Infinite mixtures for multi-relational categorical data

Janne Sinkkonen[1,2], Janne Aukia[2], Samuel Kaski[1]

[1]Helsinki University of Technology

[2]Xtract Ltd

A simple generative model for enriched graph data

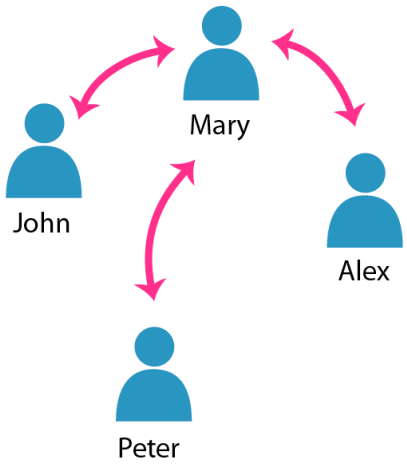
Janne Sinkkonen[1,2], Janne Aukia[2], Samuel Kaski[1]

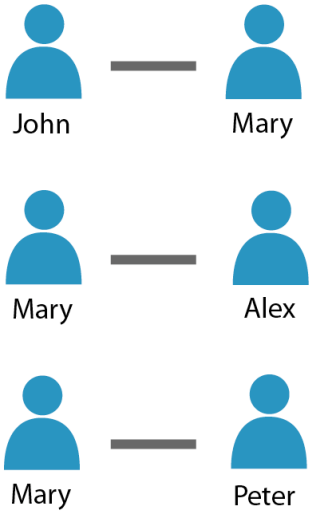
[1]Helsinki University of Technology

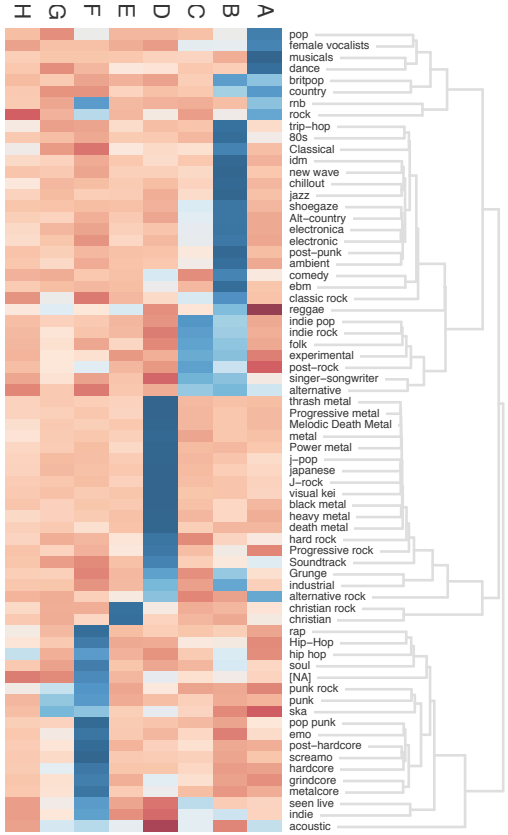
[2]Xtract Ltd

Background

- MLG'07: Probabilistic community finding
  - Community: tightly connected subgraph
  - Applied to a large social network (10^6 nodes) of Last.fm
  - For nondirected graphs

