

Carnegie Mellon

Mining graphs and time series: patterns, anomalies, and fraud detection


Part 1: Graphs

Anomaly detection & B.P.

Christos Faloutsos

CMU SCS


<https://www.cs.cmu.edu/~christos/TALKS/19-Gol>



1

Carnegie Mellon

Roadmap




- Introduction
- Part#1: Graphs and Tensors
- Part#2: Time series
- Part#3: extras (visualization, etc)
- Conclusions

Gov. of India Copyright (C) 2019 C. Faloutsos

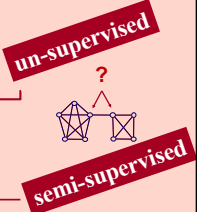
2

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Graphs
 - ...
 - P1.3: community detection
 - ➔ – P1.4: fraud/anomaly detection
 - P1.5: belief propagation




Gov. of India Copyright (C) 2019 C. Faloutsos

3


Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Graphs
 - ...
 - P1.3: community detection
 - P1.4: fraud/anomaly detection
 - P1.4.1. Outliers
 - P1.4.2. Lock-step behavior
 - P1.5: belief propagation

un-supervised



Gov. of India

Copyright (C) 2019 C. Faloutsos





4

4

Carnegie Mellon

'Recipe' Structure:

- Problem definition
- Short answer/solution
- LONG answer – details
- Conclusion/short-answer

Gov. of India

Copyright (C) 2019 C. Faloutsos

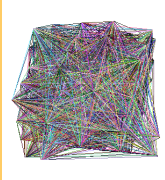
5

5

Carnegie Mellon

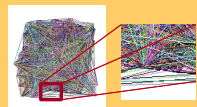


Problem

Given:



Find:

- 1) Outliers
- 2) Lock-step

Gov. of India

Copyright (C) 2019 C. Faloutsos

6

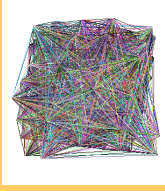
6

2

Carnegie Mellon

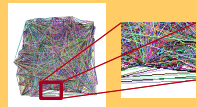
Solution

Given:





Find:

- 1) Outliers
- 2) Lock-step



OddBall

SVD

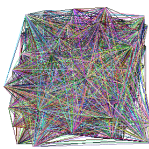
Gov. of India Copyright (C) 2019 C. Faloutsos 7

7

Carnegie Mellon

P1.4.1. Outliers

- Which node(s) are strange?
 - Q: How to start?



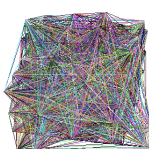
Gov. of India Copyright (C) 2019 C. Faloutsos 8

8

Carnegie Mellon

P1.4.1. Outliers

- Which node(s) are strange?
 - Q: How to start?
 - A1: egonet; and extract node features

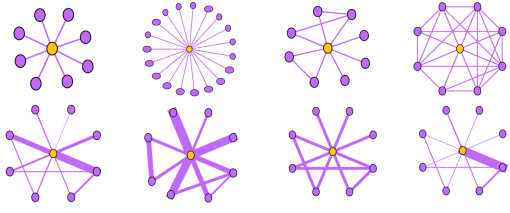



Gov. of India Copyright (C) 2019 C. Faloutsos 9

9

Carnegie Mellon

Ego-net Patterns: Which is strange?

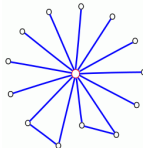



Oddball: Spotting anomalies in weighted graphs, Leman Akoglu, Mary McGlohon, Christos Faloutsos, PAKDD 2010

10

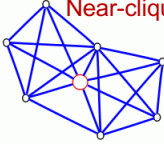
Carnegie Mellon

Ego-net Patterns: Which is strange?



Near-star

telemarketer, port scanner, people adding friends indiscriminately, etc.



Near-clique

tightly connected people, terrorist groups?, discussion group, etc.

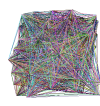
Gov. of India Copyright (C) 2019 C. Faloutsos

11

Carnegie Mellon

P1.4.1. Outliers

- Which node(s) are strange?
 - Q: How to start?
 - A: egonet; and extract node features
 - Q': which features?
 - A': ART! Infinite! Pick a few, e.g.:




Gov. of India Copyright (C) 2019 C. Faloutsos

12


Carnegie Mellon

Ego-net Patterns

- N_i : number of neighbors (degree) of ego i
- E_i : number of edges in egonet i



- W_i : total weight of egonet i
- $\lambda_{w,i}$: principal eigenvalue of the weighted adjacency matrix of egonet i

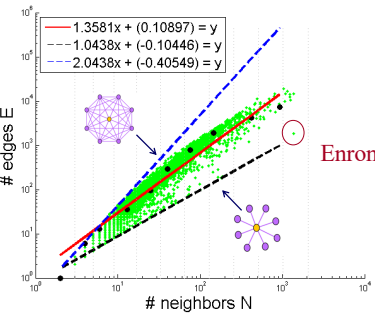


Oddball: Spotting anomalies in weighted graphs, Leman Akoglu, Mary McGlohon, Christos Faloutsos, PAKDD 2010

