Data Mining – Graph Mining

Andreas Karwath   Jörg Wicker

Johannes Gutenberg University Mainz

March 21, 2017

Outline

Acknowledgments

- Slides partially from
  - Stefan Kramer
  - Cecilia Mascolo (Univ. Cambridge)

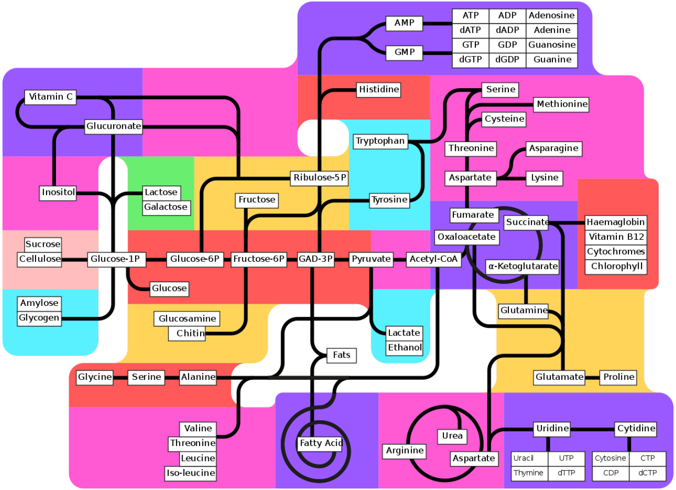
Section 1

Graph Mining

Examples - Social Network

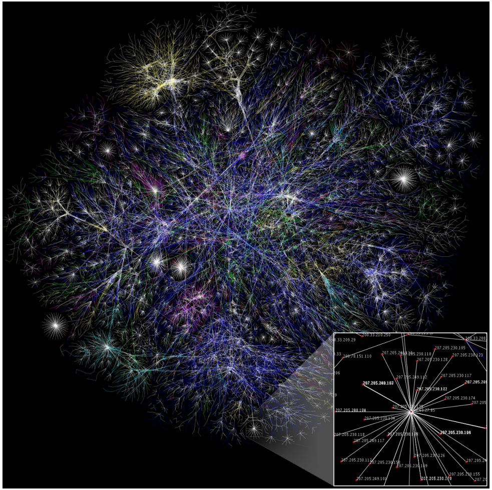
wikimedia.org | Social_Network_Analysis_Visualization.png

Examples - Biological Pathways



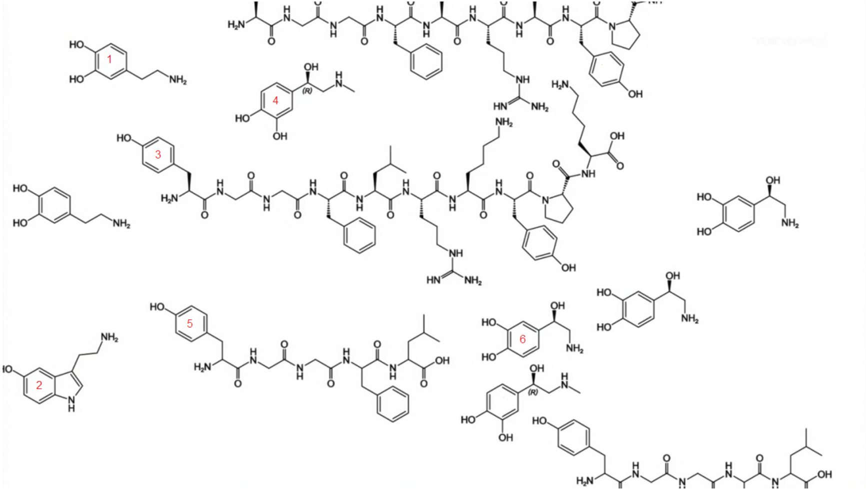
wikimedia.org | Metabolism_pathways_(partly_labeled).svg

Examples - Internet Network



wikimedia.org | Internet_map_1024.jpg

Examples - Chemical Compounds



i.stack.imgur.com | 1jjlI.png

What is Graph Mining?

- Graph: A collection of vertices (nodes) and edges.
  - Vertices can be labelled or unlabelled
  - Edges can be labelled or unlabelled
  - Edges can be directed or undirected
  - Vertices and edges can contain attributes
- Typical settings
  - Finding similarities between (small) graphs in a collection of graphs. Similarities are commonly expressed as so-called sub-graphs. However, other approaches exist.
  - Finding intersting sub-structures (sub-graphs) in one large network or detect missing edges (links) or vertices (nodes).
  - Comparison of a small number of networks.

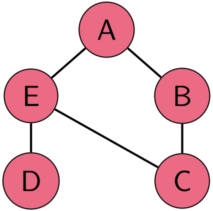
Graph Definitions

- A **graph** G is a tuple (V, E) of a set of vertices V and edges E. An edge in E connects two vertices in V.
- A **neighbour set** $N(v)$ is the set of vertices adjacent to v:

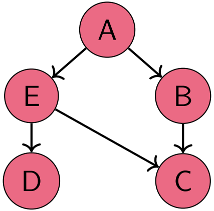
$$N(v) = \{u \in V | u \neq v, (v, u) \in E\}$$

- The **degree** of a node is the number of neighbours of a node.