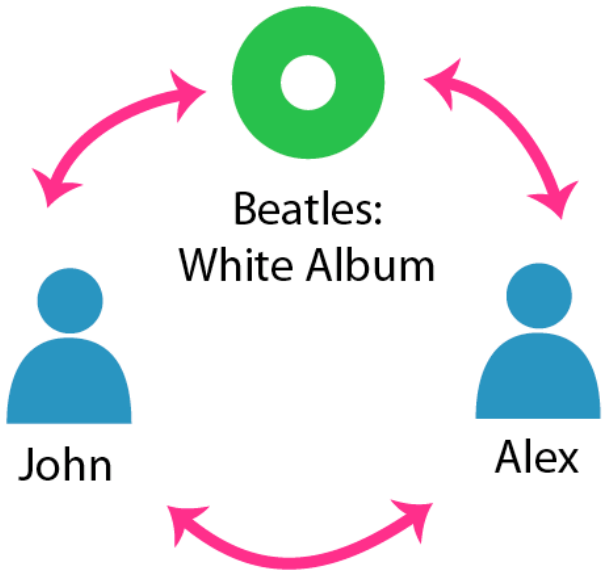






The graph model is restrictive

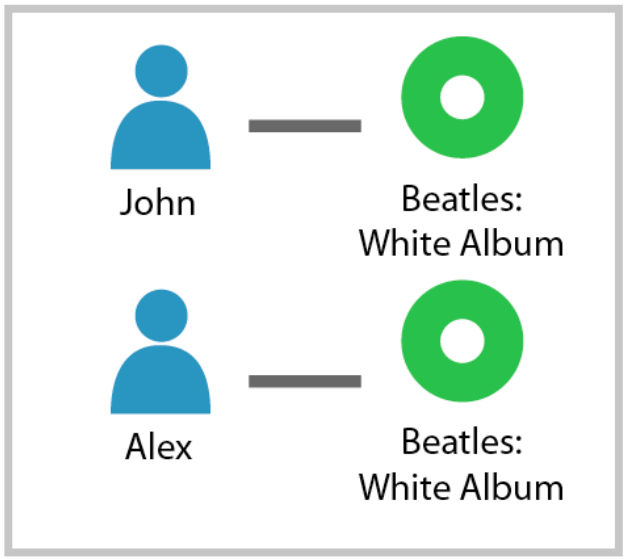
In real world pure graphs are often a poor data model



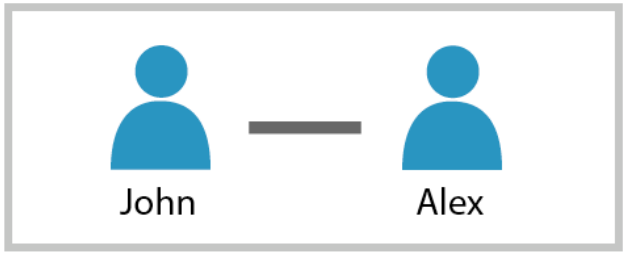
Motivation:

1. Include richer data
2. Still have global components

Type 1 relation

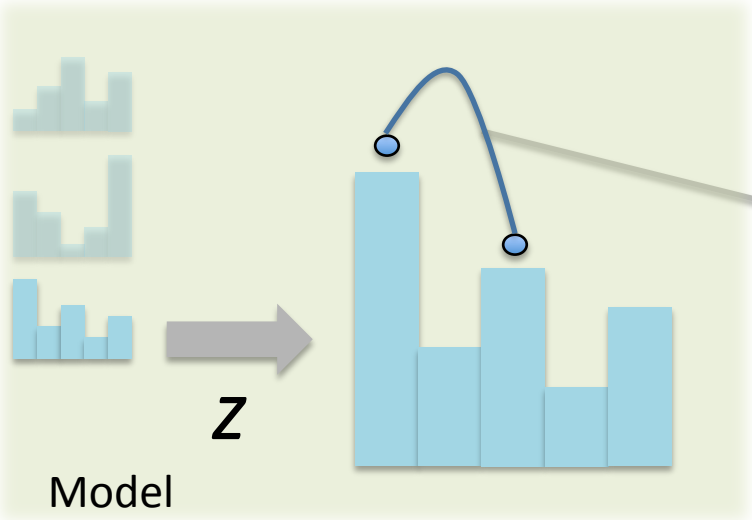


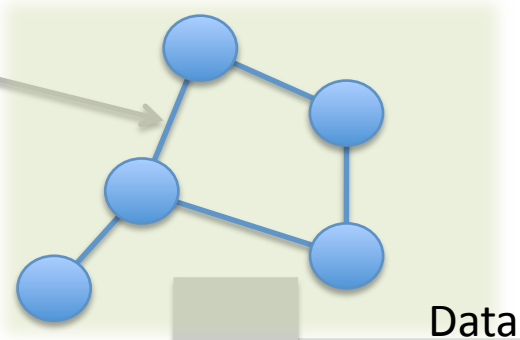
Type 2 relation



Probabilistic framework

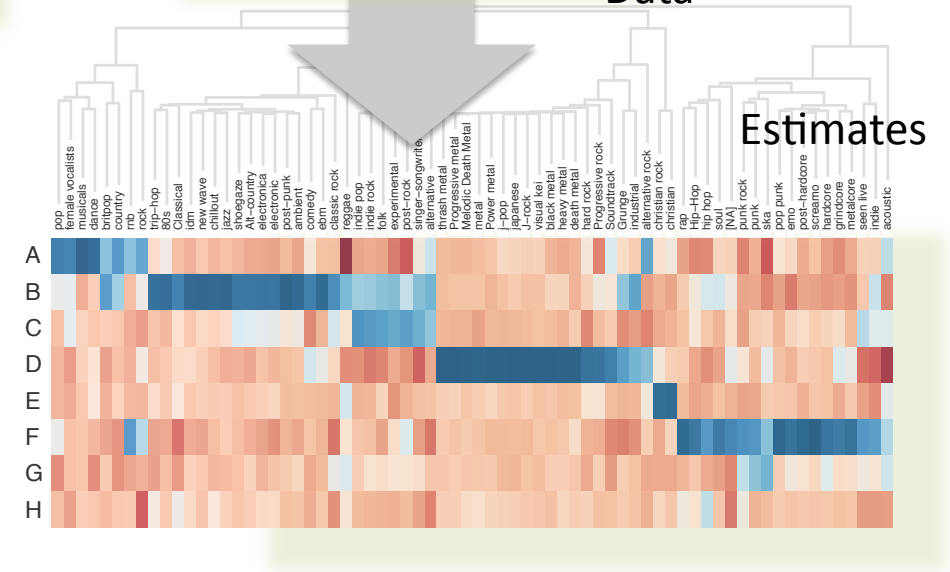
Components generate edges, according to some known process (model) and unknown or latent parameters.