13

Carnegie Mellon

Pattern: Ego-net Power Law Density



$E_i \propto N_i^\alpha$
 $1 \leq \alpha \leq 2$

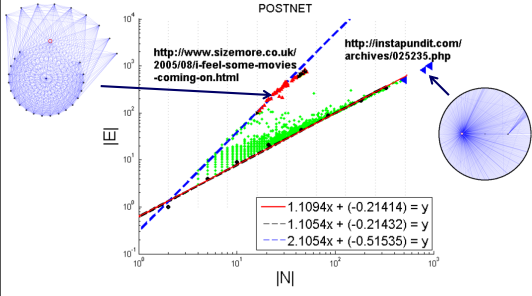
Enron CEO

Oddball: Spotting anomalies in weighted graphs, Leman Akoglu, Mary McGlohon, Christos Faloutsos, PAKDD 2010

14

Carnegie Mellon

Pattern: Ego-net Power Law Density




$E_i \propto N_i^\alpha$
 $1 \leq \alpha \leq 2$

Oddball: Spotting anomalies in weighted graphs, Leman Akoglu, Mary McGlohon, Christos Faloutsos, PAKDD 2010


15

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Graphs
 - ...
 - P1.3: community detection
 - P1.4: fraud/anomaly detection
 - Outliers
 - Lock-step behavior
 - P1.5: belief propagation



Gov. of India


Copyright (C) 2019 C. Faloutsos

16

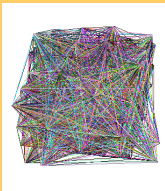
16

Carnegie Mellon

Problem

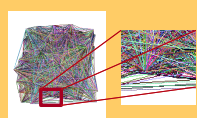



Given:



Find:

- 1) Outliers
- 2) Lock-step

Gov. of India

Copyright (C) 2019 C. Faloutsos


17

17


Carnegie Mellon

P1.4.1. How to find ‘suspicious’ groups?

- ‘blocks’ are normal, right?



idols



fans

Gov. of India

Copyright (C) 2019 C. Faloutsos

18


18

Carnegie Mellon

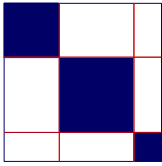
P1.4.1. How to find 'suspicious' groups?

- 'blocks' are normal, right?

idols



fans




Gov. of India Copyright (C) 2019 C. Faloutsos 19


19

Carnegie Mellon

Except that:

- 'blocks' are normal, right?
- 'hyperbolic' communities are more realistic [Araujo+, PKDD'14]





Gov. of India Copyright (C) 2019 C. Faloutsos 20


20


Carnegie Mellon

Except that:

- 'blocks' are usually suspicious
- 'hyperbolic' communities are more realistic [Araujo+, PKDD'14]

Q: Can we spot blocks, easily?





Gov. of India Copyright (C) 2019 C. Faloutsos 21

21

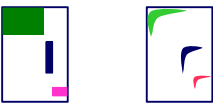
Carnegie Mellon

Except that:

- 'blocks' are usually **suspicious**
- 'hyperbolic' communities are more realistic [Araujo+, PKDD'14]

Q: Can we spot blocks, easily?

A: Silver bullet: SVD!



Gov. of India Copyright (C) 2019 C. Faloutsos 22

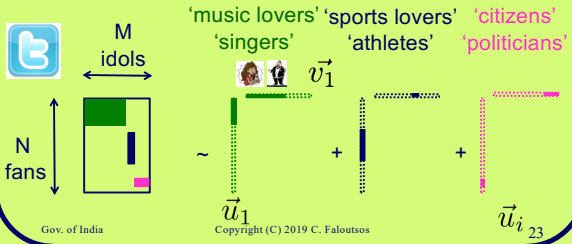
22

Carnegie Mellon

Crush intro to SVD

From: HITS

- Recall: (SVD) matrix factorization: finds blocks



Gov. of India Copyright (C) 2019 C. Faloutsos

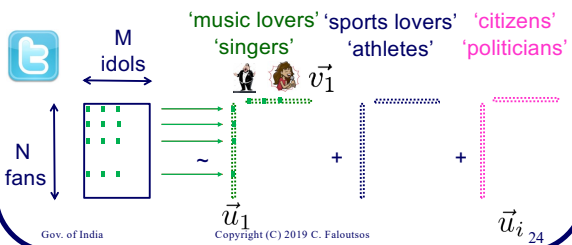
23

Carnegie Mellon

Crush intro to SVD

- (SVD) matrix factorization: finds blocks

A) Even if shuffled!



Gov. of India Copyright (C) 2019 C. Faloutsos

24

Carnegie Mellon

Crush intro to SVD

- (SVD) matrix factorization: finds blocks
- B) Even if 'salt+pepper' noise**

Gov. of India

Copyright (C) 2019 C. Faloutsos

25

Carnegie Mellon

SVD - intuition

- Hidden/latent variable detection
- Block detection
- Dimensionality reduction
- embedding