Directed and Undirected Graphs



Undirected graph



Directed graph

Examples of ...

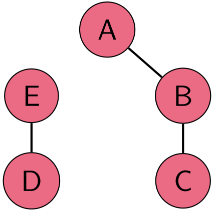
... undirected Graphs: Facebook, Co-presence, WhatsApp. *etc.*
... directed: Twitter, Email, Phone Calls, *etc.*

Paths and Cycles

- A **path** is a sequence of nodes in which each pair of consecutive nodes is connected by an edge.
  - If a graph is directed the edge needs to be in the right direction.
  - e.g. A-E-D is a path in both previous graphs
- A **cycle** is a path where the start node is also the end node.
  - e.g. E-A-B-C is a cycle in the undirected graph

Connectivity

- A graph is **connected** if there is a path between each pair of nodes.
- Example of a **disconnected** graph:



Components

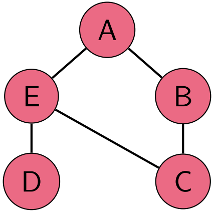
- A **connected component** of a graph is the subset of nodes for which each of them has a path to all others (and the subset is not part of a larger subset with this property).
  - Connected components: A-B-C and E-D in the example before

- A **giant component** is a connected component containing a significant fraction of nodes in the network.
  - Real networks often have one unique giant component.

Path Length/Distance

- The **distance** (d) between two nodes in a graph is the length of the shortest path linking the two graphs.

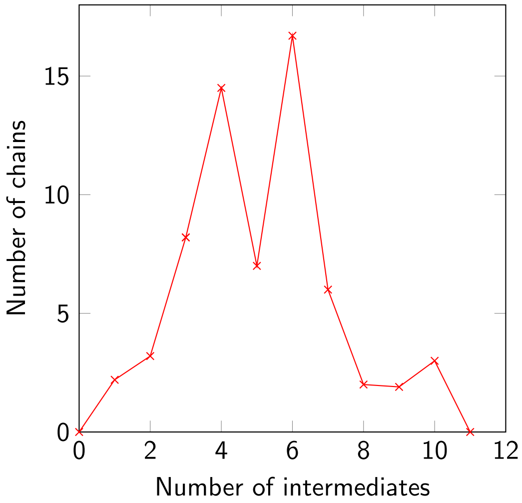
- The **diameter** of the graph is the maximum distance between any pair of its nodes.



What is the diameter here?

Small-world Phenomenon Milgrams Experiment

- Two random people are connected through only a few (6) intermediate acquaintances.
- Milgrams experiment (1967) shows the known "six degrees of separation":
  - Choose 300 people at random Ask them to send a letter through friends to a stockbroker near Boston.
  - 64 successful chains.



Milgram's Findings

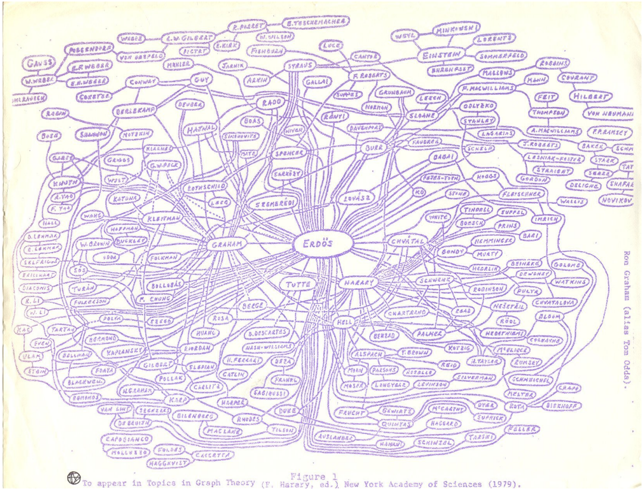
- Use of "weak ties" and professional relationships
- Median of 5-7 steps
- "Network structure alone is not everything"

- Some different incentives had a high impact on completion rate of chains
  - If the target was in a prominent place (e.g. a professor)

- Has been repeated using Facebook
  - Backstrom *et al.* (2012):"Four Degrees of Separation", WebSci 2012, pp. 33-42, ACM

Erdős number

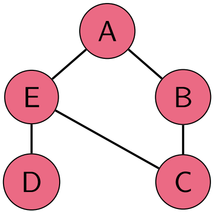
- Erdős Number: distance from the mathematician (most people are 4-5 hops away) based on collaboration.