Data

Estimates

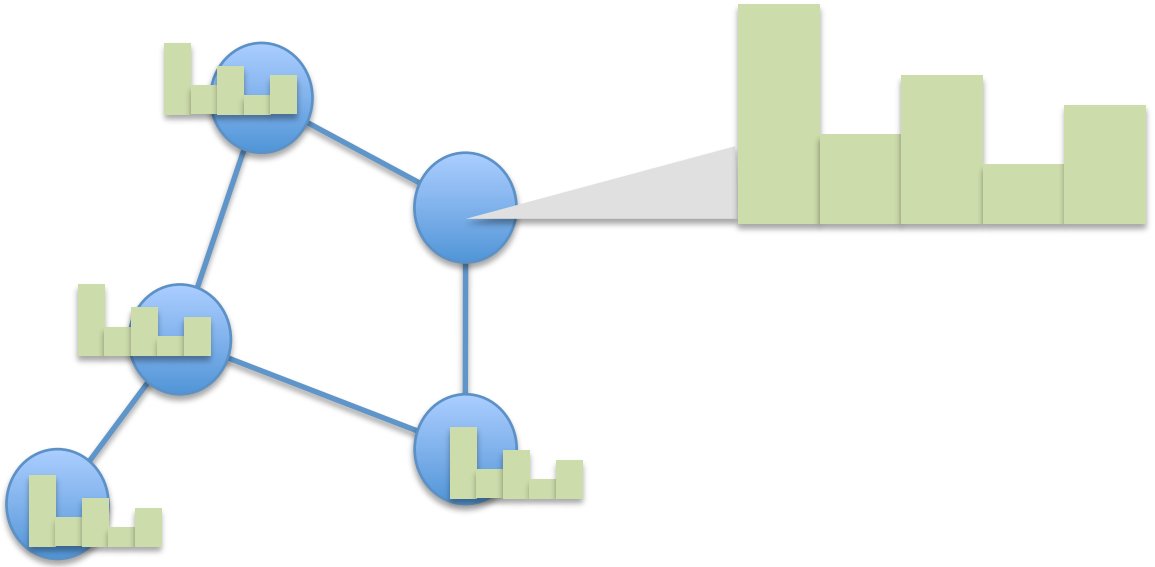


Unknowns are inferred from the observed graph by Bayesian techniques (Gibbs sampling).

