Massive-Scale Graph Analysis

Vinay Setty Siaram Gurajada Mohamed Yahya Stephan Seufert

[vsetty, gurajada, myahya, sseufert]@mpi-inf.mpg.de

Agenda

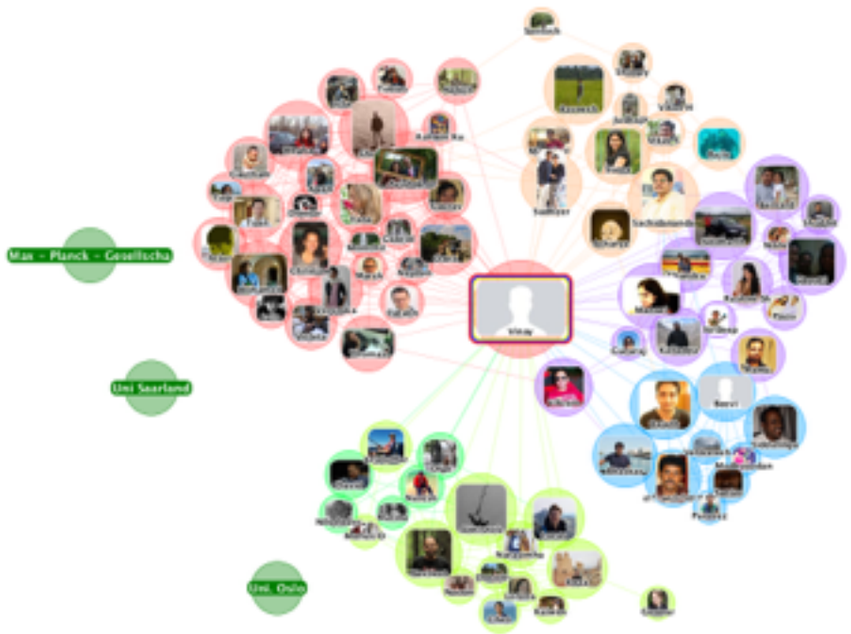
- Introduction
- Topics
- Seminar rules and requirements
- Registration

Overview

- Fridays 10:00 - 11:30 in E 1.4 / R 021
- Today (24th April 2015) kickoff meeting
- 8th May 2015 to 10th July 2015 your seminars
- No holidays on seminar days!

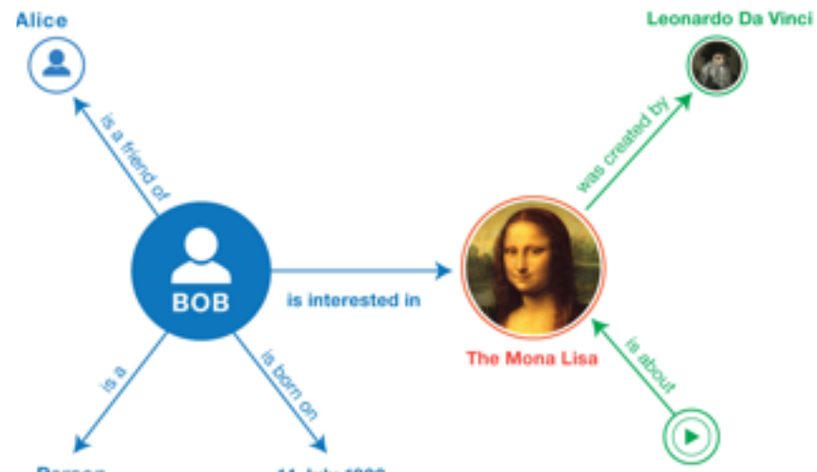
data
performance
algorithms
large
computing
processing
distributed
edges
systems
streams
stream
partitioning
computation
networks
mining
paper
streaming
several
high
MapReduce
subgraph
communication
efficient
existing
describe
network
including
many
number
show
graph
PageRank
patterns
large-scale
machines
heuristics
experiments
size
RDF
applications
multiplication
web
GIM-V
various
like
single
problem
certain
used
vertex-centric
linear
system
approach
GraphChi
parallel
social
introduce
pattern
search
massive
using
dense
cost
achieving
best
scale
PowerGraph
components
edge
algorithm
structure
time
nodes
entities
knowledge
efficiently
Web
small
novel
present
problems
study
highly
query
general
computational
projections
design
important
management
implementation
Trinity
frameworks
GraphLab
etc
new
cluster
techniques
queries
called
results
based
performing
However
updates
natural
scalable
memory
framework
propose
real-world
analysis
TLAV
main
one
users
magnitude
abstraction
billion
methods
scalability
sketches
standard
challenges
available
implemented
real
version
also
underlying
datasets
address
top
implementations
popular
challenging
setting
synopses
work
library
join
largest
model
source
operations
simple
matrix-vector
Many
well
set

Graphs Are Sexy!