Gov. of India

Copyright (C) 2019 C. Faloutsos

26

Carnegie Mellon

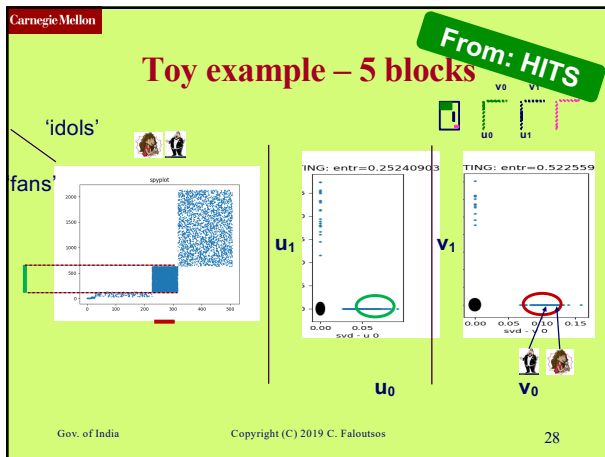
Toy example – 5 blocks

From: HITS

Gov. of India

Copyright (C) 2019 C. Faloutsos

27



28

Carnegie Mellon

SVD - intuition

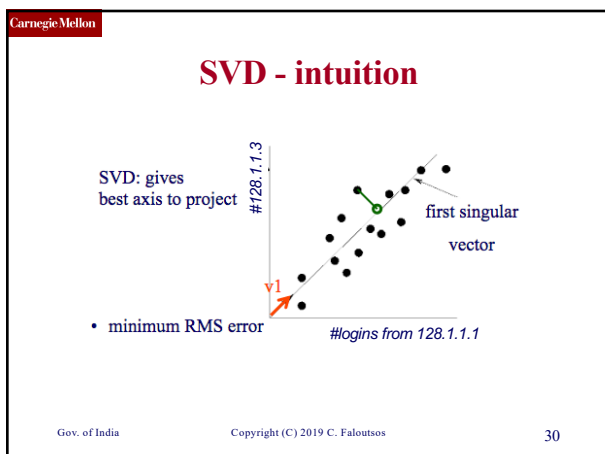
- Hidden/latent variable detection
- Block detection
- Dimensionality reduction
- embedding

Gov. of India

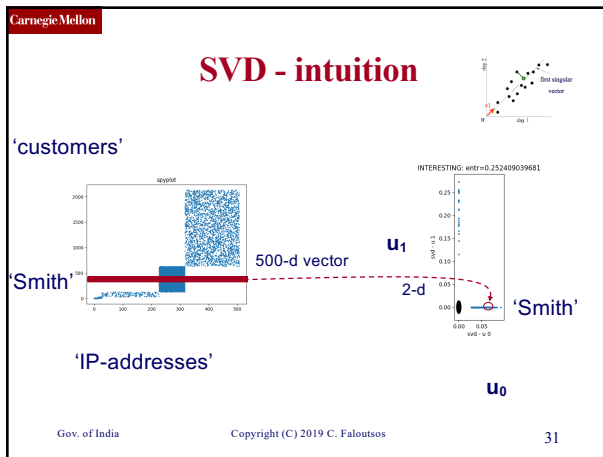
Copyright (C) 2019 C. Faloutsos

29

29



30



31

Carnegie Mellon

SVD - intuition

- Hidden/latent variable detection
- Block detection
- Dimensionality reduction
- Embedding (linear)
 - SVD is a special case of 'deep neural net'

Gov. of India Copyright (C) 2019 C. Faloutsos 32


32


Carnegie Mellon

OPTIONAL

Inferring Strange Behavior from Connectivity Pattern in Social Networks

PAKDD'14





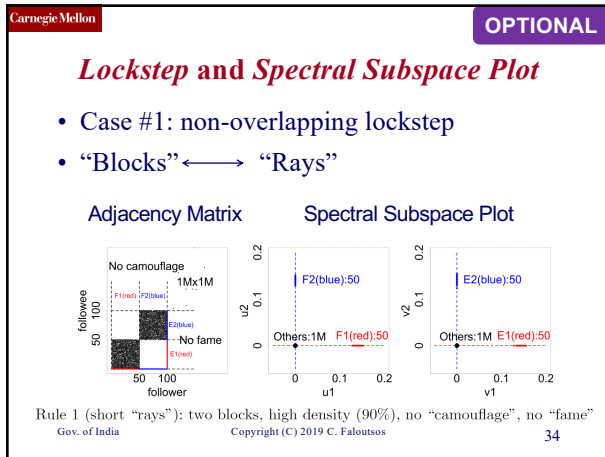
Meng Jiang, Peng Cui, Shiqiang Yang (Tsinghua)

Alex Beutel, Christos Faloutsos (CMU)

