neoGERM: Mining Graph Evolution Rules in Neo4j

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Nataliia Pobiedina
Electronic Commerce Group,
Vienna University of Technology

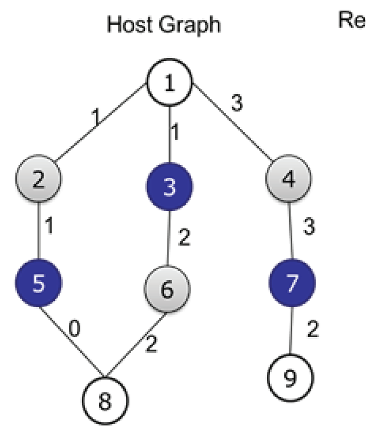
electronic commerce group
Institute of Software Technology and Interactive Systems

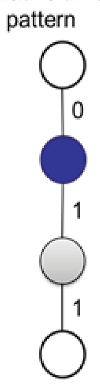
Problem Statement

- Given graph $G=(V,E,L)$:
  - V is the set of nodes
  - E is the set of edges
  - L is the set of labels over nodes and edges
- Time evolving graph G
  - t assigns timestamps to edges
  - λ assigns labels to nodes
- The task is to find **frequent based evolution rules** in the graph G.

Following GERM (Graph Evolution Rule Miner) from
** B. Bringmann et al.,*
Learning and Predicting the Evolution of Social Networks.
IEEE Intelligent Systems.
July/August 2010, Vol. 25, 4, pp. 26-34.

Definitions





A	B	C
8	1	9
5	3	7
2	6	4
1	8	1

$\Delta \in \mathbb{R}$ *and* $\varphi: V_p \to V_g$ *such that* (1) $\forall v \in V_p \Rightarrow \lambda_p(v) = \lambda_g(\varphi(v))$
and (2) $\forall (u,v) \in E_p \Rightarrow (\varphi(u),\varphi(v)) \in E_g$
and (3) $\forall (u,v) \in E_p \Rightarrow t(\varphi(u),\varphi(v)) = t(u,v) + \Delta$

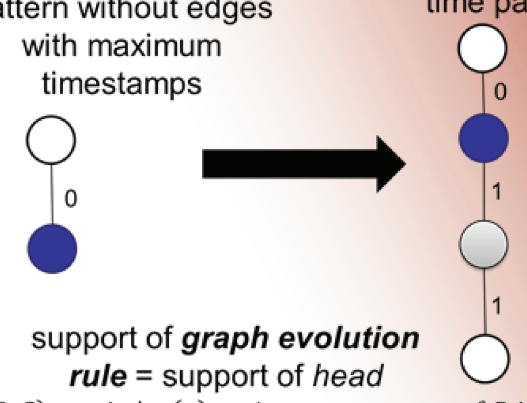
- Colors of the nodes = labels

- **Frequent based pattern** is a pattern with support bigger than certain threshold
- **Support** is a non-negative anti-monotonic function of P

Graph evolution rule
body → head

body = relative time pattern without edges with maximum timestamps

head = relative time pattern



support of ***graph evolution rule*** = support of *head*

- **Minimum image based support**

$$\sigma(P,G) = \min_{v \in V_P} |\varphi_i(v): \varphi_i \text{ is an occurrence of } P \text{ in } G|$$

Motivation

With the increasing popularity of Web large network data can be retrieved to study the evolution of social networks and information flow at microscopic level. The evolution of such network data can be represented in one single structure (*time evolving graph*) since the data in these networks does not disappear with time but rather grows.

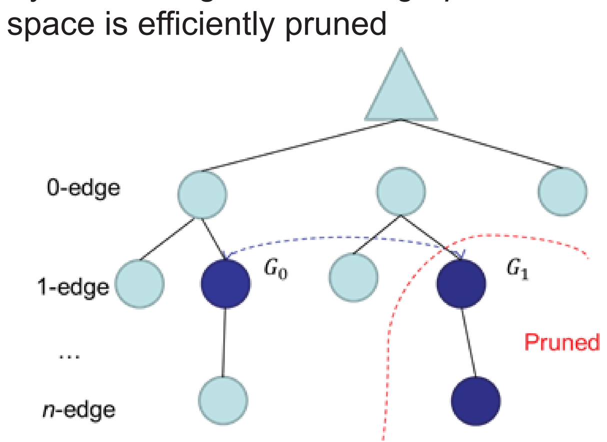
Subgraph Isomorphism

Support calculation requires solving subgraph isomorphism problem.