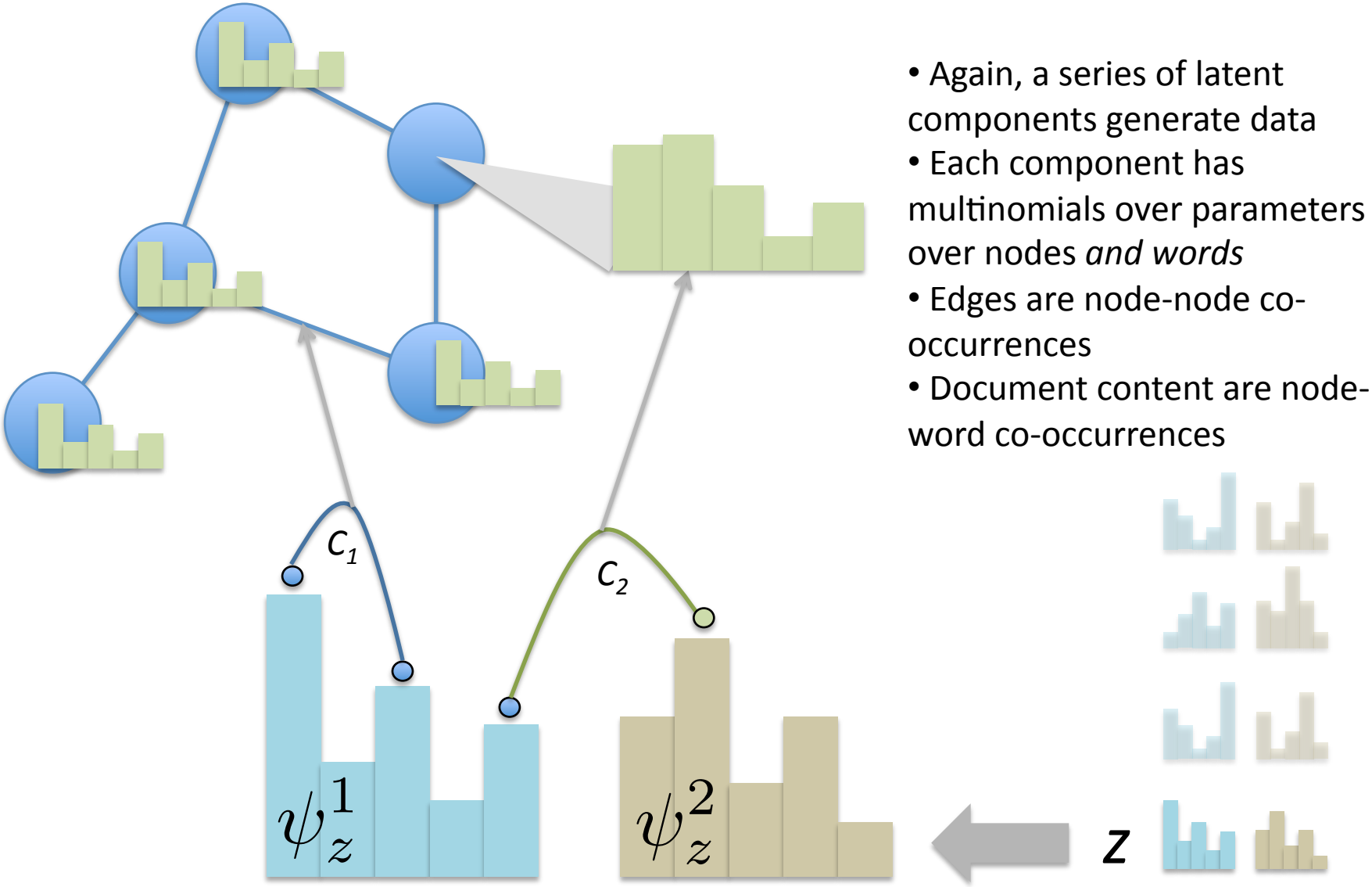
Parameters tell community memberships.

A richer graph

- Scientific papers
  - Citations modeled as a graph
  - Content modeled as a word histogram