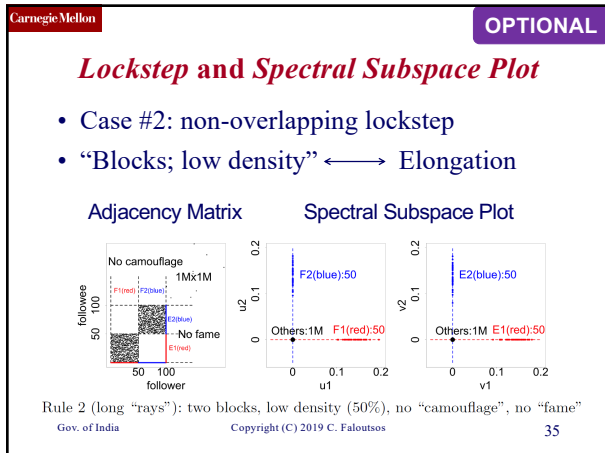


Gov. of India Copyright (C) 2019 C. Faloutsos 33

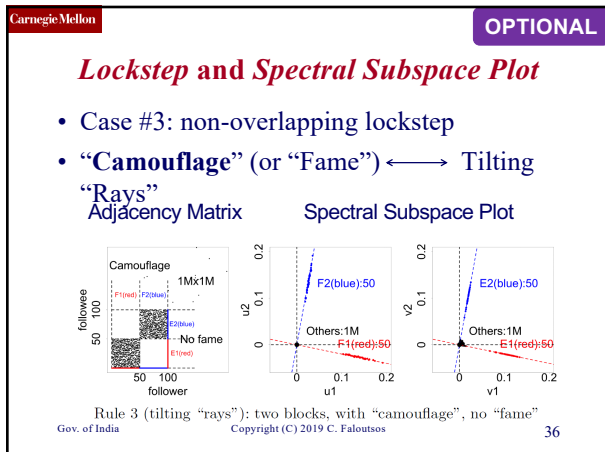
33



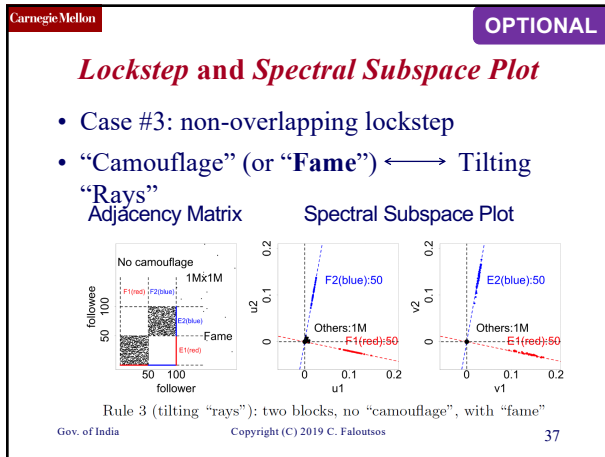
34



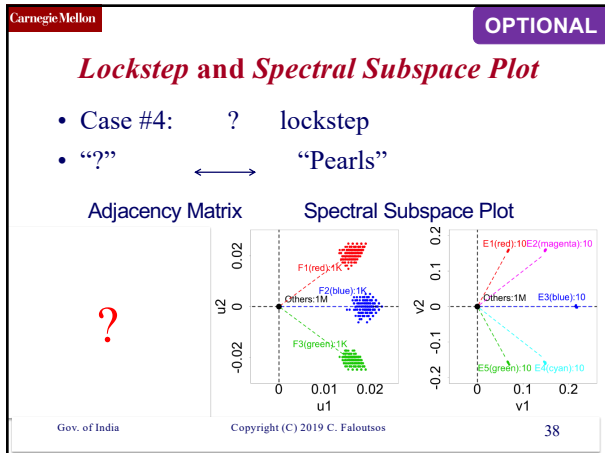
35



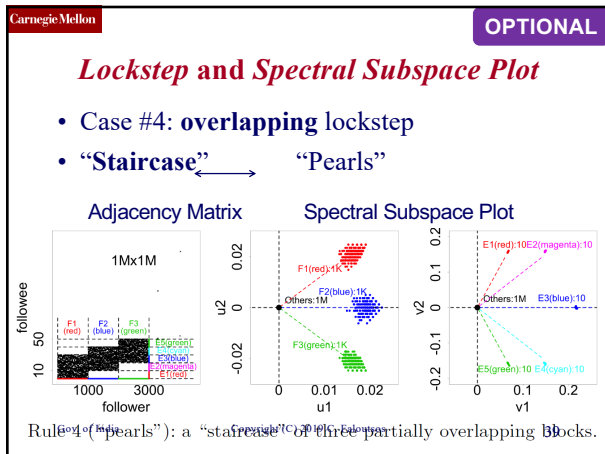
36



37



38




39

Carnegie Mellon

Dataset

- Tencent Weibo
- 117 million nodes (with profile and UGC data)
- 3.33 billion directed edges



Gov. of India

Copyright (C) 2019 C. Faloutsos

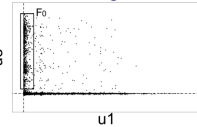
40

40

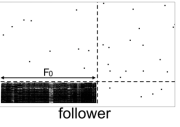
Carnegie Mellon

Real Data

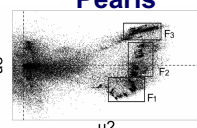
“Rays”



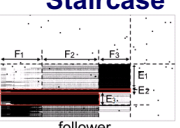
“Block”



“Pearls”



“Staircase”



Gov. of India

Copyright (C) 2019 C. Faloutsos

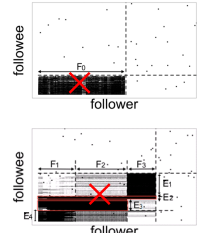
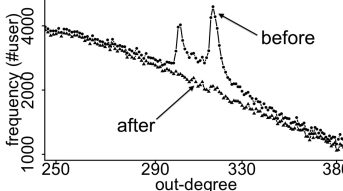
41

41

Carnegie Mellon

Real Data

- Spikes on the out-degree distribution

Gov. of India

Copyright (C) 2019 C. Faloutsos

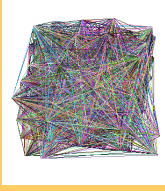
42

42

Carnegie Mellon

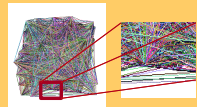
Solution

Given:





Find:

- 1) Outliers
- 2) Lock-step



OddBall

SVD





Gov. of India Copyright (C) 2019 C. Faloutsos 43


43

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Graphs
 - ...
 - P1.3: community detection
 - P1.4: fraud/anomaly detection
 - ➔ – P1.5: belief propagation



un-supervised


semi-supervised

Gov. of India Copyright (C) 2019 C. Faloutsos 44


44

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Graphs
 - P1.1: properties/patterns in graphs
 - P1.2: node importance
 - P1.3: community detection
 - P1.4: fraud/anomaly detection
 - ➔ – P1.5: belief propagation



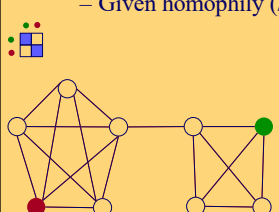
Gov. of India Copyright (C) 2019 C. Faloutsos 45

45

Carnegie Mellon

Problem

- What color, for the rest?
 - Given homophily (/heterophily etc)?