Current implementation of GERM uses Ullmann's algorithm (Ullmann, 1976).

gSpan

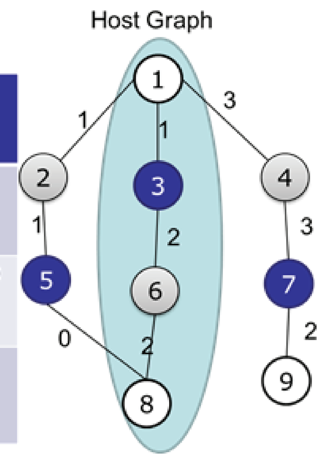
- gSpan is developed for *transactional setting* (Yan and Han, 2002)
- By constructing *DFS Code Tree* patterns are taken from the original graph and not generated randomly
- By introducing *DFS Lexicographic Order* search space is efficiently pruned



DFS Code

- **DFS Code** $\alpha = (a_0, a_1, \ldots, a_m)$, where $a_k = (i_k, j_k, l_{i_k}, l_{(i_k,j_k)}, l_{j_k})$
- **DFS Lexicographic Order** is a lexicographic combination of DFS order over edges and full order over node labels

Edge no.	α	β	γ
0	(0,1,W,1,P) *Edge (1,3)*	(0,1,G,2,W) *Edge (6,8)*	(0,1,P,2,G) *Edge (3,6)*
1	(1,2,P,2,G) *Edge (3,6)*	(1,2,G,2,P) *Edge (6,3)*	(1,2,G,2,W) *Edge (6,8)*
2	(2,3,G,2,W) *Edge (6,8)*	(2,3,P,1,W) *Edge (3,1)*	(2,3,P,1,W) *Edge (3,1)*



- DFS Code encodes a pattern, but not uniquely

GERM vs neoGERM

- GERM adapts gSpan to one single graph setting
- GERM works only for undirected graphs
- GERM requires data to be stored as document
- GERM allows only integer numbers as labels over nodes

- neoGERM requires data to be stored in Neo4j and incorporates in-built methods from Neo4j
- neoGERM works for directed graphs
- neoGERM works with node labels as set of integers and redefines pattern for partial matching
- neoGERM further improves support calculation

