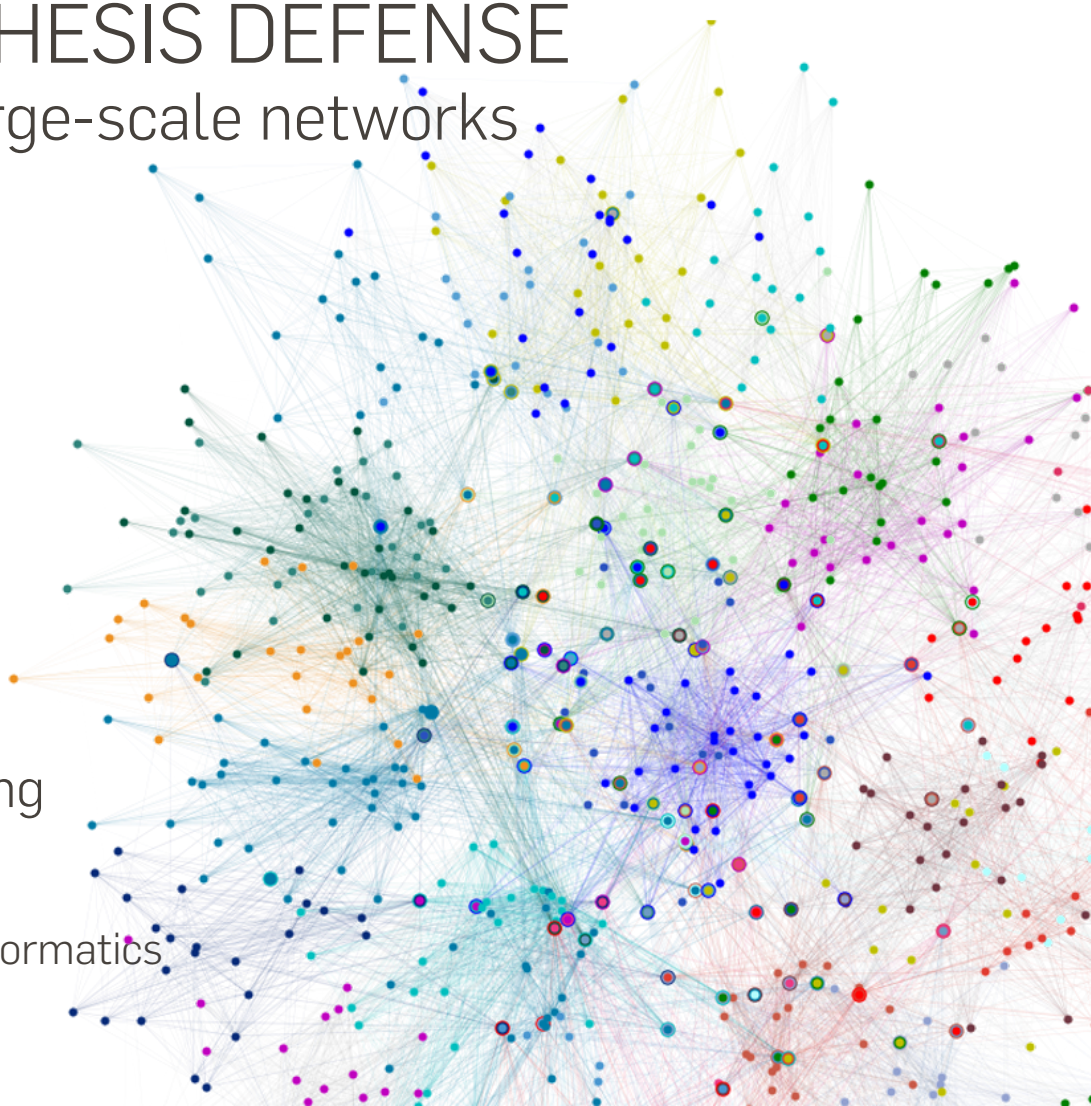
ThangLong University

UNDERGRADUATE THESIS DEFENSE

Community detection in large-scale networks

July 7, 2017



Advisor: Ph.D. Tran Vinh Duc

Student name: Nguyen Duc Thang
Student code: A22852

Computer science - Mathematics and Informatics
ThangLong University

Overview

.

In this thesis, I provide a general view of communities and its the real life applications. I introduce BigCLAM models proposed by Yang and Leskovec (2013), a popular model is used overlapping community detection algorithm. In particular, I proposed a few methods convex optimization and implemented BigCLAM in Apache Spark is evaluated as lightning-fast cluster computing to able detect community in the large-scale networks.

Overview

1. Large-scale Network & Community
2. BigCLAM
3. Grid computing
4. Demo & conclusion

LARGE-SCALE NETWORK & COMMUNITY

Network

Networks are a general language for describing complex systems.

Networks are everywhere!

- → Friends & family
- → Society
- → media & information
- → world economy
- → roads
- → human cell
- → brain