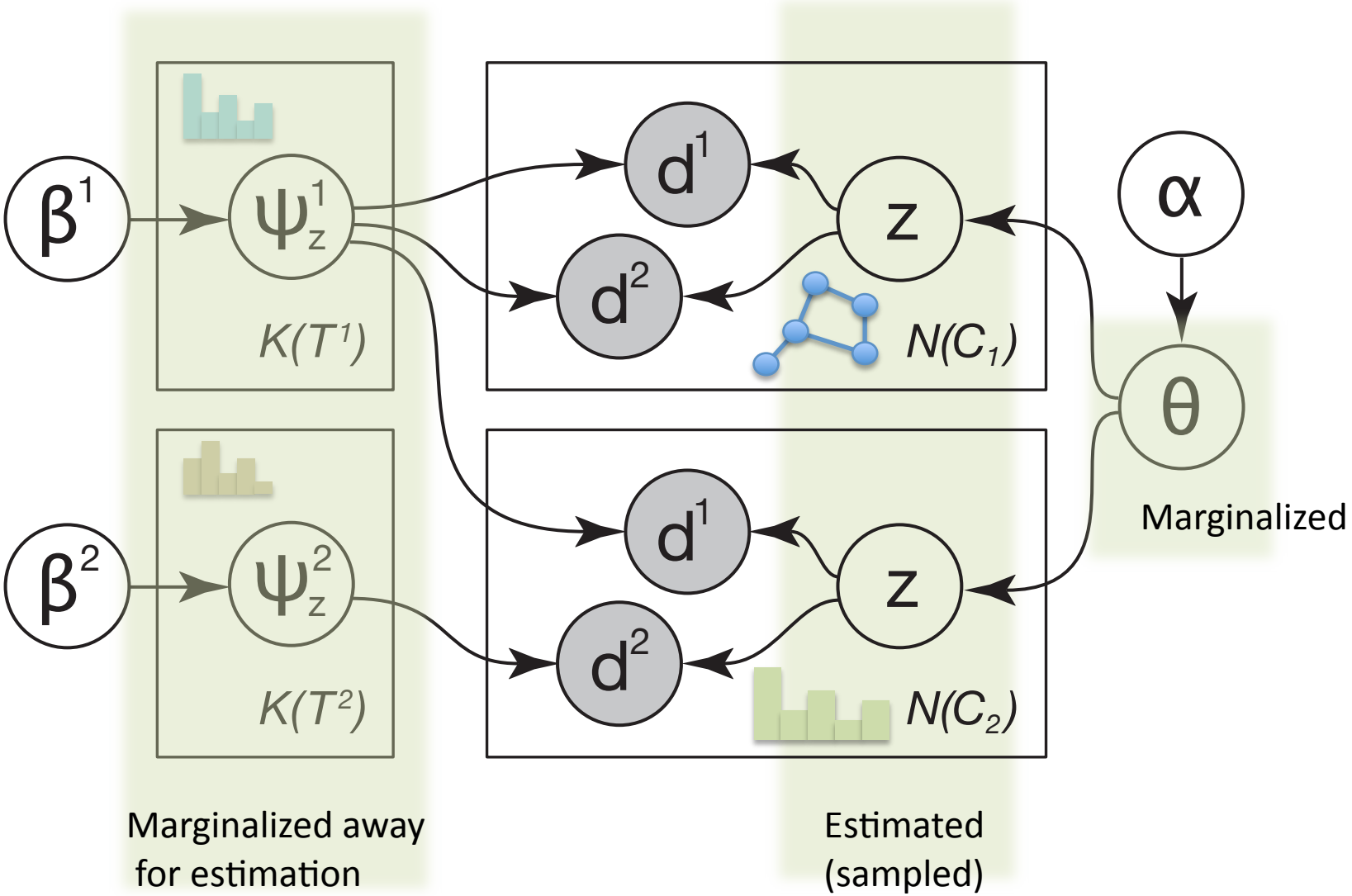


Generative process



- Again, a series of latent components generate data
- Each component has multinomials over parameters over nodes *and words*
- Edges are node-node co-occurrences
- Document content are node-word co-occurrences

Generative process II



Estimation

- Estimation by collapsed Gibbs sampling
- Iterate over objects (citations, words)
  - Draw latent component for the object, given data and components of all other objects
  - Component assignments eventually converge to the true posterior distribution

$$p(z, D_i|\dots) \propto \frac{\{n_z, \alpha\}}{N+\alpha} \times \begin{cases} g^{(1)}_{z,l_1} g^{(1)}_{z,l_2} / (g^{(1)}_{z,\cdot}(g^{(1)}_{z,\cdot}+1)) & \text{for } D_i \in C_1 \\ g^{(1)}_{z,l_1} g^{(2)}_{z,l_2} / (g^{(1)}_{z,\cdot}(g^{(2)}_{z,\cdot})) & \text{for } D_i \in C_2 \end{cases}$$

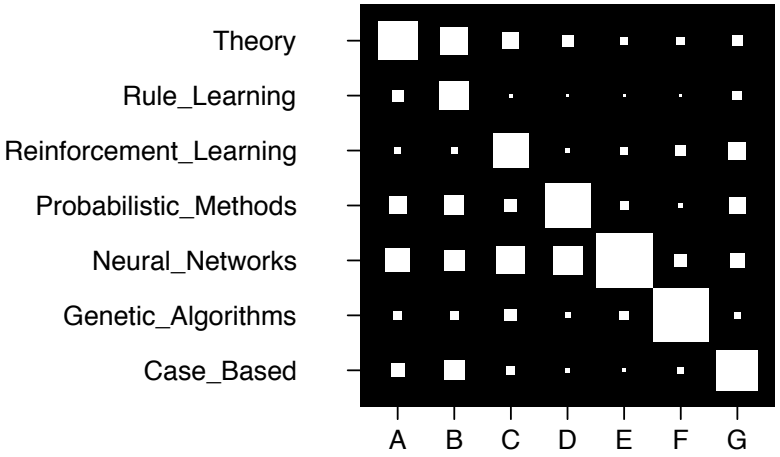
- Collect samples of *z*, summarize somehow.