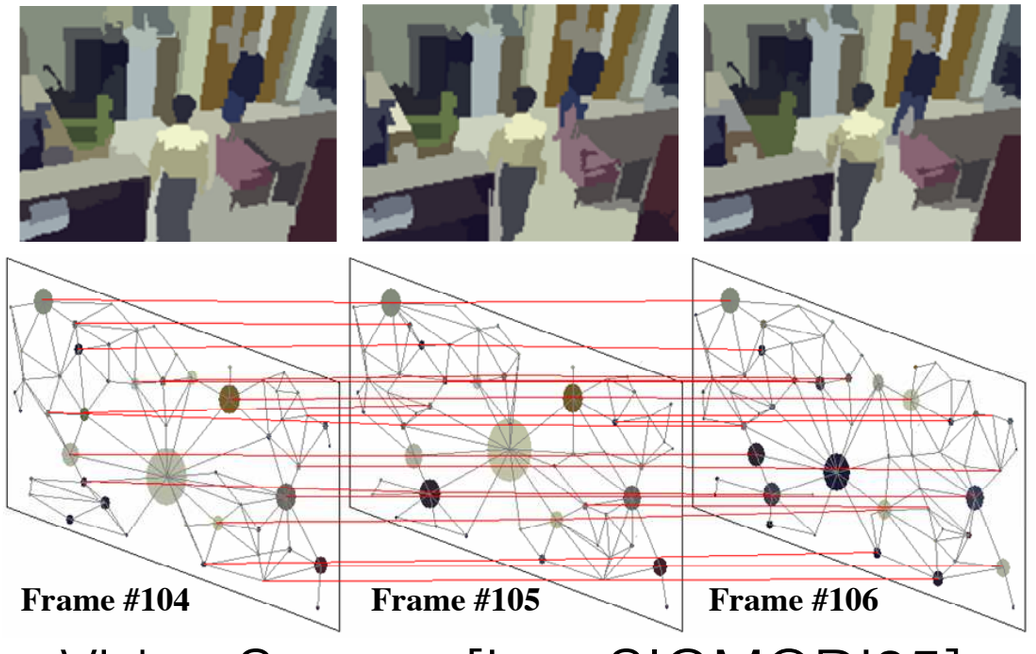
Social Graphs

Protein to Protein Interactions (PPI)



Knowledge Graphs

Co-author, Citation Graphs



Video Scenes [Lee SIGMOD'05]

Massive-Scale Graphs

- Web Graphs: trillions of nodes and edges
    - Clue web: 4.7 billion web pages and 8 billion links
- Social Graphs
    - Facebook: 1.25 billion monthly active users with hundreds of billions of relationships (as of March 31, 2015)
    - Twitter: 288 million monthly active users
- Knowledge Graphs
    - Google knowledge graph: 570 million nodes 18 billion facts
    - Freebase: 1.9 billion triples

Graph Algorithms

- Page Rank
- Shortest paths
- Connected components (strongly and weakly connected components)
- Traversal (BFS, DFS)
- Enumerating triangles (for computing clustering coefficient)
- Graph matching

Topic 1: Map/Reduce for Graphs

- 08/05/2015
- Cohen: **Graph twiddling in a MapReduce world,** Computing in Science & Engineering 2009
- Lin and Schatz: **Design patterns for efficient graph algorithms in MapReduce**, Workshop on Mining and Learning with Graphs 2010
- Additional Reference (to introduce Map/Reduce) Lin et al.: **Data-intensive text processing with MapReduce**, 2010
- **Preferred background**: Databases, Knowledge of Map/Reduce, fundamental graph algorithms
- **What is expected:**
  - Present the Map/Reduce paradigm
  - Clearly explain all the the graph algorithms and their implementations using Map/Reduce paradigm
  - Must cover both papers in detail
- Tutor: Sairam Gurajada