Social media networks


World Wide Web

TRANSPORTATION CLUSTERS

3.200 airports
60.000 routes

GEOGRAPHICAL LAYOUT

Color = Longitude
Size = Number of routes

EUROPE

Alaska

Canada

Russia

NORTH AMERICA

ASIA

AFRICA

India

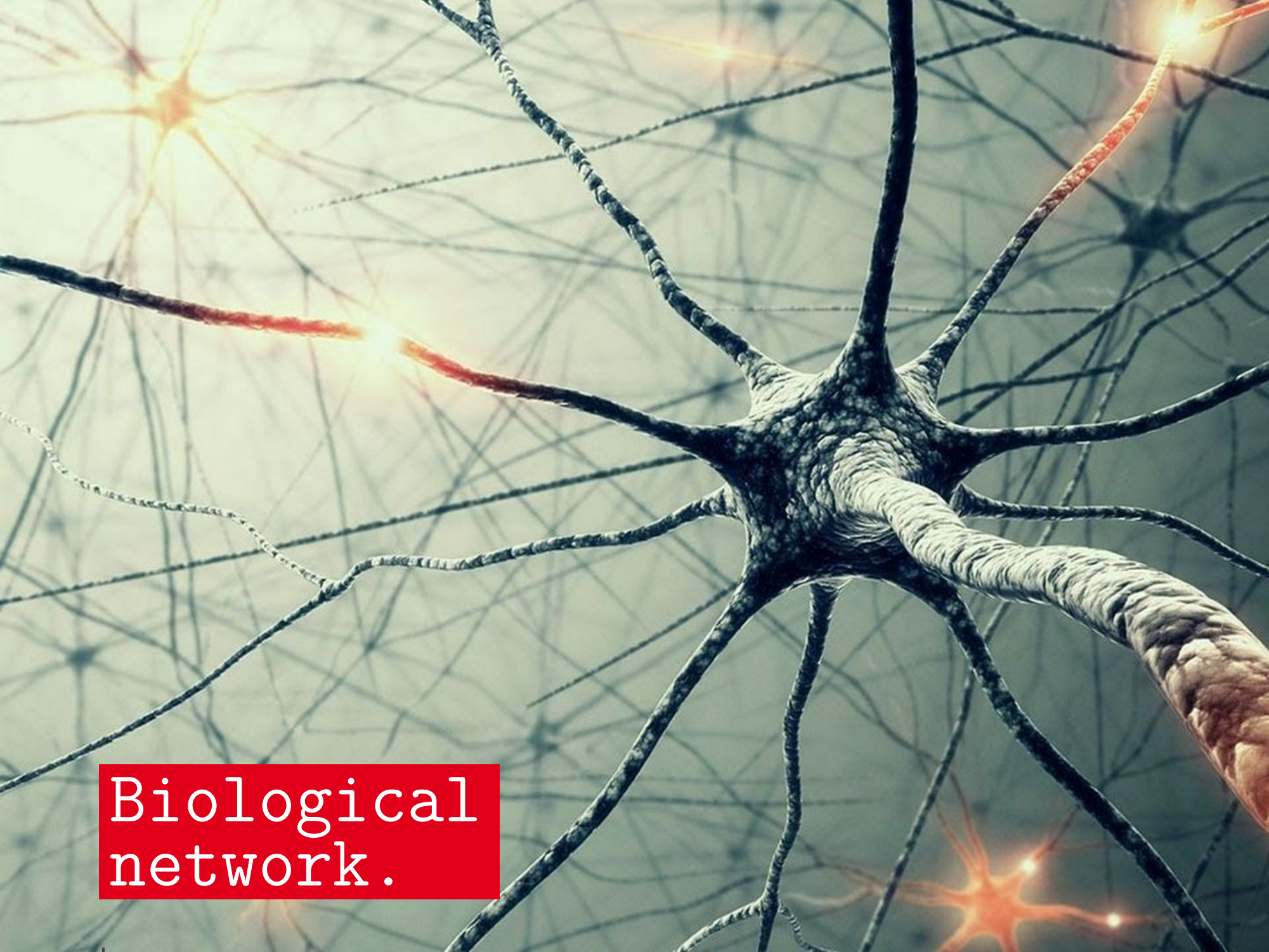
MIDDLE EAST

LATIN AMERICA

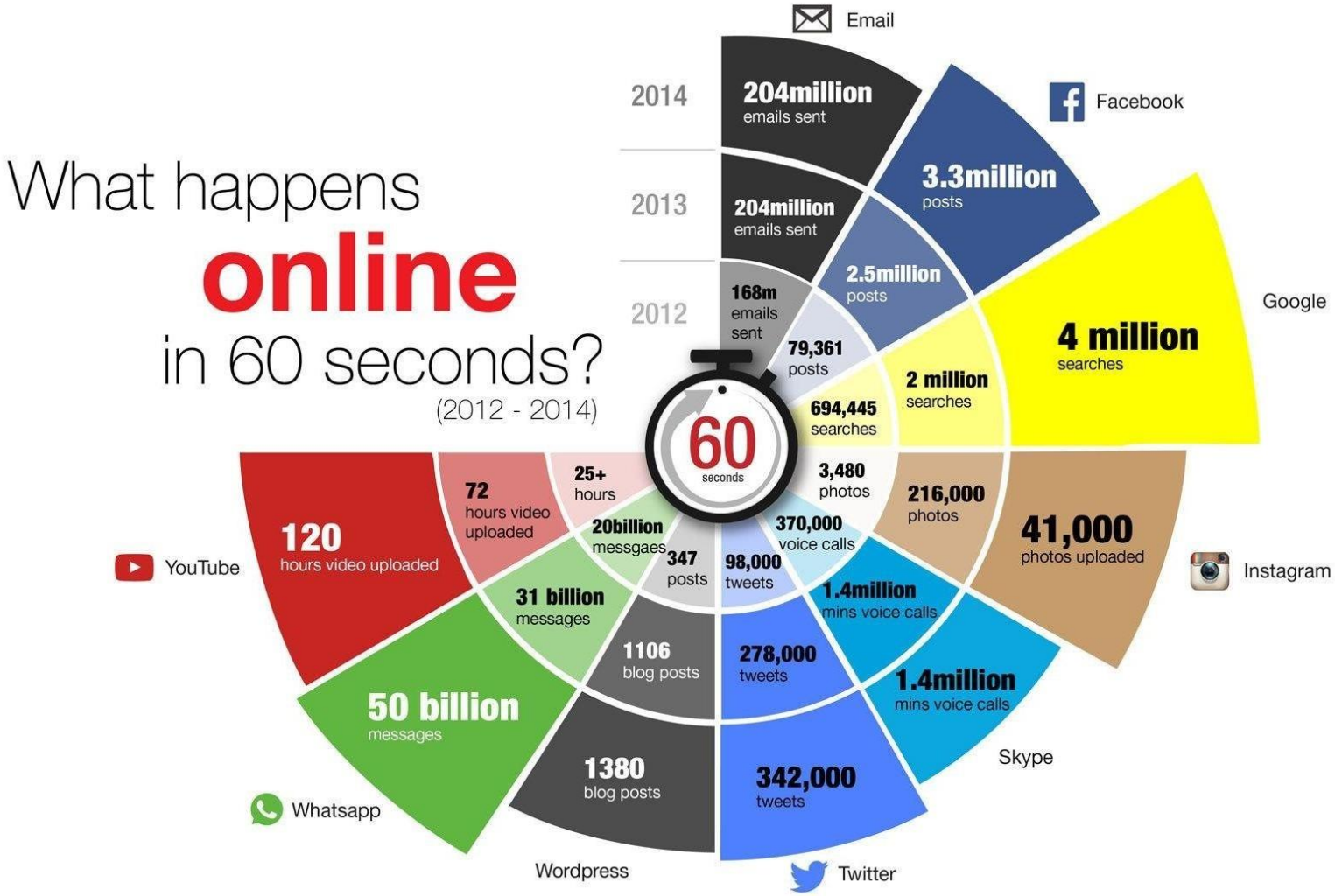
Pacific islands

OCEANIA

Data: openflights.org


Biological
network.


Brain
network.



Article on Business Insider Australia, 2015

Data generated every 60s.

How are COMPLEX & LARGE-SCALE NETWORKS?

What are Community Networks?

. **Network clusters or Communities.**

→ Cluster: A set of appropriately connected nodes.

→ Community: Nodes with a shared latent property.

Many reasons and benefits for why communities form.

→ In World Wide Web link farms can be easily detected with community detection. A link farm is a group of websites which hyperlink to every other site in the group.

→ In network community of customer with similar interest can be used to make recommender system to enhance the business

→ Communities will help us to understand the structure of social network. Communities clarify the properties and functions of network.

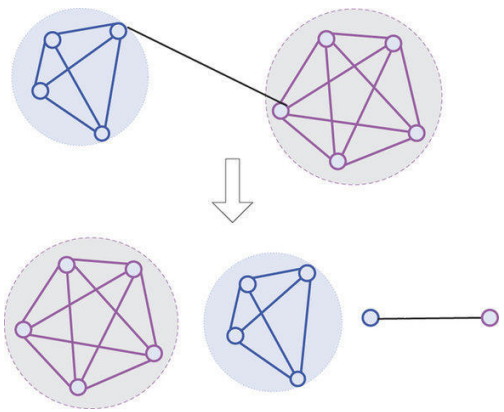
→ In biological network communities helps us to understand basic mechanisms which control normal cellular processes.

Finding communities

Q: **How and why do communities form?**

A: **Find weak ties and identify communities**

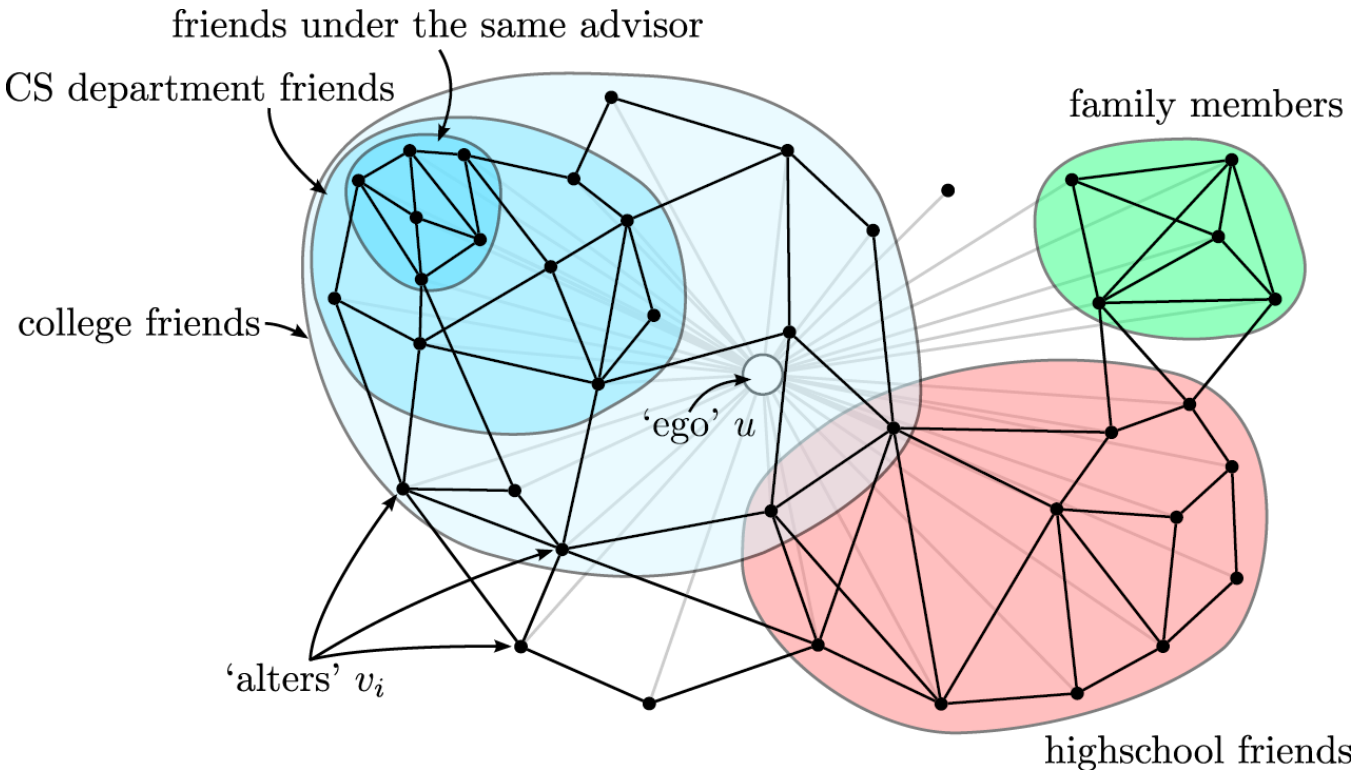
Givan-Newman's betweenness centrality, modularity and graph partitioning methods are all based on this idea.



Q: **Can It find overlapping communities and communities in large-scale network?**

A: **No!It dont detect overlapping communities and often has Big O$((n+m)mn^2)$**

Overlapping communities

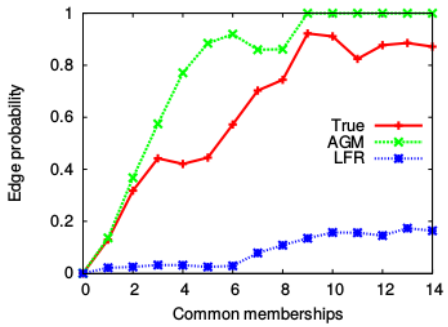


[McAuley, Leskovec: Discovering social circles in ego networks, 2012]

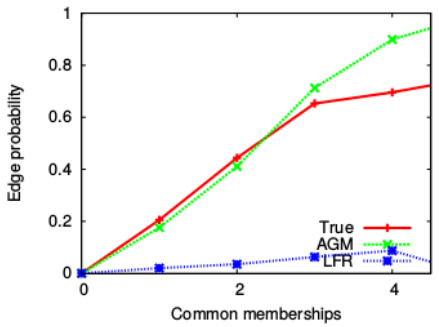
Edge Probability

Nodes u and v share k groups.

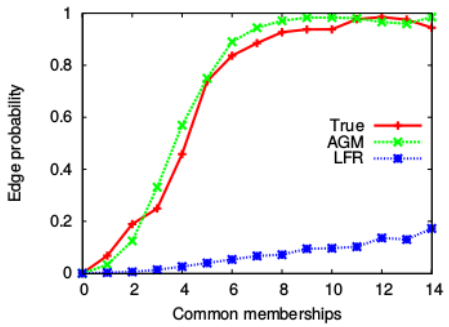
What is edge prob $P(edge|k)$ as a func of k?