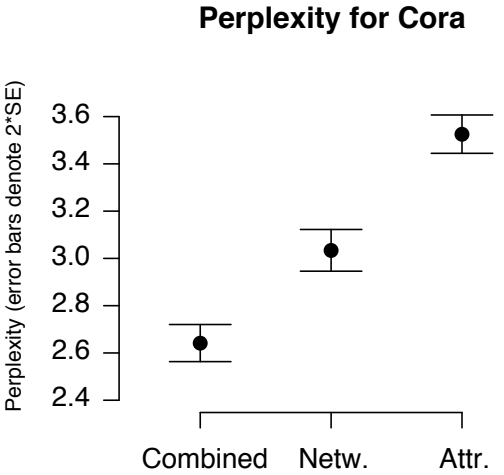
Component priors

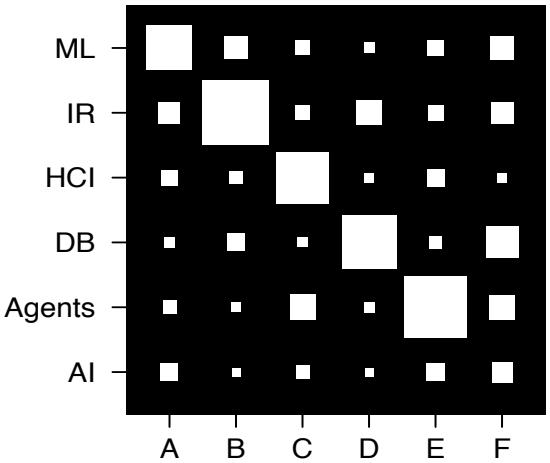
- Dirichlet prior for a fixed number of components.
  - Requires component number and a hyperparameter.
- Dirichlet process prior.
  - Only a hyperparameter, no fixed number of components.
  - "Infinite mixture"
    - But only part of the components generate any observations.
- Both are equally easy to sample.

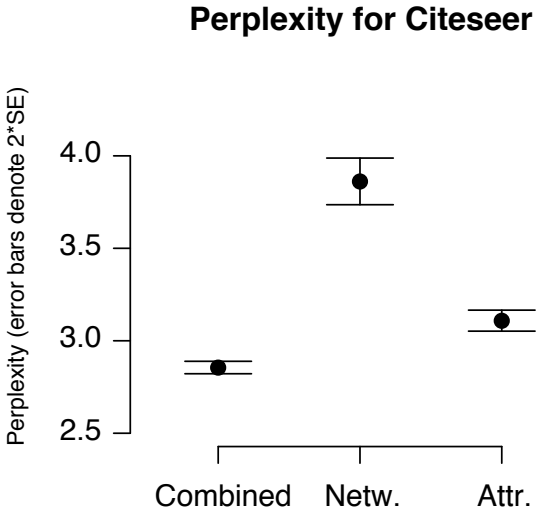
$$p(z|\alpha, N) = \frac{n_z + \alpha}{N + K\alpha} \qquad p(z|\alpha) = \frac{\{n_z, \alpha\}}{N + \alpha}$$