Topic 2: Graph Analysis Using Map/Reduce

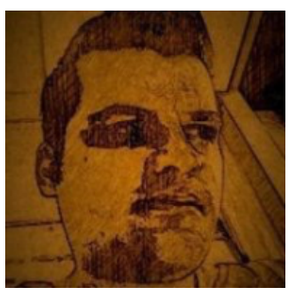
- 15/05/2015
- Kang et al.: **Pegasus: A peta-scale graph mining system implementation and observations,** ICDM 2009
- Kang et al.: **PEGASUS: mining peta-scale graphs**, Knowledge and Information Systems 2011
- **Preferred background**: Databases, Knowledge of Map/Reduce, fundamental graph algorithms, matrix operations
- **What is expected:**
  - Build on and relate to previous topic
  - Focus on second paper (first paper is a subset of the second paper)
  - Clearly explain the matrix multiplication implementation and graph algorithms in Pegasus
  - Discuss evaluations
- Tutor: Sairam Gurajada

Topic 3: Pregel

- 22/05/2015
- Malewicz et al.: **Pregel: a system for large-scale graph processing,** SIGMOD 2010
- Salihoglu and Widom: **Optimizing Graph Algorithms on Pregel-like System**, VLDB 2014
- Additional Reference: McCune et al.: **Thinking Like a Vertex: a Survey of Vertex-Centric Frameworks for Large-Scale Distributed Graph Processing**., ACM Computing Surveys 2015
- **Preferred background**: Databases, Distributed Systems, Message Passing model, Bulk Synchronous Parallel (BSP) model
- **What is expected:**
    - Introduce BSP, contrast it to Map/Reduce model (Refer to Thinking Like a Vertex by McCune et. al. for explanation of BSP and other communication models)
    - Explain Pregel architecture
    - Explain Pregel applications : graph algorithm implementations on Pregel in detail from second paper
    - Discuss evaluations
- Tutor: Sairam Gurajada

Topic 4: GraphLab

- 22/05/2015
- Gonzalez et al.: **PowerGraph: Distributed Graph-Parallel Computation on Natural Graphs**, OSDI 2012, [Paper]
- Low et al.: **Distributed GraphLab: A Framework for Machine Learning and Data Mining in the Cloud**, VLDB 2012, [Paper]
- **Preferred background**: Databases, Distributed Systems, Message Passing model, Map/Reduce, Pregel, fundamental graph algorithms
- **What is expected:**
  - Introduce GraphLab (second paper has better introduction)
  - Compare and contrast Pregel, GraphLab, PowerGraph, Hadoop
  - Explain distributed GraphLab applications (Netflix recommendation challenge, video co-segmentation, Named entity recognition)
  - Discuss evaluations
- Tutor: Stephan Seufert



Topic 5: Graph Partitioning

- 05/06/2015
- Stanton and Kliot: **Streaming Graph Partitioning for Large Distributed Graphs**, KDD 2012, [Paper]
- Tsourakakis et al.: **FENNEL: streaming graph partitioning for massive scale graphs**, WSDM 2014, [Paper]
- **Preferred background**: Databases, Algorithms and Datastructures
- **What is expected:**
    - Introduce the problem of balanced graph partitioning
    - Explain the heuristics from the first paper (including METIS)
    - Introduce formalization of the graph partitioning problem (from FENNEL paper) if time permits
    - Explain the streaming algorithm
    - Discuss evaluations
- Tutor: Mohamed Yahya

Topic 6: Large-Scale Graph Engines

- 12/06/2015
- Kyrola et al.: **GraphChi: Large-Scale Graph Computation on Just a PC**, OSDI 2012
- Shao et al.: **Trinity: A Distributed Graph Engine on a Memory Cloud**., SIGMOD 2013,
- **Preferred background**: Databases, Distributed Systems, Operating Systems
- **What is expected**:
  - Present either GraphChi or Trinity in detail
  - If you want to present Trinity introduce GraphChi briefly
  - Explain the challenges of graph algorithms for evolving graphs and explain how they are handled in GraphChi or Trinity
  - Discuss experiments
- Tutor: Vinay Setty

Topic 7: Comparison of Approaches

- 19/06/2015
- Lu et al.: **Large-Scale Distributed Graph Computing Frameworks**: An Experimental Evaluation, VLDB 2014
- McCune et al.: **Thinking Like a Vertex: a Survey of Vertex-Centric Frameworks for Large-Scale Distributed Graph Processing**., ACM Computing Surveys 2015
- **Preferred background**: Databases, Distributed Systems, Operating Systems
- **What is expected**:
  - Summarize all the presented approaches Pregel, GraphLab, GraphChi
  - Compare and contrast these approaches
  - Discuss experiments in detail
  - Use second paper (survey) mostly as additional reference to get more information
- Tutor: Vinay Setty