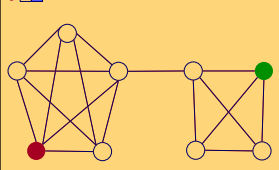
Gov. of India Copyright (C) 2019 C. Faloutsos 46

46


Carnegie Mellon

Short answer:

- What color, for the rest?
- A: Belief Propagation ('zooBP')



www.cs.cmu.edu/~deswaran/code/zooBP.zip




Gov. of India Copyright (C) 2019 C. Faloutsos 47

47

Carnegie Mellon

Roadmap

- Introduction – Motivation
- Part#1: Graphs
 - ...
 - P1.5: belief propagation
 - Basics
 - Fast, linear approximation (FaBP)
 - Latest: zooBP
 - Success stories





Gov. of India Copyright (C) 2019 C. Faloutsos 48

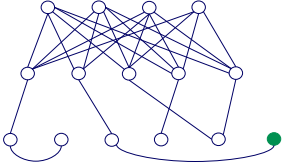
48

Carnegie Mellon

E-bay Fraud detection



w/ Polo Chau &
Shashank Pandit, CMU
[PKDD'06][WWW'07]

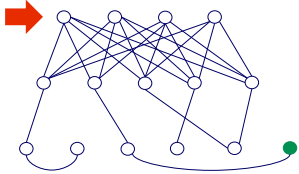


Detecting Fraudulent Personalities in Networks of Online Auctioneers. Duen Horng (Polo) Chau, Shashank Pandit, and Christos Faloutsos. (PKDD) 2006

49

Carnegie Mellon

E-bay Fraud detection



Gov. of India

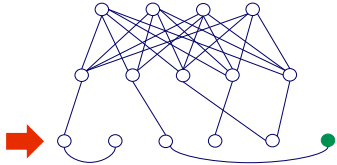
Copyright (C) 2019 C. Faloutsos

50

50

Carnegie Mellon

E-bay Fraud detection



Gov. of India

Copyright (C) 2019 C. Faloutsos

51

51

E-bay Fraud detection - NetProbe

The screenshot shows the NetProbe application interface. On the left, there's a network graph with nodes representing users and items. On the right, there's a list of suspicious items, including 'alisher' with a 'Suspicious Reason' and a 'Suspicious Score'.

Gov. of India Copyright (C) 2019 C. Faloutsos 52

52

Background

Prof. Danai Koutra
U. Michigan

Gov. of India Copyright (C) 2019 C. Faloutsos 53

53

Belief Propagation

- Iterative message-based method
- “Propagation matrix”:
 - ◊ Homophily

class of receiver

class of sender

0.9	0.1
0.1	0.9

AI

PL

stop criterion fulfilled

[Pearl '82][Yedidia+ '02] ... [Gonzalez+ '09][Checheta+ '10]

Gov. of India Copyright (C) 2019 C. Faloutsos 54

54

Ca

Belief Propagation Equations

message($i \rightarrow j$) \approx belief(i) \times homophily strength

0.9	0.1
0.2	0.8

Gov. of India

Copyright (C) 2019 C. Faloutsos

55

55

Ca

Belief Propagation Equations

belief of i

$b_i(x_i) \propto \phi_i(x_i) \cdot \prod_{j \in N(i)} m_{ij}(x_i)$

prior belief

messages from neighbors

Gov. of India

Copyright (C) 2019 C. Faloutsos

56

56

Ca

Background

Gov. of India


Copyright (C) 2019 C. Faloutsos

57


57

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Graphs
 - ...
 - P1.5: belief propagation
 - Basics
 - Fast, linear approximation (FaBP)
 - Latest: zooBP



Gov. of India
Copyright (C) 2019 C. Faloutsos
58

58

Carnegie Mellon

Unifying Guilt-by-Association Approaches: Theorems and Fast Algorithms



Danai Koutra
U Kang
Hsing-Kuo Kenneth Pao

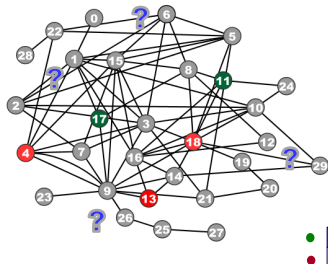
Tai-You Ke
Duen Horng (Polo) Chau
Christos Faloutsos

ECML PKDD, 5-9 September 2011, Athens, Greece

59

Carnegie Mellon

Problem Definition: GBA techniques



Given: Graph; &
few labeled nodes
Find: labels of rest
(assuming network
effects)

Gov. of India
Copyright (C) 2019 C. Faloutsos
60

60

Problem Definition:
GBA techniques

Given: Graph; &
few labeled nodes
Find: labels of rest
(assuming network
effects)

Gov. of India Copyright (C) 2019 C. Faloutsos 61

61

BP vs. Linearized BP **DETAILS**

Original [Yedidia+]:

Belief Propagation

$$m_{ij}(x_j) \leftarrow \sum_{x_i} \phi_i(x_i) \cdot \psi_{ij}(x_i, x_j) \cdot \prod_{n \in N(i) \setminus j} m_{ni}(x_i)$$

$$b_i(x_i) \leftarrow \phi_i(x_i) \cdot \prod_{j \in N(i)} m_{ij}(x_j)$$

non-linear

- Closed-form formula?
- Convergence?