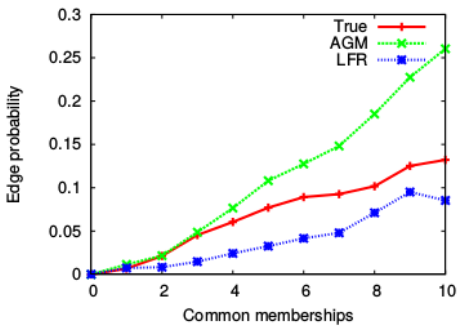
(a) LiveJournal



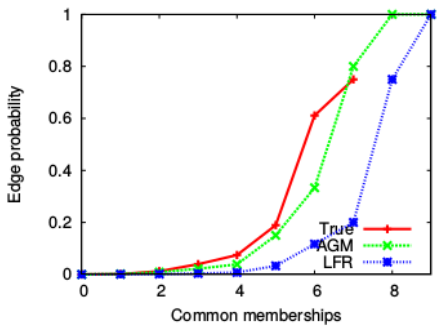
(b) Friendster



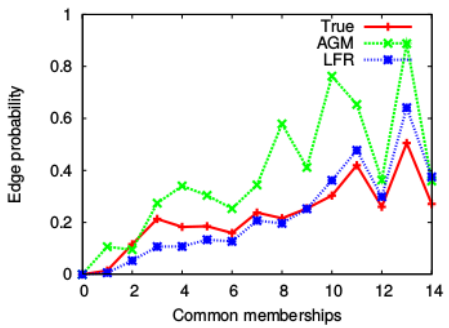
(c) Orkut



(d) YouTube



(e) DBLP



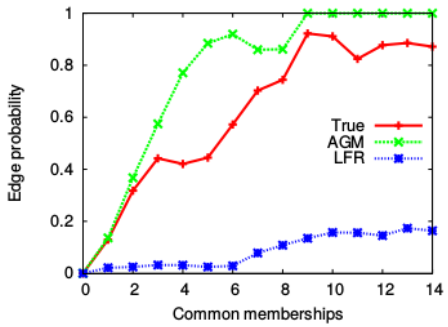
(f) Amazon

Edge Probability

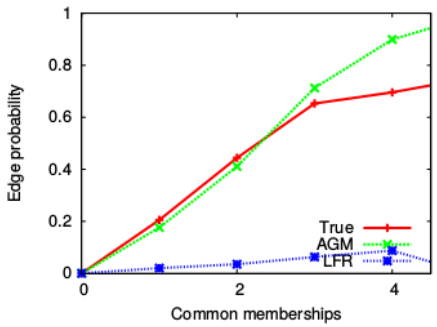
Nodes u and v share k groups.

What is edge prob $P(edge|k)$ as a func of k?

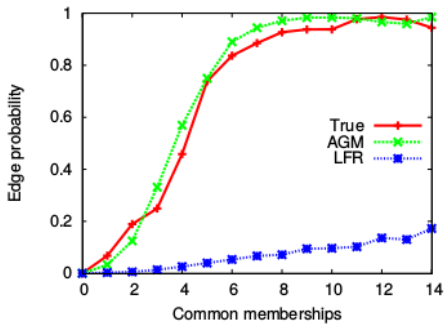
Overlaps are DENSER!



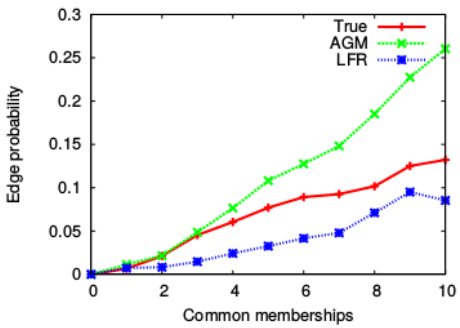
(a) LiveJournal



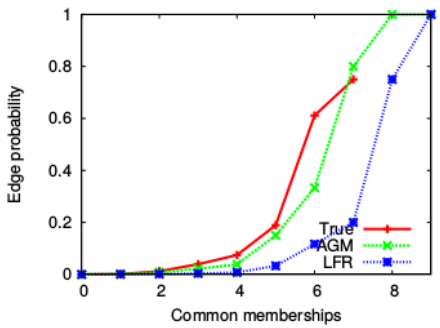
(b) Friendster



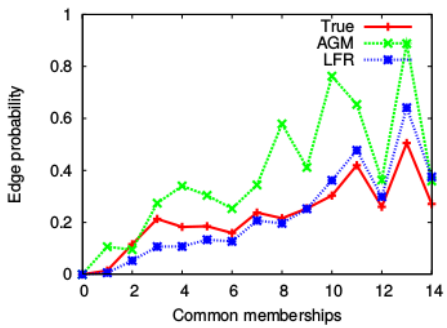
(c) Orkut



(d) YouTube

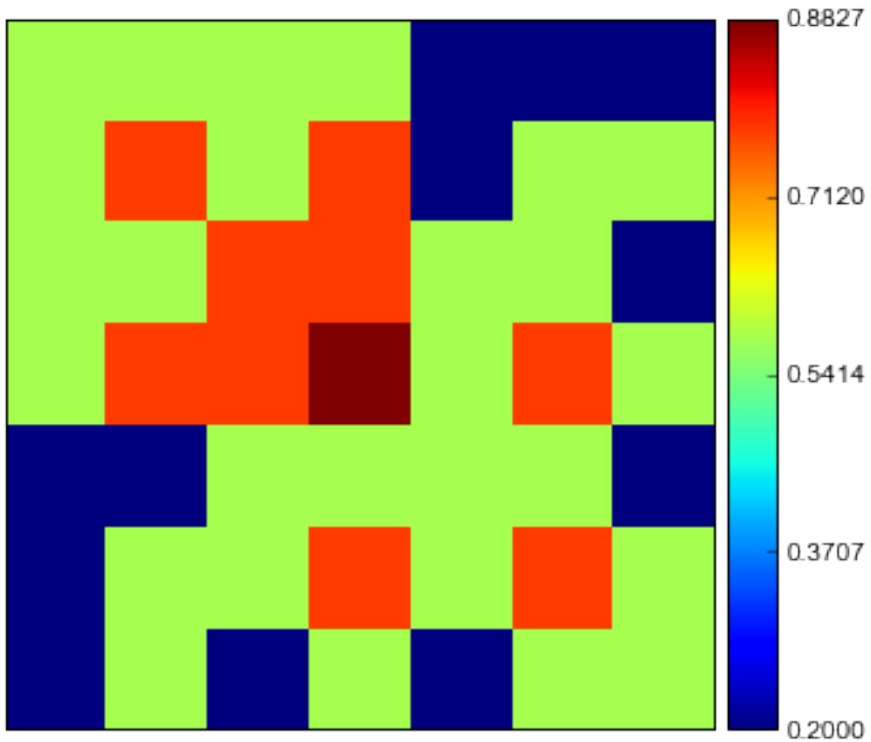


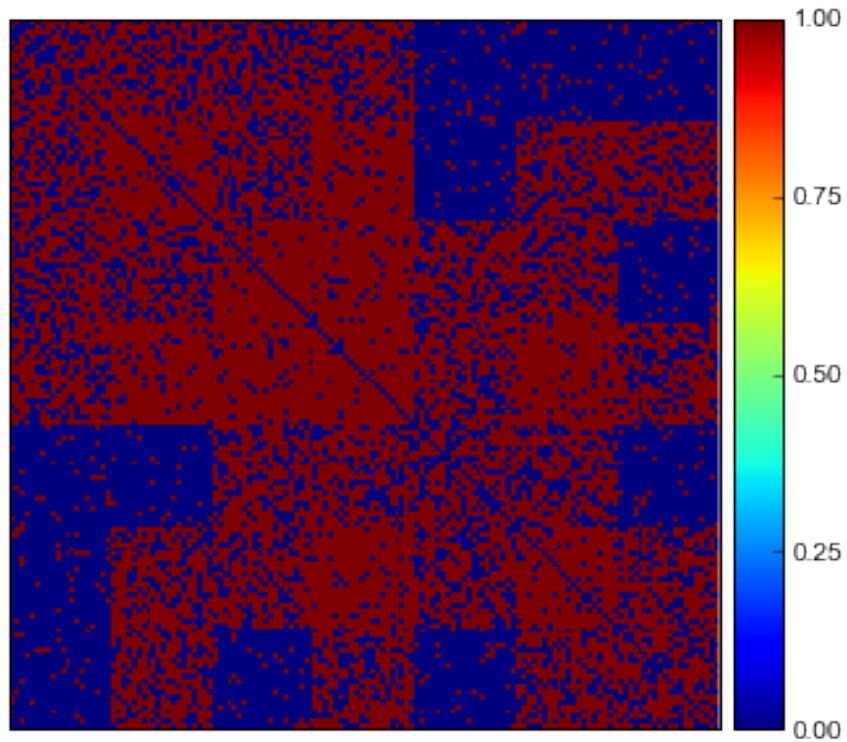
(e) DBLP



(f) Amazon

Overlaps are DENSER!