Topic 8: RDF Graph Processing

- 26/06/2015
- Neumann et al.: **RDF-3X: a RISC-style engine for RDF**, VLDB 2008,
- Huang et al.: **Scalable SPARQL Querying of Large RDF Graphs**, VLDB 2011
- **Preferred background**: Databases
- **What is expected**:
  - Introduce RDF and SPARQL briefly
  - Introduce RDF-3X discuss limitations
  - Explain the parallel SPARQL engine from second paper
  - Discuss experiments
- Tutor: Mohamed Yahya

Topic 9: Graph Streams

- 03/07/2015
- Aggarwal et al.: **On dense pattern mining in graph streams**., VLDB 2010
- Chen et al.: **Continuous Subgraph Pattern Search over Certain and Uncertain Graph Streams**, TKDE 2010
- **Preferred background**: Databases, Algorithms and Data structures
- **What is expected**:
  - Introduce graph stream processing
  - Explain (sub)graph isomorphism problem, approximate graph search and dense patterns
  - Briefly explain the approaches from both papers
  - Discuss evaluation
- Tutor: Vinay Setty

Topic 10: Graph Algorithms: Dense Subgraphs and Graph Sketches

- 10/07/2015
- Angel et al.: **Dense subgraph maintenance under streaming edge weight updates for real-time story identification**, VLDB 2014, [Paper]
- Ahn et al.: **Graph sketches: sparsification, spanners, and subgraphs**, PODS 2012, [Paper]
- **Preferred background**: Databases, Algorithms and Data structures
- **What is expected**:
  - Pick one of the papers and present in detail
  - Someone comfortable with presenting theoretical analysis is preferred
- Tutor: Mohamed Yahya 

Presentation

- 45 minutes talk in English
- Around 30 minutes of Q&A
- Talk to your tutor **at least 2 weeks** before your talk
    - If you get the first topic you have to start now! (it is an easier topic)
- **Prepare your own slides** (keynote, power point or latex code must be sent to your tutor)
- You must send your slides to and discuss them with your tutor by the **Monday before your talk (by 16:00)** at the latest, otherwise your talk will be canceled (this is a hard deadline)

Guidelines for Presenting

- It is important to **clearly introduce** the problem and the idea presented in the papers
- In the papers look for:
    - **Contributions** of the paper
    - Improvements to the **state-of-the-art**
    - Main **results**
    - **Conclusions** and **future work**
- Discuss the **insights** that are provided in the papers
- Identify **strengths** and **weaknesses**, question the **assumptions**, criticize the bad **decisions** in the papers
- Refer to https://web.stanford.edu/~jacksonm/present.pdf

Report

- Up to **8 pages** using the template in the course web page
- Maximum of **4 weeks** after your talk
- Contents of the report:
  - include the **basic idea** presented in the papers
  - **summarize** the papers
  - include the points raised in the seminar by the **opponents** (collaborate with the opponents if necessary)
  - include the **important results** and **conclusions**
  - **bonus points** if you include the content from the papers outside of your assigned papers (such as any follow up works, new results etc)

Opponents

- Two **opponents** per talk
- One of the opponents **introduces** the speaker
- Other opponent **moderates** the Q&A session
- Both opponents must **read the papers thoroughly**
  - prepare **questions**
  - **challenge** the ideas and results in the papers (if there are any weaknesses)

Participation

- **Participation** in all the talks is **mandatory** (not just your own)
  - If you are **sick** please let us know in advance
    - If you miss more than two seminars you need to provide a note from the doctor
- **Active participation** is required (we will monitor on who is active and who is not)

What counts for grade?

- Your presentation 50%
- Report 25%
- Your performance as an opponent 15%
- Active participation in the seminars 10%

Registration

- Send an email to vsetty@mpi-inf.mpg.de
    - Your name, student number and semester
    - Your background: mention the relevant lectures you have taken, any seminars you have taken
    - Three topics in the order of the their preference and a brief explanation to show if you have any relevant background for that topic
- Maximum 10 participants
- You will know the topic assignment by Monday, 27th April (may be earlier)