19 C. Faloutsos 62

62

BP vs. Linearized BP **DETAILS**

Original [Yedidia+]:

Belief Propagation

$$m_{ij}(x_j) \leftarrow \sum_{x_i} \phi_i(x_i) \cdot \psi_{ij}(x_i, x_j) \cdot \prod_{n \in N(i) \setminus j} m_{ni}(x_i)$$

$$b_i(x_i) \leftarrow \phi_i(x_i) \cdot \prod_{j \in N(i)} m_{ij}(x_j)$$

non-linear

Our proposal:

Linearized BP
BP is approximated by

$$[\mathbf{I} + a\mathbf{D} - c'\mathbf{A}] \mathbf{b}_h = \phi_h$$

linear

- ✓ Closed-form formula?
- ✓ Convergence?

19 C. Faloutsos 63

63

Carnegie Mellon

Are they related?

- RWR (Random Walk with Restarts)
 - google's pageRank ('if my parents are important, I'm important, too')
- SSL (Semi-supervised learning)
 - minimize the differences among neighbors
- BP (Belief propagation)
 - send messages to neighbors, on what you believe about them

Gov. of India
Copyright (C) 2019 C. Faloutsos
64

64

Carnegie Mellon

Are they related? YES!

- RWR (Random Walk with Restarts)
 - google's pageRank ('if my parents are important, I'm important, too')
- SSL (Semi-supervised learning)
 - minimize the differences among neighbors
- BP (Belief propagation)
 - send messages to neighbors, on what you believe about them

Gov. of India
Copyright (C) 2019 C. Faloutsos
65






65

Carnegie Mellon

Correspondence of Methods

$p = (1-c)/n [I - cB]^{-1} \vec{e}$

Method	Matrix	Unknown	known
RWR	$[I - c \underline{AD}^{-1}] \times$	\mathbf{x}	$(1-c)\mathbf{y}$
SSL	$[I + a(\underline{D} - \underline{A})] \times$	\mathbf{x}	\mathbf{y}
FABP	$[I + a \underline{D} - c' \underline{A}] \times$	\mathbf{b}_h	ϕ_h

adjacency matrix
final labels/beliefs
prior labels/beliefs

Gov. of India
Copyright (C) 2019 C. Faloutsos
66

66

Problem: e-commerce ratings fraud

- **Given a heterogeneous graph on users, products, sellers and positive/negative ratings with “seed labels”**
- **Find the top k most fraudulent users, products and sellers**

Gov. of India Copyright (C) 2019 C. Faloutsos 67

67

Problem: e-commerce ratings fraud

- **Given a heterogeneous graph on users, products, sellers and positive/negative ratings with “seed labels”**
- **Find the top k most fraudulent users, products and sellers**

Dhivya Eswaran, Stephan Günnemann, Christos Faloutsos, “ZooBP: Belief Propagation for Heterogeneous Networks”, VLDB 2017

68

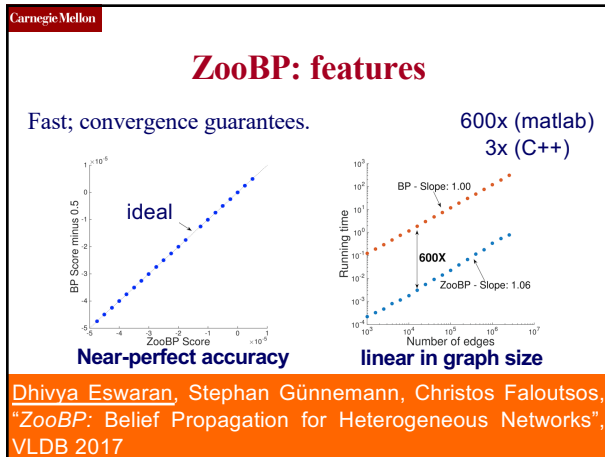
Problem: e-commerce ratings fraud

Theorem 1 (ZooBP). If $\mathbf{b}, \mathbf{e}, \mathbf{P}, \mathbf{Q}$ are constructed as described above, the linear equation system approximating the final node beliefs given by BP is:

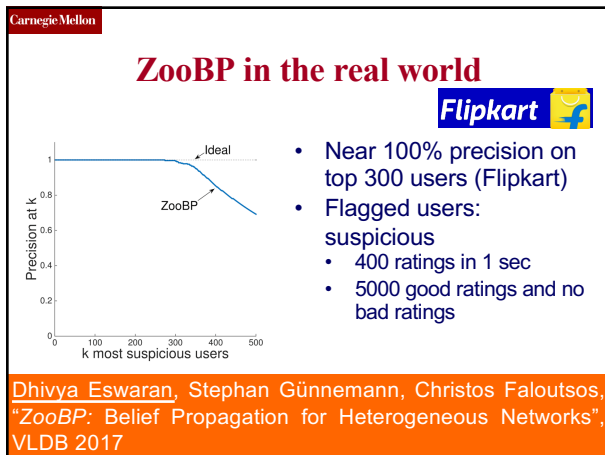
$$\mathbf{b} = \mathbf{e} + (\mathbf{P} - \mathbf{Q})\mathbf{b} \quad (\text{ZooBP}) \quad (10)$$