Citation Network

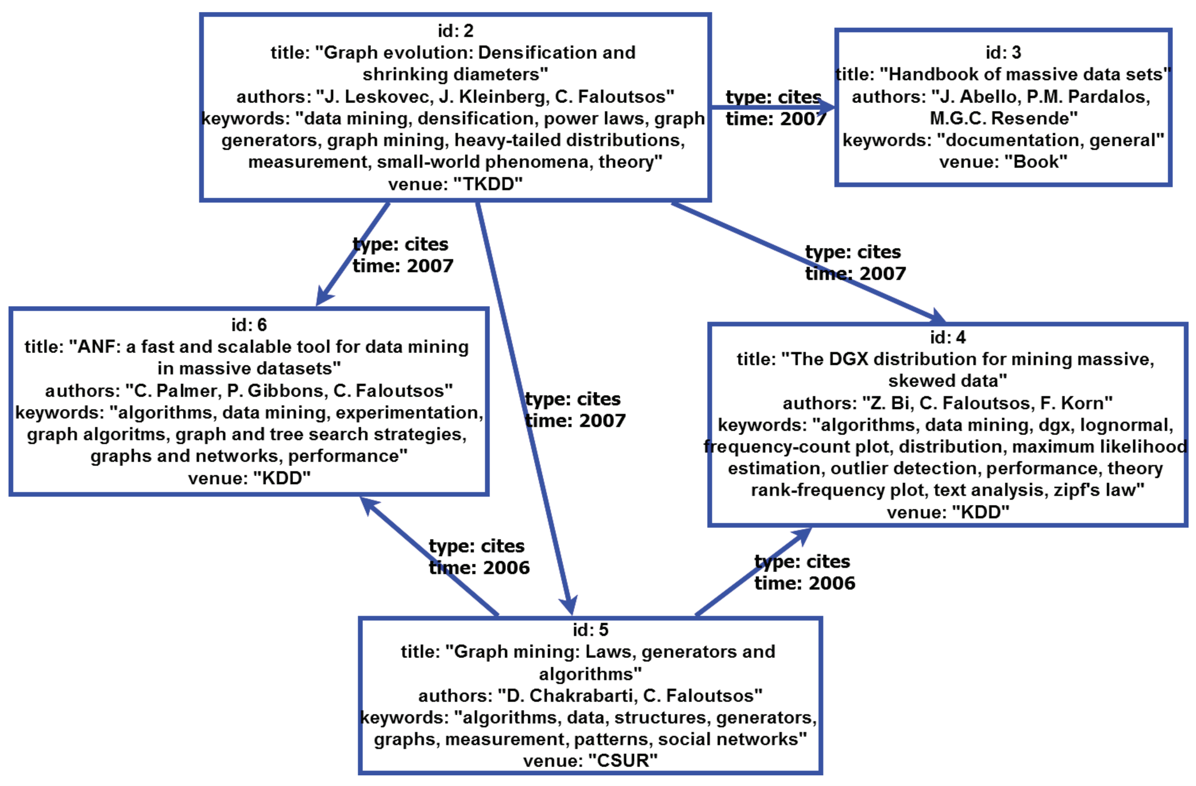
- Citation network has papers as nodes and citations as edges. If a paper *i* cites paper *j*, there is a directed edge from node representing paper *i* to node representing paper *j*.

Two major datasets are considered:

- SWIR dataset which contains papers from major conferences in Semantic Web and Information Retrieval and was crawled from CiteSeerX and DBLP (Belak, Karnstedt and Hayes, 2011)
  - Nodes: 10971
  - Edges: 25481
  - Years: 2001 - 2007
  - Node labels: keywords extracted from full text
    venues
    authors
- ArXiv HEP-TH (high energy physics theory) citation graph (KDD Cup 2003)
  - Nodes: 10971
  - Edges: 25481
  - Years: 2001 - 2007
  - Node labels: keywords extracted from abstract
    venues
    authors