Demo on Cora and Citeseer

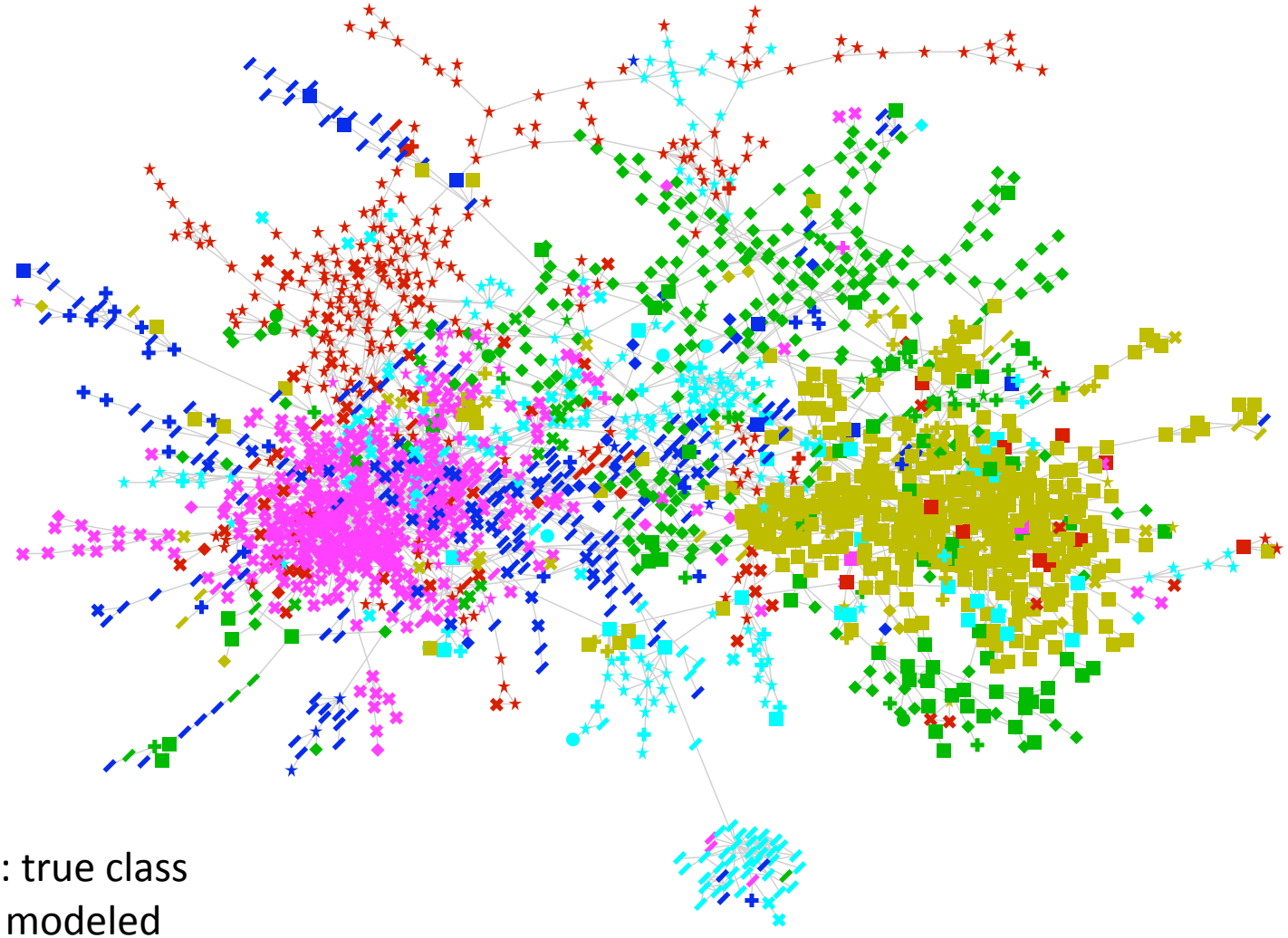






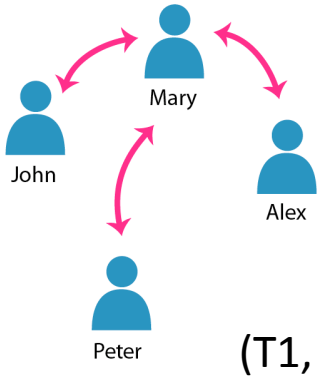


Citeseer graph

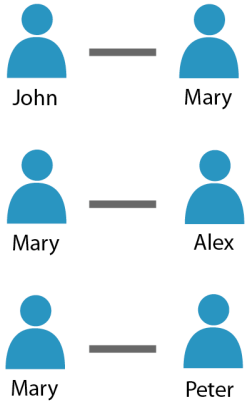


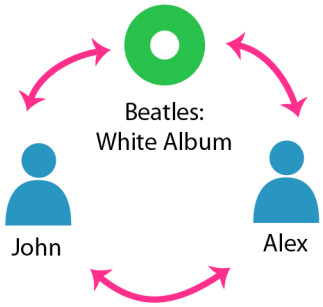
shape: true class
color: modeled

Formalization as relations

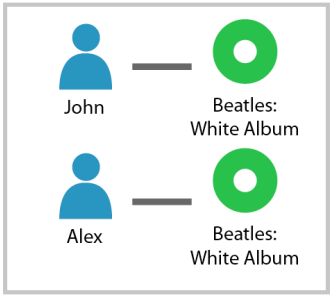


(T1, T1)

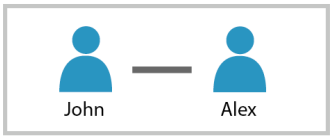


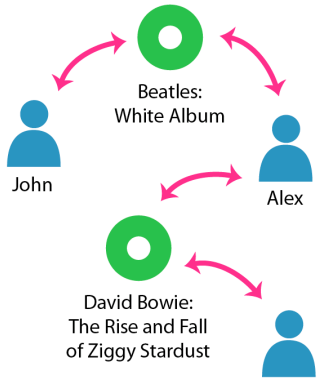


Type 1 relation

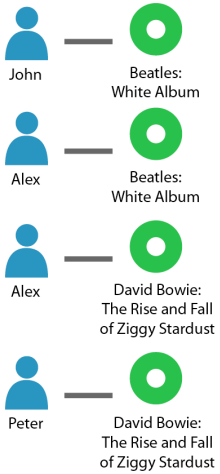


Type 2 relation

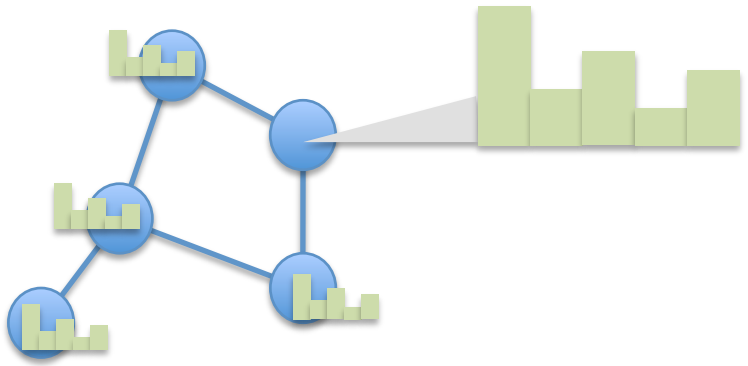




(T1, T2)