Dhivya Eswaran, Stephan Günnemann, Christos Faloutsos, “ZooBP: Belief Propagation for Heterogeneous Networks”, VLDB 2017

69



70




71




72

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Graphs
 - ...
 - P1.5: belief propagation
 - Basics
 - Fast, linear approximation (FaBP)
 - Latest: zooBP
 - Success stories



Gov. of India Copyright (C) 2019 C. Faloutsos 73

73

Carnegie Mellon

Other ‘success stories’?

- Accounting fraud
- Malware detection




Gov. of India Copyright (C) 2019 C. Faloutsos 74

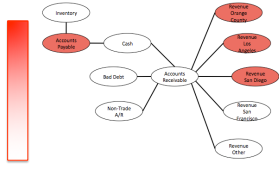

74

Carnegie Mellon

Network Effect Tools: SNARE

- Some accounts are sort-of-suspicious – how to combine weak signals?

Before


Mary McGlohon, Stephen Bay, Markus G. Anderle, David M. Steier, Christos Faloutsos: SNARE: a link analytic system for graph labeling and risk detection. KDD 2009: 1265-1274

75


Carnegie Mellon

Polonium: Tera-Scale Graph Mining and Inference for Malware Detection
SDM 2011, Mesa, Arizona


PATENT PENDING




Polo Chau
Machine Learning Dept




Carey Nachenberg
Vice President & Fellow



Jeffrey Wilhelm
Principal Software Engineer



Adam Wright
Software Engineer




Prof. Christos Faloutsos
Computer Science Dept

76

Carnegie Mellon

Polonium: The Data



60+ terabytes of data *anonymously* contributed by participants of worldwide Norton Community Watch program

50+ million machines

900+ million executable files

Constructed a machine-file bipartite graph (0.2 TB+)

1 billion nodes (machines and files)

37 billion edges

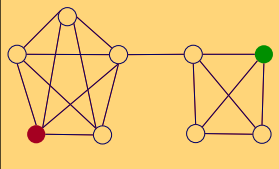
Gov. of India Copyright (C) 2019 C. Faloutsos

77


Carnegie Mellon

Short answer:

- What color, for the rest?
- A: Belief Propagation ('zooBP')



www.cs.cmu.edu/~deswaran/code/zooBP.zip




Gov. of India Copyright (C) 2019 C. Faloutsos

78

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Graphs
 - ...
 - ➔ – conclusions
- Part#2: Tensors and Knowledge Bases
- Conclusions – Future research

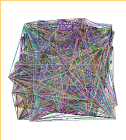
Gov. of India
Copyright (C) 2019 C. Faloutsos
79

79

Carnegie Mellon

Over-arching conclusion

- MANY, time-tested, algorithms for graph mining
- (more, are needed)




Gov. of India
Copyright (C) 2019 C. Faloutsos
80


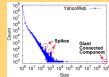




80

Carnegie Mellon

Over-arching conclusion

Problems


(some) solutions

Gov. of India
Copyright (C) 2019 C. Faloutsos
81

81

Carnegie Mellon

In more detail

- (repeating the conclusions from each part P1.1-P1.5)

Gov. of India


Copyright (C) 2019 C. Faloutsos




82

82

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Graphs
 - ➔ – P1.1: properties/patterns in graphs 
 - P1.2: node importance 
 - P1.3: community detection 
 - P1.4: fraud/anomaly detection
 - P1.5: belief propagation

Gov. of India





Copyright (C) 2019 C. Faloutsos

83

83

Carnegie Mellon

Problem definition

- Are real graphs random?
 - S*: what do **static** graphs look like?
 - T*: how do graphs evolve over **time**?

Gov. of India

Copyright (C) 2019 C. Faloutsos

84

84

Carnegie Mellon

Short answer(s)

- Are real graphs random?
 - S*: what do **static** graphs look like?
 - S.0: 'six degrees'
 - S.1: skewed degree distribution
 - S.2: skewed eigenvalues
 - S.3: triangle power-laws
 - S.4: GCC; and skewed distr. of conn. comp.
 - T*: how do graphs evolve over **time**?
 - T.1: diameters
 - T.2: densification

Gov. of India Copyright (C) 2019 C. Faloutsos 85

85

Carnegie Mellon

Short answer(s)

Power laws: $y \sim x^a$
NOT Gaussians
Take logarithms

Are real graphs random?

- S*: what do **static** graphs look like?
 - S.0: 'six degrees'
 - S.1: skewed degree distribution
 - S.2: skewed eigenvalues
 - S.3: triangle power-laws
 - S.4: GCC; and skewed distr. of conn. comp.
- T*: how do graphs evolve over **time**?
 - T.1: diameters
 - T.2: densification

Gov. of India Copyright (C) 2019 C. Faloutsos 86

86

Carnegie Mellon

Roadmap


- Introduction – Motivation
- Part#1: Graphs
 - P1.1: properties/patterns in graphs
 - P1.2: node importance
 - P1.3: community detection
 - P1.4: fraud/anomaly detection
 - P1.5: belief propagation

Gov. of India Copyright (C) 2019 C. Faloutsos 87

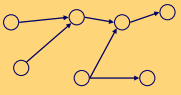
87

Carnegie Mellon

Node importance - Motivation:



- Given a graph (eg., web pages containing the desirable query word)
- Q: Which node is the most important?