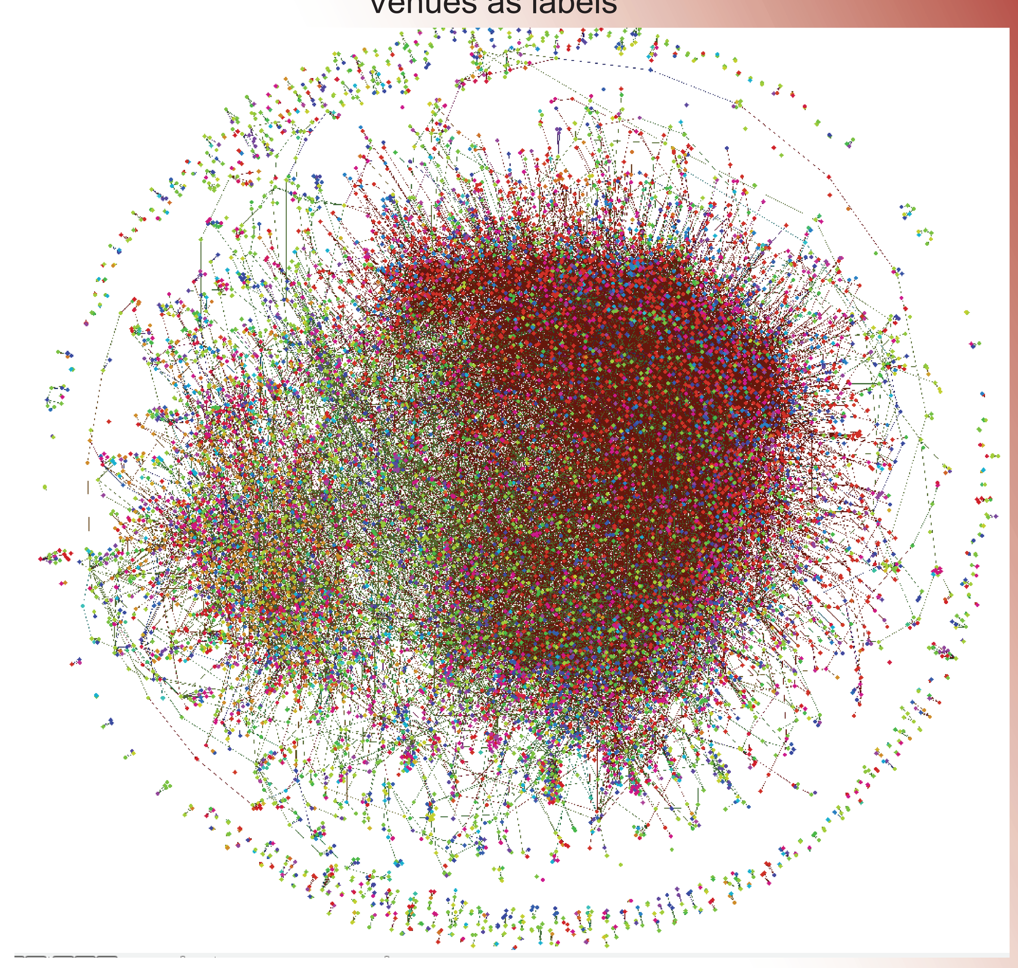
Neo4j Snapshot

- Timestamp on an edge represent the year at which citation takes place, i.e. edge label corresponds to the year of the latest paper.
- Different networks are created from the same dataset:
  - venues as node labels;
  - authors as node labels;
  - keywords as node labels.



SWIR dataset

Venues as labels



Neo4j

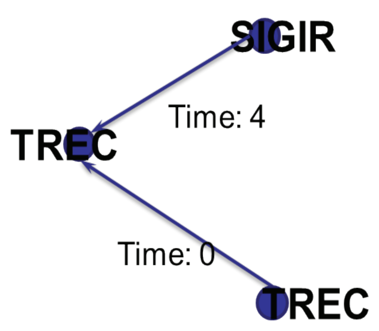
Neo4j is a **NoSQL graph database** (www.neo4j.org).
It has a traversal framework for high speed traversals on the nodes.
Gephi has a plug-in to visualise databases from Neo4j.

Neo4j is an open source project in a GPLv3 Community edition with Advanced and Enterprise editions available under both AGPLv3 and commercial licenses.

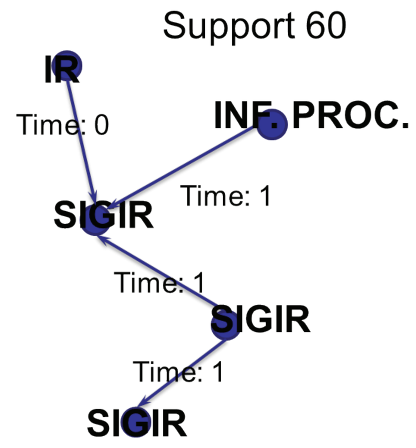
Initial Results

Venues as labels

Support 50



Support 60



Related Work

- Global evolution of networks:
  (Leskovec, Kleinberg, Faloutsos, 2005)
  (Backstrom, Huttenlocher, Kleinberg, Lan, 2006)
- Microscopic evolution:
  (Lescovec, Backstrom, Kumar, Tomkins, 2008)
- Link prediction algorithms like Adamic-Adar/Liben-Nowell and Kleinberg are capable to predict links between existing nodes.

Future Work

- Experiment on different datasets with different settings.
- Further improve support calculation.
- Develop link prediction based upon discovered graph evolution rules.
- Investigate neoGERM scalability and performance on larger datasets.

This work is supported by Vienna PhD School of Informatics.

pobiedina@ec.tuwien.ac.at