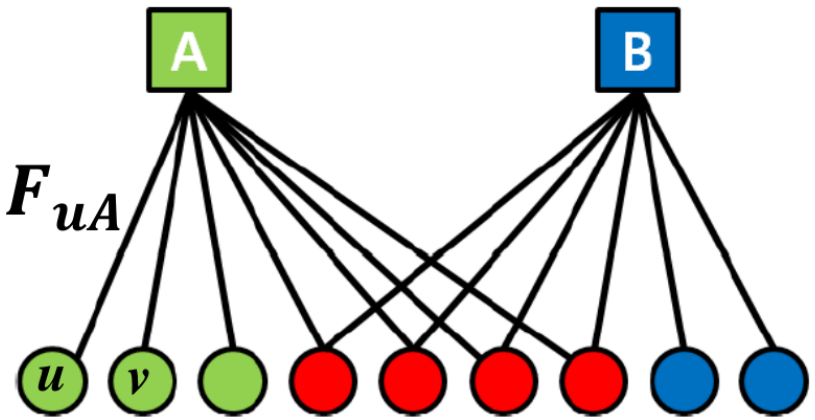


[Stanford University: http://snap.stanford.edu/agm]

BIGCLAM

BigCLAM

Memberships have strengths



[Stanford University: http://snap.stanford.edu/agm]

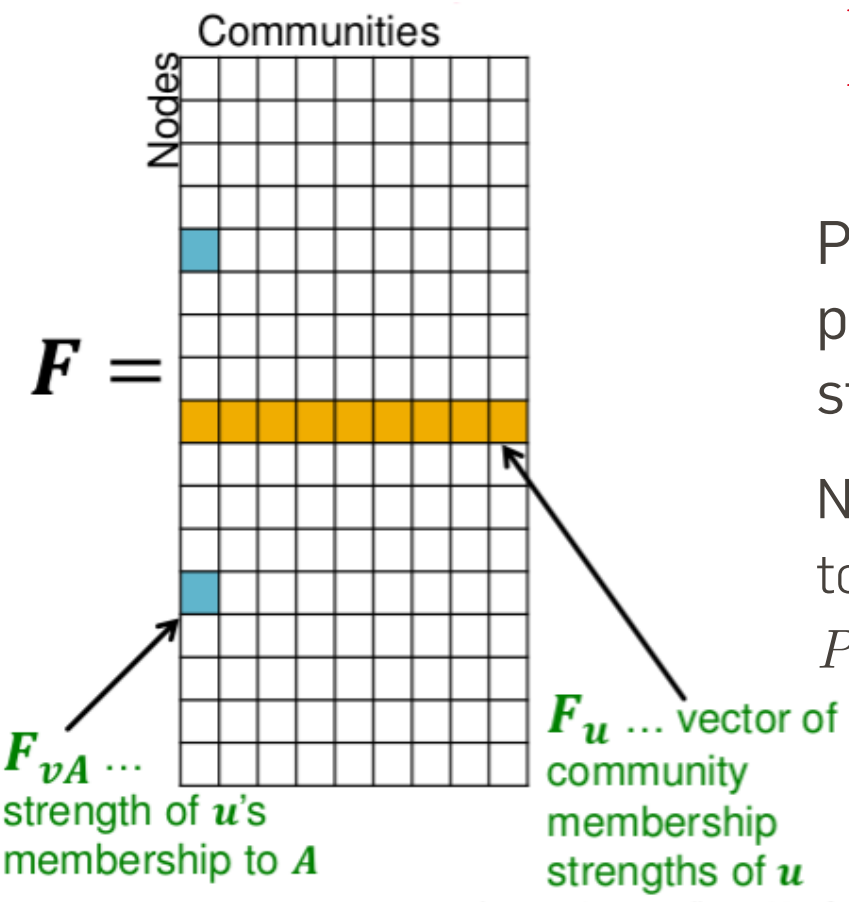
→ F_{uA}: The membership strength of node u to community A ($F_{uA} = 0$: no membership)

→ Each community A links nodes independently:

$$P_A(u, v) = 1 - \exp(-F_{uA}.F_{vA})$$

Factor matrix F

Community membership strength matrix F



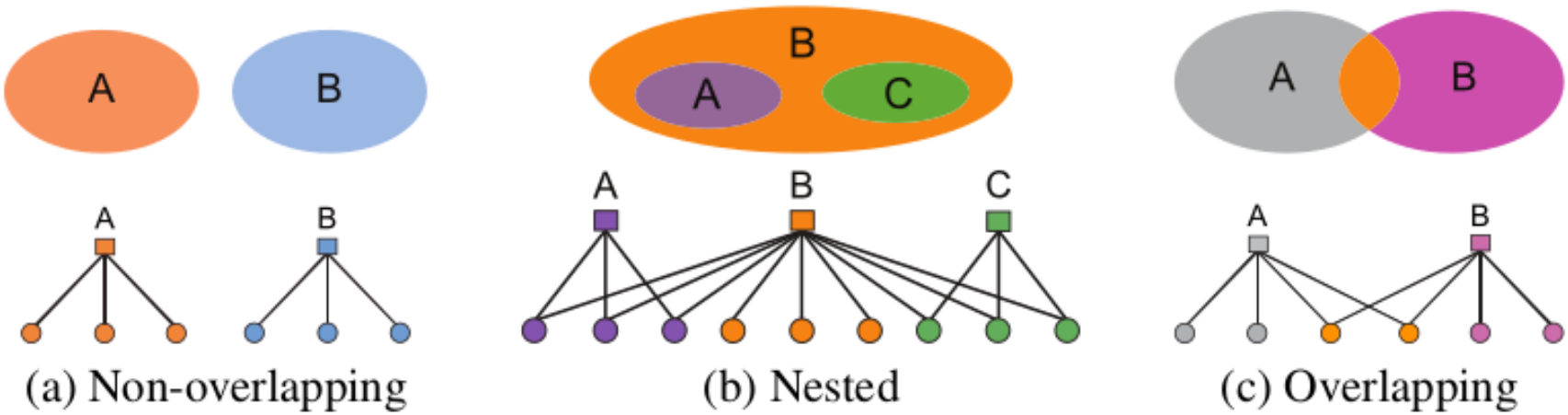
$$P_A(u,v) = 1 - \exp(-F_{uA}.F_{vA})$$
$$P(u,v) = 1 - \exp(-F_u.F_v^T)$$

Probability of connection is proportional to the product of strengths.

Notice: If one node doesn't belong to the community ($F_{uC=0}$) then $P(u,v) = 0$

BigCLAM: Flexibility

BigCLAM can express variety of network structures:
Non-overlapping, Nested, Overlapping.



[Stanford University: http://snap.stanford.edu/agm]

Maximum Likelihood Estimation of Graph

Task: Given a network $G(\mathcal{V}, \mathcal{E})$, estimate F

With F, the **likelihood** of the graph is computed by

$$P(G|F) = \prod_{(u,v)\in\mathcal{E}} p(u,v) \prod_{(u,v)\neq\mathcal{E}} (1 - p(u,v)). \quad (1)$$

Where $p(u,v) = 1 - \exp(-F_u.F_v^T)$

Goal: Find F that maximizes the log-likelihood

$$\hat{F} = \arg\max_{F\geq 0} l(F) \quad (2)$$

Where $l(F) = logP(G|F)$

$$l(F) = \sum_{(u,v)\in\mathcal{E}} log(1 - \exp(-F_u F_v^T)) - \sum_{(u,v)\neq\mathcal{E}} \exp(F_u F_v^T) \quad (3)$$

Convex optimization

Maximization as convex optimization problem. Equation $l(F)$ can be broken down into one sub-problem per $u \in \mathcal{V}$.

$$\hat{F}_u = \arg \max_{F_{uc} \geq 0, c \in \mathcal{C}} l(F_u) \tag{4}$$

So, $l(F_u)$ is computed by

$$l(F_u) = \sum_{v \in \mathcal{N}(u)} log(1 - \exp(-F_u F_v^T)) - \sum_{v \notin \mathcal{N}(u)} F_u F_v^T \tag{5}$$

Where $\mathcal{N}(u) = \{v : v \in \mathcal{V}, (u, v) \in \mathcal{E}\}$ is neighbours of u

Convex optimization

Compute gradient of a single row F_u of F:

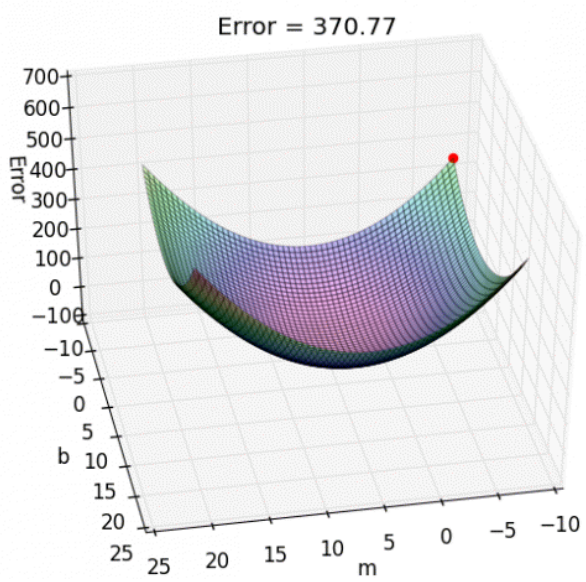
$$\bigtriangledown(l(F_u)) = \frac{\partial l(F_u)}{\partial F_u} = \sum_{v \in \mathcal{N}(u)} F_v \frac{\exp(-F_u F_v^T)}{1 - \exp(-F_u F_v^T)} - \sum_{v \notin \mathcal{N}(u)} F_v \quad (6)$$

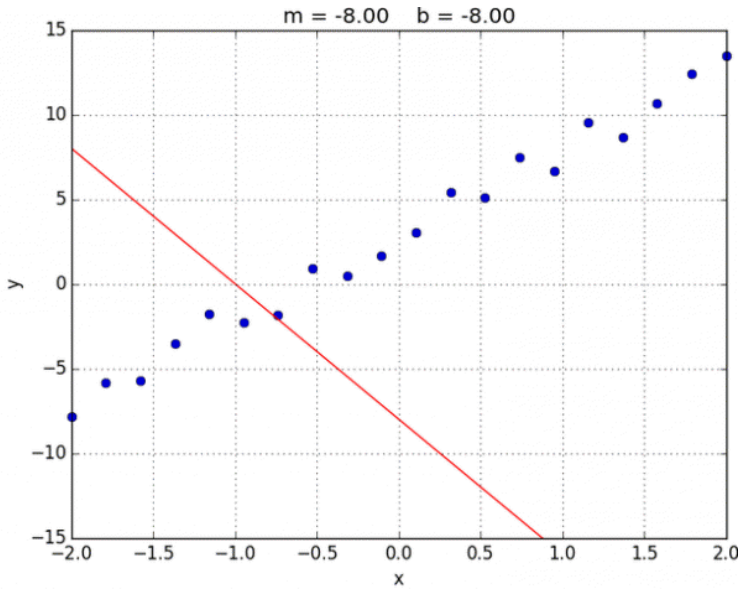
Stochiate Gradient ascent method:

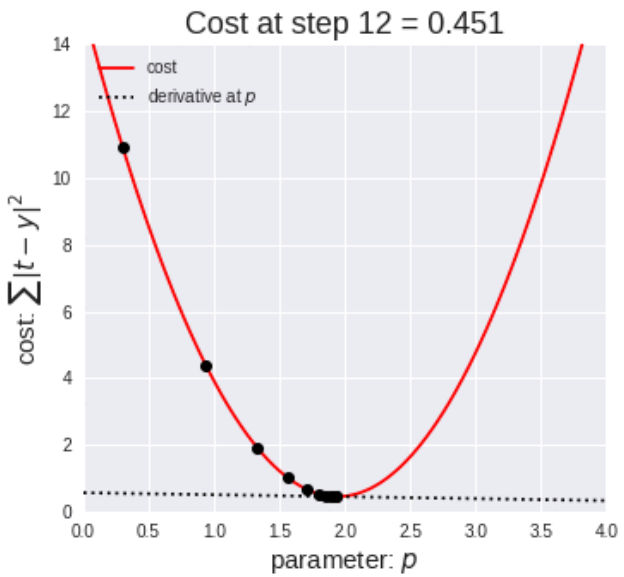
* Iterate over the rows of F
  1. Compute gradient $\bigtriangledown(l(F_u))$ of row u
  2. Update the row $F_u : F_u \leftarrow F_u + \alpha. \bigtriangledown (l(F_u))$
  3. Project F_u back to a non-negative vector: If $F_{u\mathcal{C}} < 0 : F_{u\mathcal{C}} = 0$

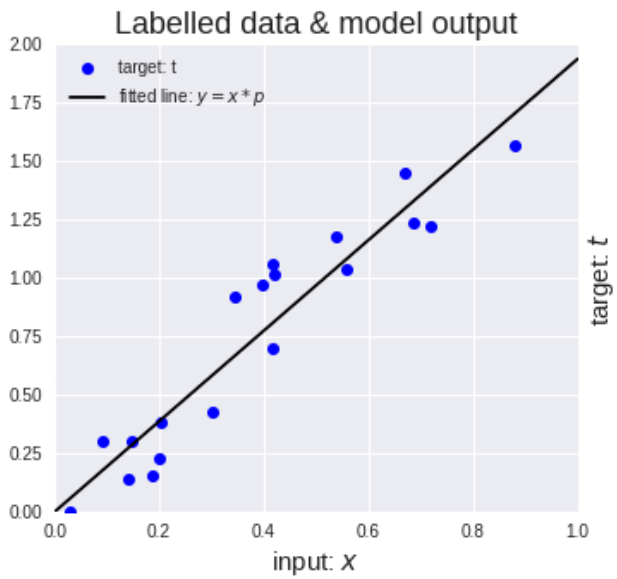
This is slow! Computing $\bigtriangledown(l(F_u))$ take linear time with $O(|\mathcal{V}|)$.

Gradient descent method

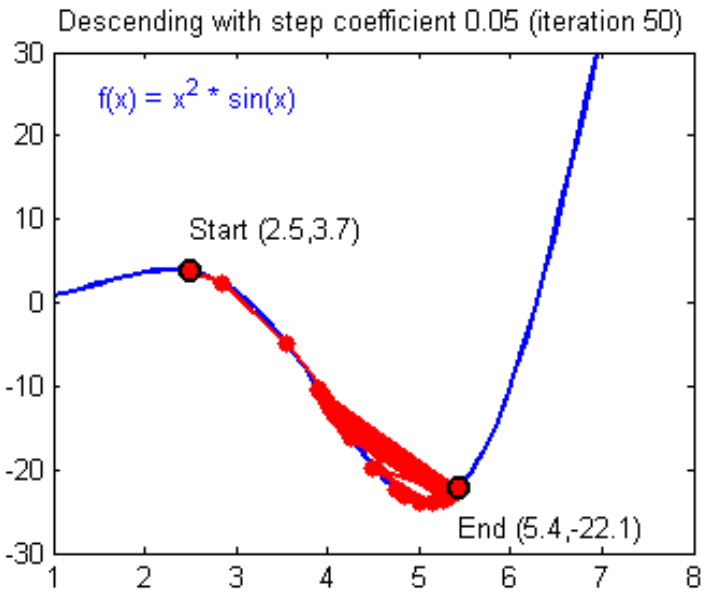




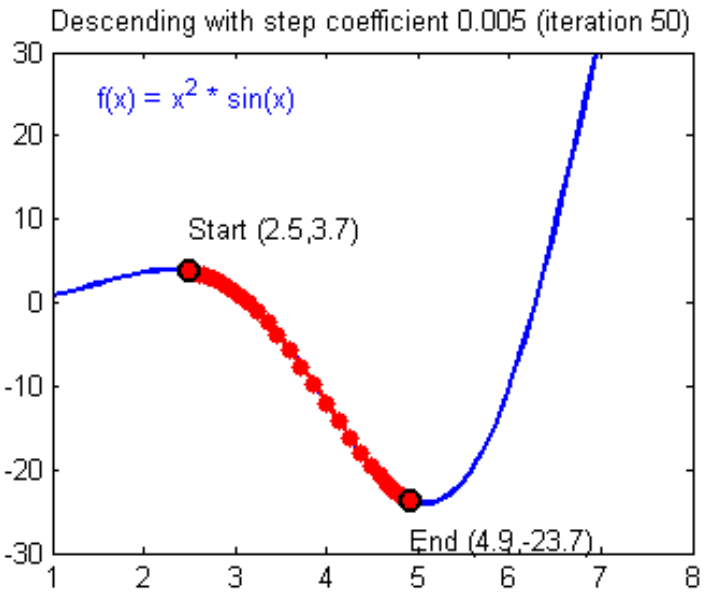




Learning rate



Large learning rate α



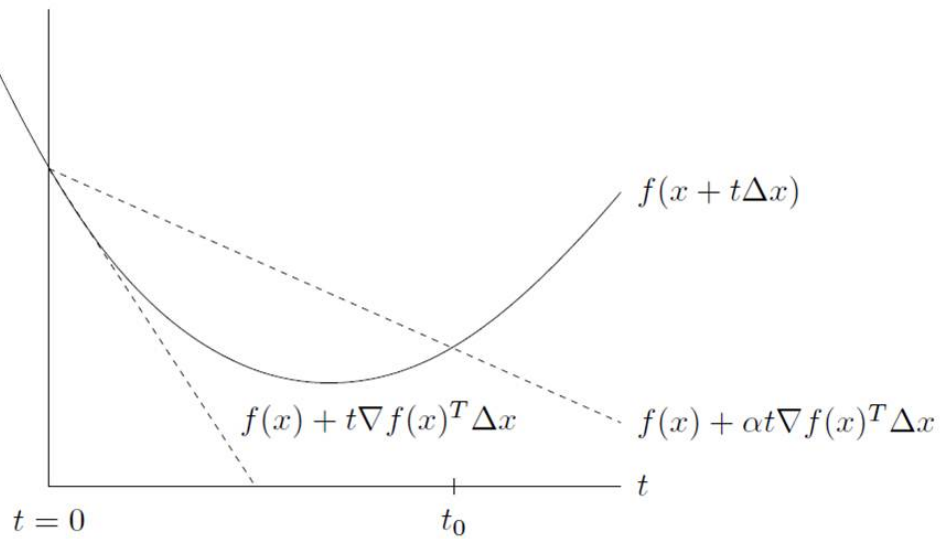
Small learning rate α

Need good way to choose and adjust a "learning rate"!

Backtracking line search

Algorithm 2 Backtracking line search.

Given a starting point $\triangle\theta$ for f at $\theta \in \mathbf{dom} f, \gamma \in (0, 0.5), \beta \in (0, 1)$
While $f(\theta + \alpha \triangle \theta) > f(\theta) + \gamma\alpha\nabla f(\theta)^T \triangle \theta$
$\alpha = \beta\alpha$



MBSGD method

Algorithm 4 Mini-batch Stochastic gradient descent method.

Given a starting point $\theta = \{\theta_1, \ldots, \theta_n\} \in \mathbf{dom} f$ and $0 \leq k \leq N$
repeat
    1. Shuffle training data
    for $i = 1; i \leq N; i+ = k$ **do**
        1. $\Delta\theta := -\nabla_\theta f\left(\theta; x^{i:i+k}, y^{i:i+k}\right)$.
        2. Choose step size α via backtracking line search
        3. *Update.* $\theta := \theta + \alpha\Delta\theta$
    end for
until Stopping criterion is satisfied.

