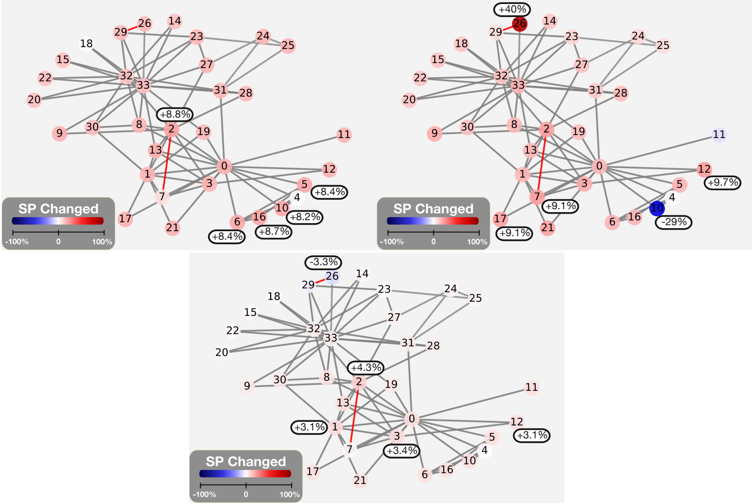


If (x, y) is removed, there are likely alternative paths from x to y that give similar distance. $\rightarrow$ Much better robustness under edge insertion/deletion.

Ricci Flow Metric



Ricci Flow Metric



Conclusion

Geometric approaches for network analysis has great potential.

- Community detection.
- Anomaly detection.

Acknowledgement

- Yu-Yao Lin, Dr. Chien-Chun Ni, David Gu (Stony Brook), Feng Luo (Rutgers).

- https://www3.cs.stonybrook.edu/~jgao
- Questions and comments?