Adjacency Lists

- Each vertex has an associated list of its adjacent vertices

- Example:

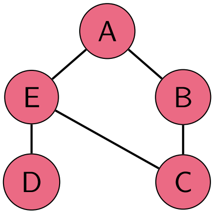


A: B, E
B: A,C
C: B, E
D: E
E: A,C, D

Adjacency Matrices

- The adjacency matrix of a graph $G = (V, E)$ is an $n \times n$ matrix A, such that:
  $a_{ij} = \begin{cases} 1 & if (i, j) \in E \\ 0 & otherwise \end{cases}$

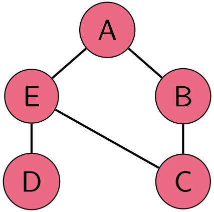
- Example:



	A	B	C	D	E
A	0	1	0	0	1
B	1	0	1	0	0
C	0	1	0	0	1
D	0	0	0	0	1
E	1	0	1	1	0

Incidence Matrices

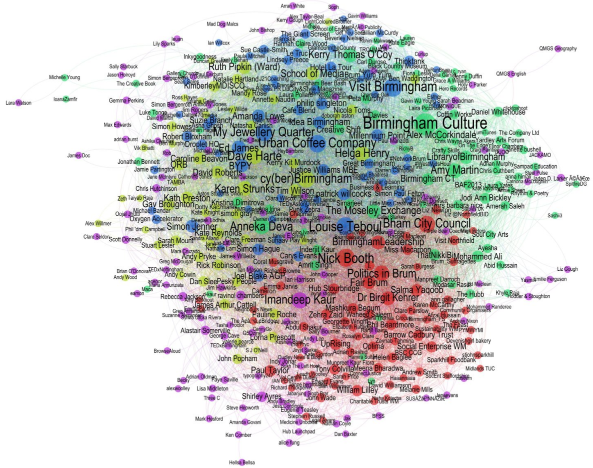
- Similar to adjacency matrices, but the incidence matrix (symbolized here using C) represents vertices and edges.
- Example:



	A-B	A-E	B-C	E-C	E-D
A	1	1	0	0	0
B	1	0	1	0	0
C	0	0	1	1	0
D	0	0	0	0	1
E	0	1	0	1	1

- Relationship between an adjacency A and incidence matrix C:
  $A = C^T C - 2I$

Graph Clustering



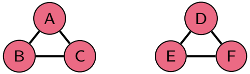
thedatamine.files.wordpress.com | tweeters-till-sat-am-graph-org2-1680x1187.jpg

Graph Clustering - Spectral Partitioning

- Spectral partitioning (roughly):
  - Given an adjacency matrix A of a graph and the degree matrix D (a diagonal matrix containing at d_{ii} the degree of node i)
  - Calculate the Laplacien matrix $L = D - A$
  - Calculate the Eigenvalues and Eigenvectors of L
  - The eigenvalue λ_0 tells one whether the graph is connected or not.
  - The eigenvector v_1 of the second lowest Eigenvalue λ_1 (a.k.a the Fiedler vector) provides an assignment to each vertex in the graph. This assignment can be used to partition (cluster) the graph.
  - If a graph has k connected components, then eigenvalue (λ_0) has multiplicity k (i.e. k distinct non-trivial eigenvectors).

Graph Clustering - Spectral Partitioning - Example

- Toy example:

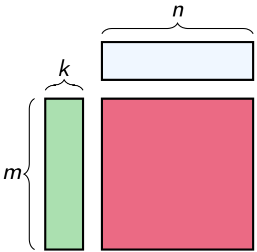


- Assume λ_0 has multiplicity 2 and the 2 distinct eigenvectors look like:
  - $\langle 1, 1, 1, 0, 0, 0 \rangle$ and $\langle 0, 0, 0, 1, 1, 1 \rangle$
- Consider a matrix with these eigenvectors as its columns: $\begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$
- Each column corresponds to a cluster of vertices.

Graph Clustering - Non-negative Matrix Factorization (NMF)

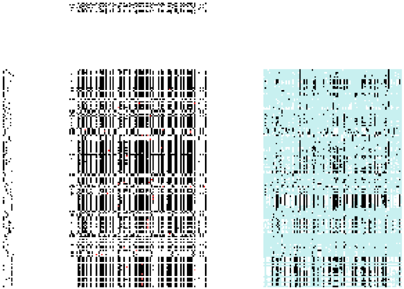
- Given a matrix $A = [a_1, a_2, \dots, a_n] \in \mathbb{R}^{m \times n}$
- NMF aims to find two non-negative matrices U and V with:
  - $U = [u_{il}] \in \mathbb{R}^{m \times k}$ and $V = [v_{lj}] \in \mathbb{R}^{k \times n}$

  whose product can well approximate the original matrix A:
  - $A \approx UV$
- Graphically:



BMaD – A Boolean Matrix Decomposition Framework

- Java library for Boolean matrix decomposition
- Modular steps can be freely combined into a decomposition algorithm
  - Candidate generation
  - Basis selection
  - Boolean combination
- Capable of handling missing values
- Can read and write WEKA instances
- Freely available, licensed under GPLv3



- Applications in (multi-label) classification, clustering, pattern mining, . . .

Assignment : Clustering using NMF

- Given a datasets with different networks:
  - Construct adjacency matrix A
  - Use a NMF approach (BMAD) to find k clusters (U)
  - Evaluate different ks with regards to the reconstruction error for the *best* clustering.
  - Visualize the results