Complexity reduction

Two equations below, the time complexity is reduced to $O(|\mathcal{N}(u)|)$ with ($|\mathcal{N}| \ll |\mathcal{V}|$).

$$\sum_{v \notin \mathcal{N}(u)} F_u F_v^T = \sum_{v} F_u F_v^T - F_u F_u^T - \sum_{v \in N(u)} F_u F_v^T \tag{7}$$

Where $N(u) = u \cup \mathcal{N}(u)$ is neighbourbood of u

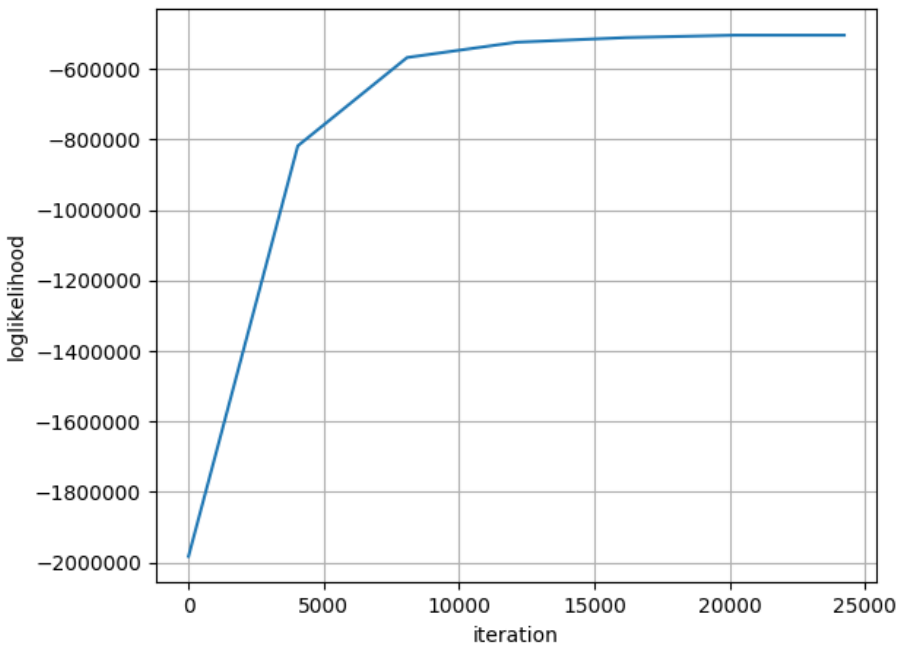
$$\sum_{v \notin \mathcal{N}(u)} F_v = \left(\sum_{v} F_v \right) - F_u - \left(\sum_{v \in \mathcal{N}(u)} F_v \right) \tag{8}$$

Stopping condition

With $l'(F_u)$ is the log-likelihood of F for u after update by gradient ascent method. The iterative ascent process terminates if:

$$\frac{l'(F_u) - l(F_u)}{l(F_u)} < 0.0001, \forall u \in \mathcal{V} \tag{9}$$

the log-likelihood increases by less than $0.01\% \forall u \in \mathcal{V}$



Initialization of F

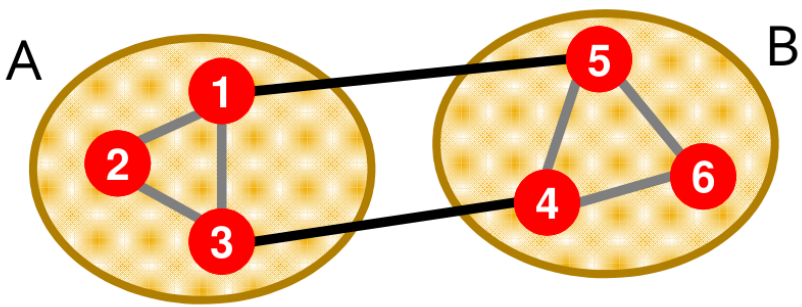
Let F be an affiliation weights matrix. The algorithm initialises F as follows:

$$F_{(u')(N(u))} = \begin{cases} 1 & \text{if } u' \in N(u) \text{ and } N(u) \text{ is a locally} \\ & \text{minimal neighbourhood of } u \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

Where $N(u)$ is said to be locally minimal if it has lower conductance than all $N(v), v \in \mathcal{N}(u)$.
$\varphi(N(u)) \leq \varphi(N(v)), \forall v \in \mathcal{N}(u)$.

$$\varphi(\mathcal{S}) = \frac{\sum_{i \in \mathcal{S}, j \notin \mathcal{S}} A_{ij}}{min(\mathcal{A}(\mathcal{S}), \mathcal{A}(\bar{\mathcal{S}}))} \quad (11)$$

Where $\mathcal{A}(\mathcal{S}) = \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{V}} A_{ij}$



Determining community affiliations

Let $G(\mathcal{V}, \mathcal{E})$, F be the most likely affiliation weights matrix under G. $u \in \mathcal{V}$ as a member of $c \in \mathcal{C}$ if:

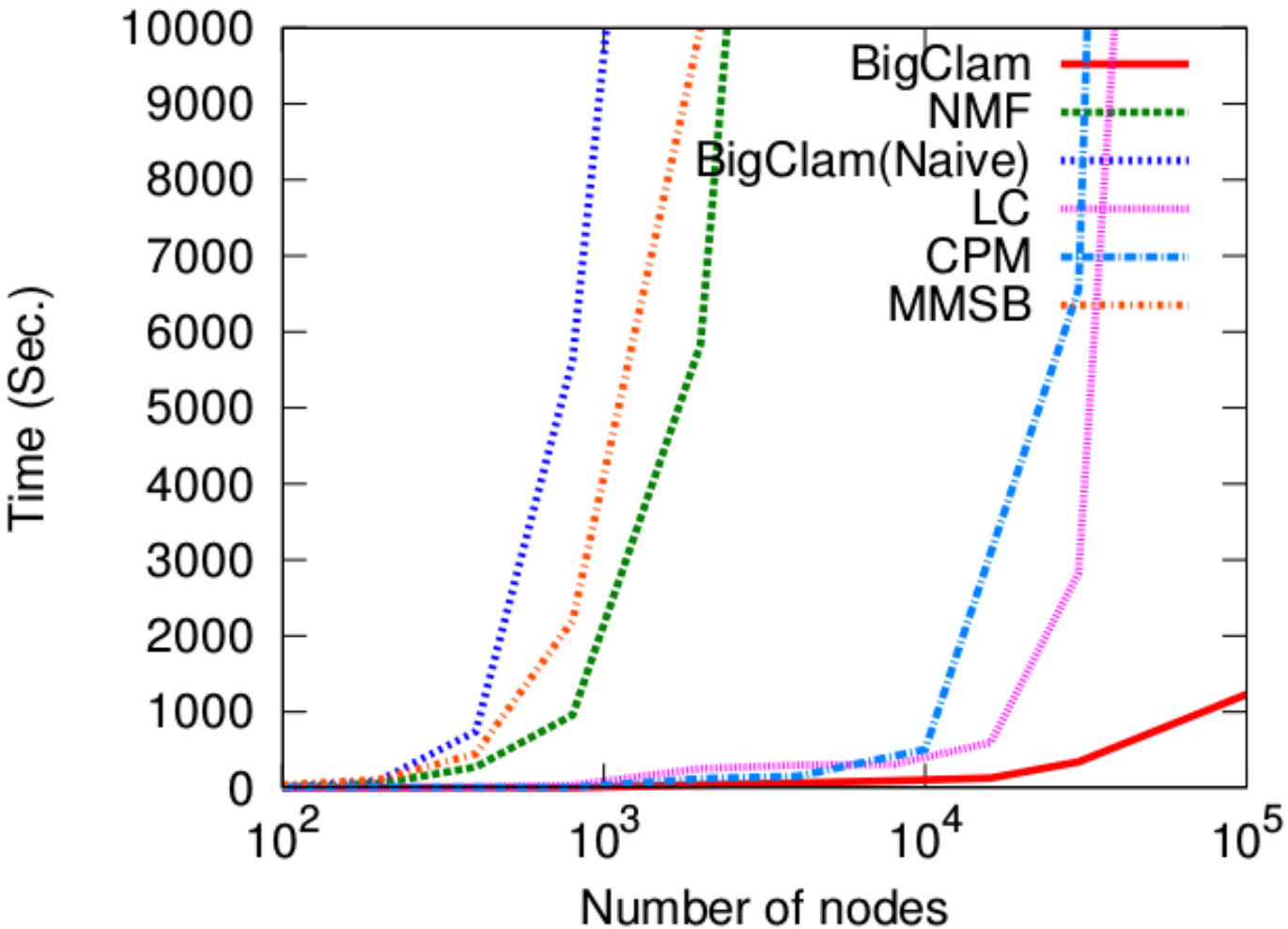
$$F_{uc} \geq \delta = \sqrt{-\log(1-\epsilon)} \tag{12}$$

Where $\epsilon = \frac{2|\mathcal{E}|}{|\mathcal{V}|(|\mathcal{V}|-1)}$

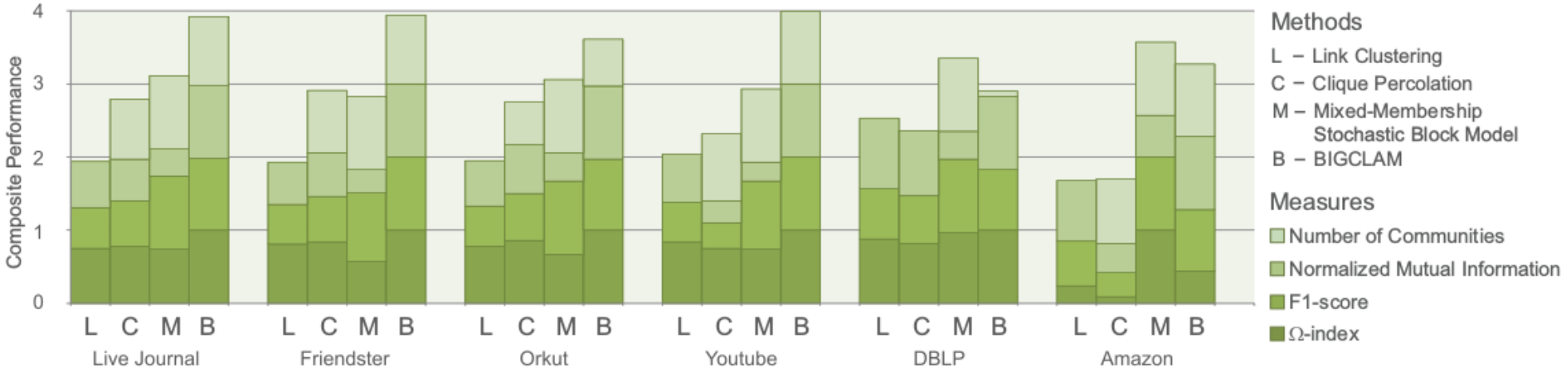
Determining number of communities

1. Select 20% node pairs randomly form G as hold out set H
2. For every K as a candidate on number of community:
   - → Run community detection algorithm, with K as the number of communities. $K \in [minCom; maxCom]$ with stepsize: $\alpha = \exp\left(\log\left(\frac{maxCom}{minCom}\right) \times \frac{1}{divCom}\right)$
   - → Evaluate the likelihood of affiliation weights with respect to H,$l_k(F_H)$
3. Obtain the most likely number of communities $\hat{K} = \arg\max_K l_k(F_H)$

Experiments



[Stanford University: http://snap.stanford.edu/agm]



Name	Type	Nodes	Edges	Communities	Description
com-LiveJournal	Undirected, Communities	3,997,962	34,681,189	287,512	LiveJournal online social network
com-Friendster	Undirected, Communities	65,608,366	1,806,067,135	957,154	Friendster online social network
com-Orkut	Undirected, Communities	3,072,441	117,185,083	6,288,363	Orkut online social network
com-Youtube	Undirected, Communities	1,134,890	2,987,624	8,385	Youtube online social network
com-DBLP	Undirected, Communities	317,080	1,049,866	13,477	DBLP collaboration network
com-Amazon	Undirected, Communities	334,863	925,872	75,149	Amazon product network

[Stanford University: http://snap.stanford.edu/agm]

GRID COMPUTING

Grid computing

is a type of parallel and distributed system or a computer network in which each computer's resources are shared with every other computer in the system.