Gov. of India Copyright (C) 2019 C. Faloutsos P1-88

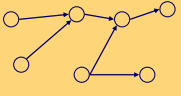
88

Carnegie Mellon

Node importance - Motivation:



- Given a graph (eg., web pages containing the desirable query word)
- Q: Which node is the most important?
- A1: PageRank (PR)
- A2: HITS
- A3: SALSA




Gov. of India Copyright (C) 2019 C. Faloutsos P1-89

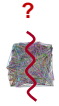

89

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Graphs
 - P1.1: properties/patterns in graphs
 - P1.2: node importance
 - ➔ – P1.3: community detection
 - P1.4: fraud/anomaly detection
 - P1.5: belief propagation

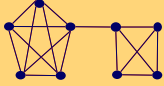
Gov. of India Copyright (C) 2019 C. Faloutsos 90

90

Carnegie Mellon

Problem Definition

- Given a graph, and k
- Break it into k (disjoint) communities




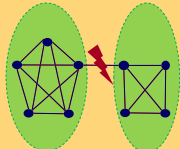
Gov. of India Copyright (C) 2019 C. Faloutsos 91

91

Carnegie Mellon

Short answer

- METIS [Karypis, Kumar]
- (but: maybe NO good cuts exist!)



Gov. of India Copyright (C) 2019 C. Faloutsos 92

92

Carnegie Mellon

Roadmap

- Introduction – Motivation
- Part#1: Graphs
 - P1.1: properties/patterns in graphs
 - P1.2: node importance
 - P1.3: community detection
 - ➔ – P1.4: fraud/anomaly detection
 - P1.5: belief propagation

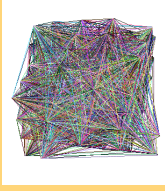



Gov. of India Copyright (C) 2019 C. Faloutsos 93

93


Problem

Given:



Find:

- 1) Outliers
- 2) Lock-step

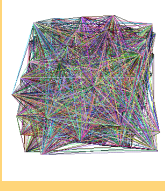


Gov. of India Copyright (C) 2019 C. Faloutsos 94

94

Solution

Given:

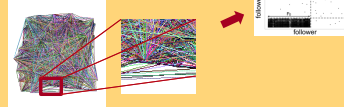


Find:

- 1) Outliers
- 2) Lock-step

OddBall

SVD




Gov. of India Copyright (C) 2019 C. Faloutsos 95

95

Roadmap

- Introduction – Motivation
- Part#1: Graphs
 - P1.1: properties/patterns in graphs
 - P1.2: node importance
 - P1.3: community detection
 - P1.4: fraud/anomaly detection
 - ➔ – P1.5: belief propagation



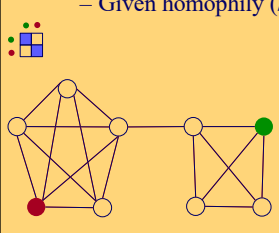
Gov. of India Copyright (C) 2019 C. Faloutsos 96

96

Carnegie Mellon

Problem

- What color, for the rest?
– Given homophily (/heterophily etc)?



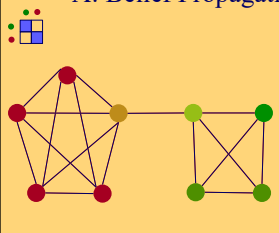
Gov. of India Copyright (C) 2019 C. Faloutsos 97

97


Carnegie Mellon

Short answer:

- What color, for the rest?
- A: Belief Propagation ('zooBP')



www.cs.cmu.edu/~deswaran/code/zooBP.zip



Gov. of India Copyright (C) 2019 C. Faloutsos 98

98

Carnegie Mellon

Thanks to



Danai Koutra
U. Michigan



Namyong Park
CMU



Dhivya Eswaran
CMU



Hyun Ah Song
CMU



Vagelis Papalexakis
UCR


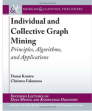
Gov. of India Copyright (C) 2019 C. Faloutsos 99

99

Carnegie Mellon

P1 – Graphs - More references

Danai Koutra and Christos Faloutsos,
[Individual and Collective Graph Mining: Principles, Algorithms, and Applications](#)
 October 2017, Morgan Claypool



Gov. of India Copyright (C) 2019 C. Faloutsos 100

100

Carnegie Mellon

P1 – Graphs - More references

Deepayan Chakrabarti and Christos Faloutsos,
[Graph Mining: Laws, Tools, and Case Studies](#)
 Oct. 2012, Morgan Claypool.

Gov. of India Copyright (C) 2019 C. Faloutsos 101

101

Carnegie Mellon

P1 – Graphs - More references

Anomaly detection

- Leman Akoglu, Hanghang Tong, & Danai Koutra, [Graph based anomaly detection and description: a survey](#) Data Mining and Knowledge Discovery (2015) 29: 626.
- Arxiv version:
<https://arxiv.org/abs/1404.4679>

Gov. of India Copyright (C) 2019 C. Faloutsos 102

102