Benefits & Challenges

Many **benefits** and **challenges** of grid computing.

Benefits

- → Can solve larger, more complex problems in a shorter time
- → Easier to collaborate with other organizations
- → Make better use of existing hardware
- → etc.

Challenges

- → Some messages can be lost in the network system
- → Security problem due to sharing
- → Schedule and optimization problem
- → etc.

Apache Hadoop

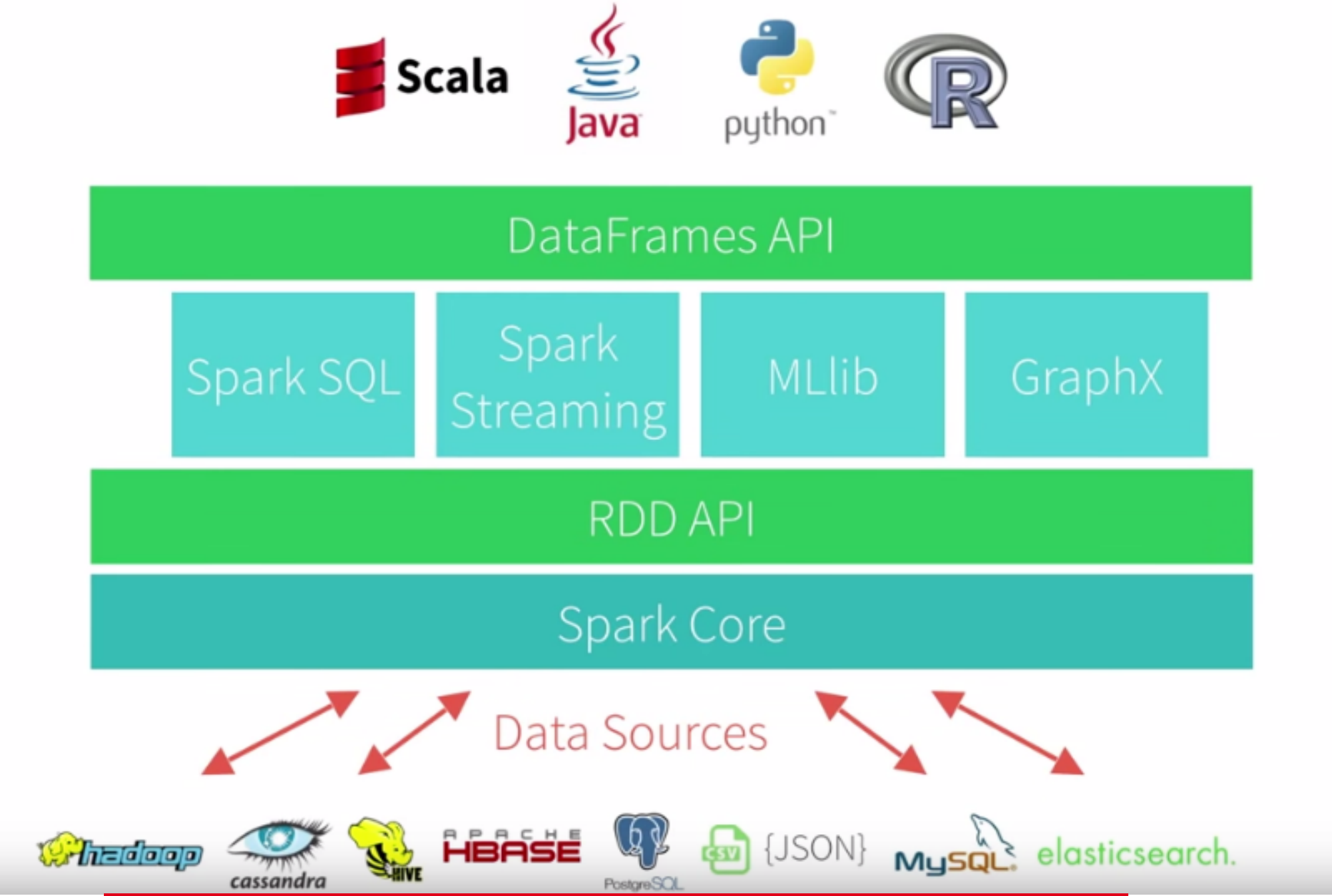
Apache Hadoop is a framework that allows for distributed processing of large data sets across clusters of commodity computers using a simple programming model.

Apache Spark

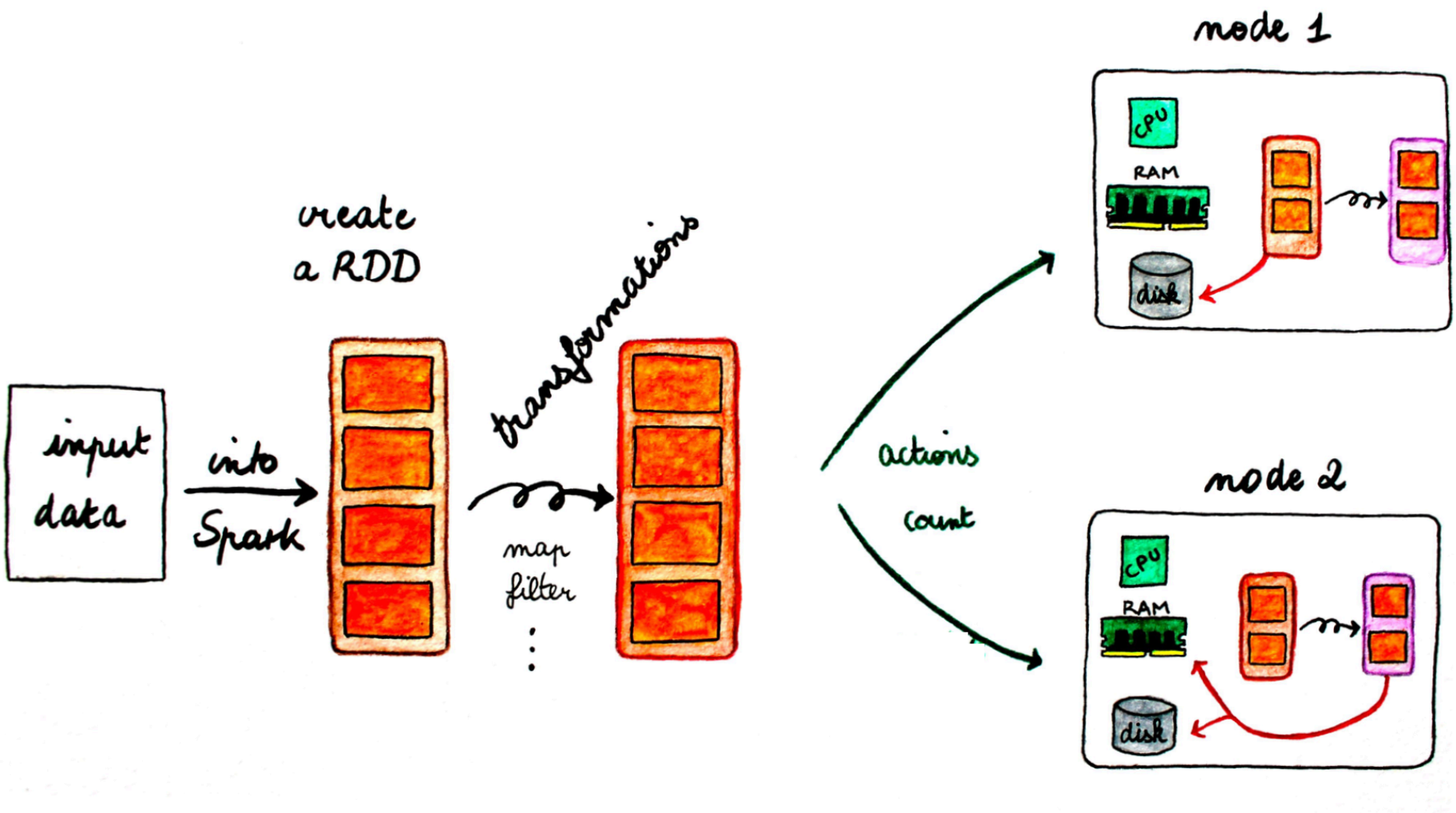
is an open-source luster-computing framework for real time processing developed by the Apache Software Foundation.

Spark provides an interface for programming entire clusters with implicit data parallelism and fault-tolerance.

It was built on top of Hadoop MapReduce and it extends the MapReduce model to efficiently use more types of computations.



Apache Spark Architecture



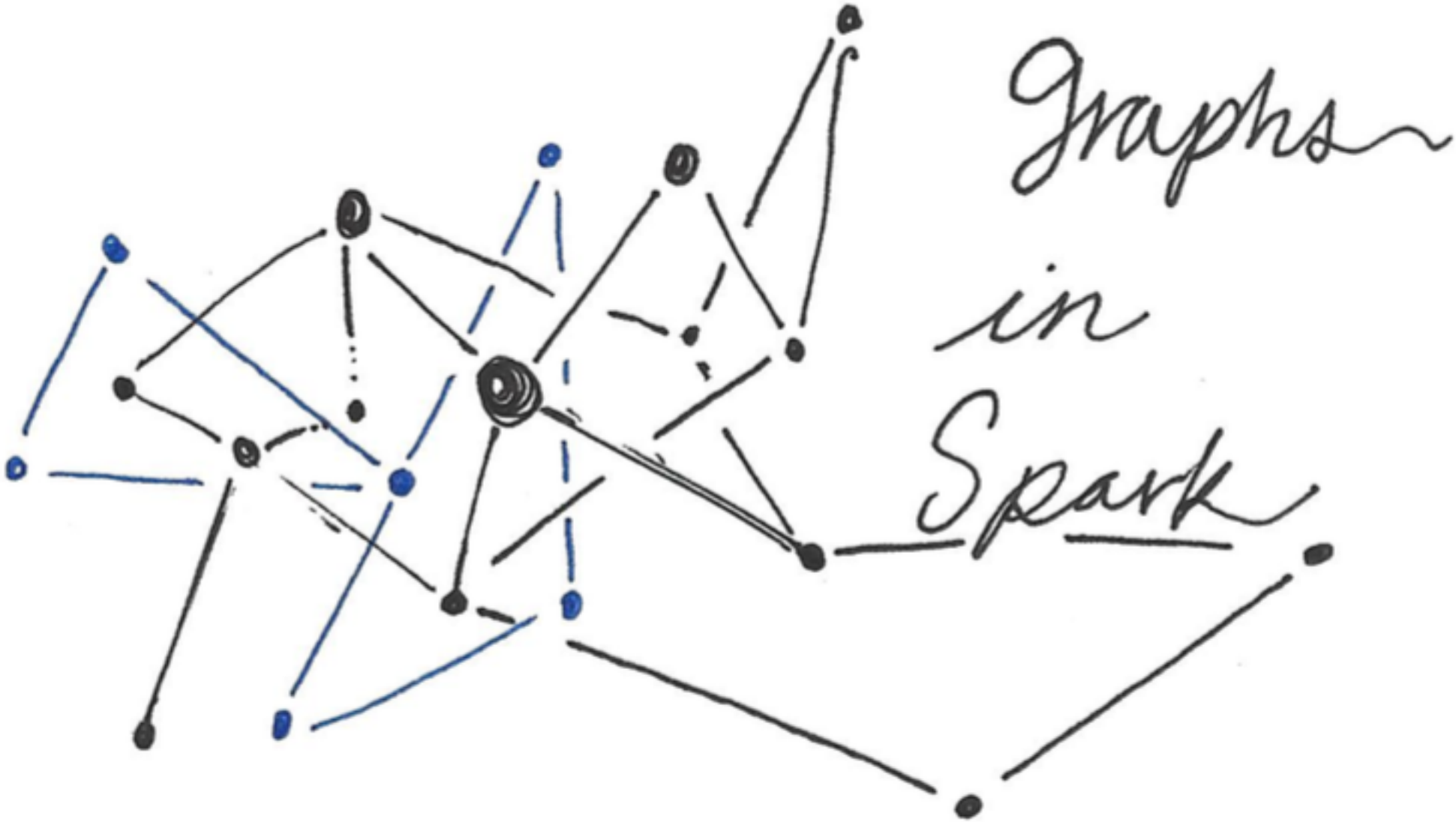
Resilient Distributed Datasets

Transformations

map(), mapPartitions(), filter(), ...

Actions

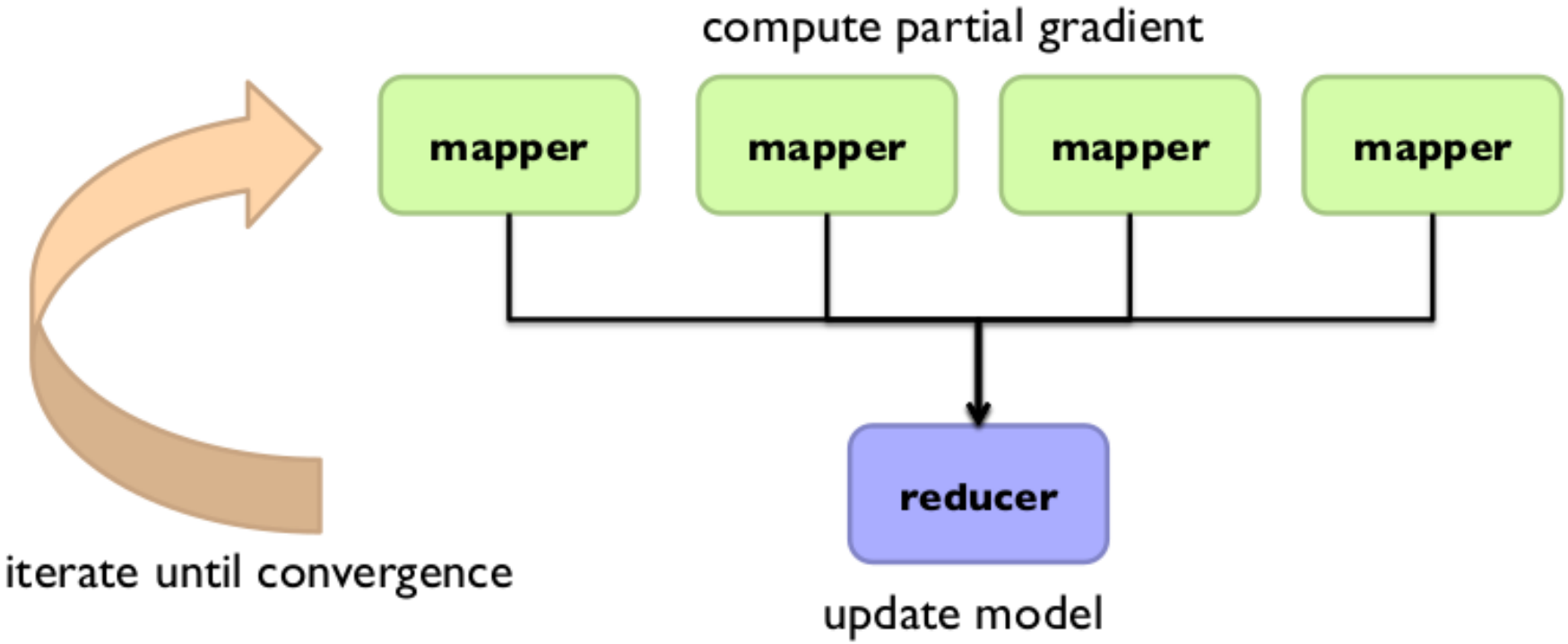
count(), collect(), sortBy(), reduce(), ...



Graphx

GraphX is Apache Spark's API for graphs and graph-parallel computation.

$$\theta^{(t+1)} \leftarrow \theta^{(t)} - \gamma^{(t)} \underbrace{\frac{1}{n} \underbrace{\sum_{i=0}^{n} \nabla \ell(f(\mathrm{x}_i; \theta^{(t)}), y_i)}_{\text{mappers}}}_{\text{single reducer}}$$



Map-Reduce Model

The information of grid

Specs	Number	Role
OS: Ubuntu 16.04 LTS HDD: 95 GB RAM: 4 GB CPU: intel Core i5-4590U CPU $3.30 GHz$ x 4	1	master
OS: Ubuntu 16.04 LTS HDD: 95 GB RAM: 4 GB CPU: intel Core i5-4590U CPU $3.30 GHz$ x 4	9	workers

Experiment and Performance

Table: The table shows detail network is used to experiment.

Name	Nodes	Edges
Ego-Facebook	4,039	88,234
My ego-Facebook	20,812	27,282
com-Amazon	334,863	925,872
com-Youtube	1,134,890	2,987,624

Evaluation

Average $F1$ score: To compute the F1 score, we need to determine which $C_i \in \mathcal{C}$ corresponds to which $C'_i \in \mathcal{C}'$. F1 score is computed:

$$\frac{1}{2}\left(\frac{1}{K}\sum_{C_i \in \mathcal{C}} F1(C_i, C'_{g(i)}) + \frac{1}{K'}\sum_{C'_i \in \mathcal{C}'} F1(C_{g'(i)}, C'_i)\right)$$

Where the best matching g and g' is defined as follows:

$$g(i) = \arg\max_j F1(C_i, C'_j), g'(i) = \arg\max_j F1(C_j, C'_i)$$

Name	F1 score
com-Amazon	0.49
com-Youtube	0.42

References

1. David F. Gleich and C. Seshadhri. Neighborhoods are good communities.CoRR abs/1112.0031, 2011.
2. Jaewon Yang and Jure Leskovec. Overlapping community detection at scale: a nonnegative matrix factorization approach. In Proceedings of the sixth ACM international conference on Web search and data mining, pages 587–596. ACM, 2013
3. John Aldrich et al. Ra fisher and the making of maximum likelihood 1912-1922. Statistical Science, 12(3):162–176, 1997

DEMO & CONCLUSION

Main contributions

- → Overlapping community detection algorithm using BigCLAM model has successfully implemented.
- → This algorithm able to detect community in large-scale on Apache Spark.
- → MBSGD method and backtracking line search method have proposed working better than the author's methods on computing grid.
- → This algorithm able to detect communities in the large-scale network.

Future work

- → Research some techniques and redesign code to improve the performance algorithm on computing grid system and accuracy of algorithm.
- → Design and organise cluster computer to create a supercomputing grid system.
- → Apply this algorithm to solve real-life problems.
  - → Build recommendation system in social network in education for ThangLong University project. It will increase the student's performance in study and activities.
  - → Build effective advertising campaign by understanding structure social.
  - → etc.

Thank you!

Thank you for watching and I hope you enjoyed my presentation. If you have any questions, I'll be happy to answer them.