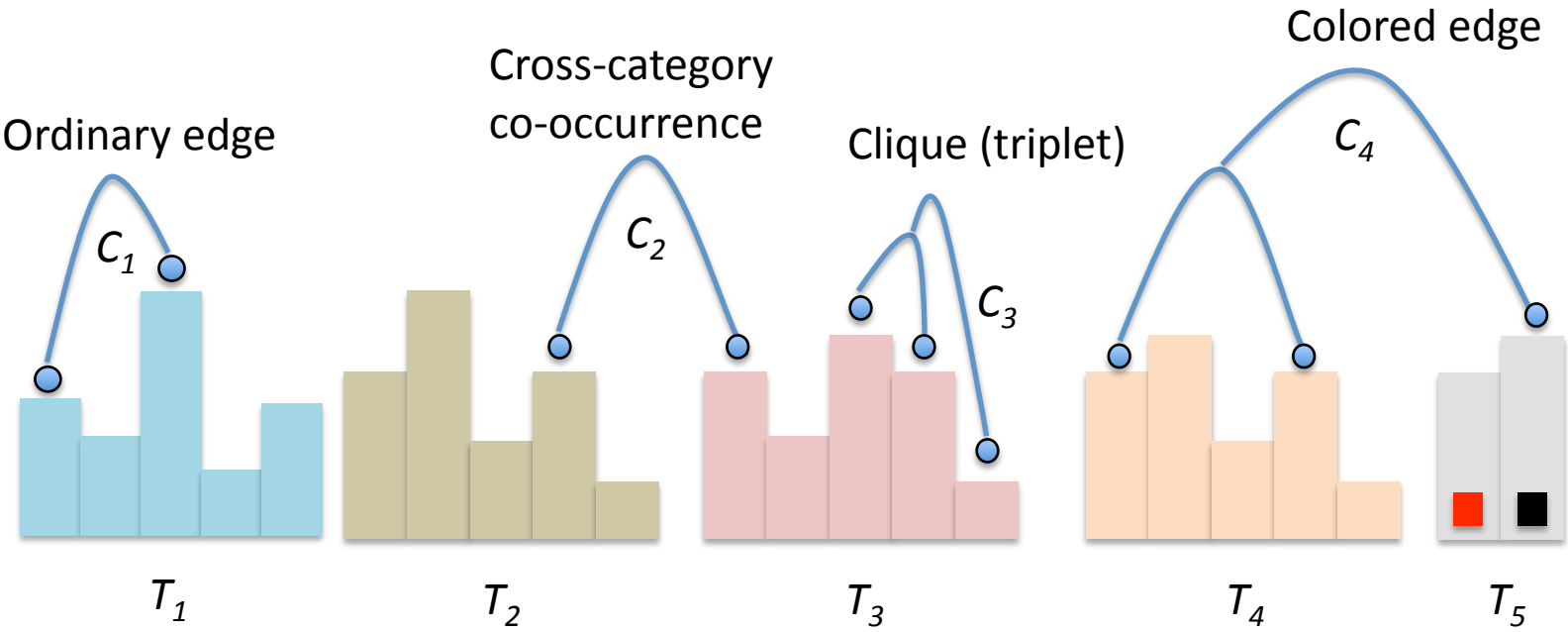


(T1, T1)
(T1, T2)



General case: examples

Objects: co-occurrences of atomic *events*



Parameters of atomic event types (multinomial), given a component z

(Other components have the same parameters)

General case

- Data: heterogeneous objects
  - Example: musical taste and communication
    - (John, ambient) of type (T1, T2)
      (John, Paul, SMS/call) of type (T1, T1, T3)
  - That is ,tuples of atomic events, object types known

1. $\theta \sim \mathrm{DP}(\alpha)$; $\psi_z^{(t)} \sim \mathrm{Dir}(\beta^{(t)})$, $t = 1, \ldots, n(T)$;
2. For each $i \in 1, \ldots, N$:

   - $z_i \sim (\theta)$;
   - $d_i^{(j)} \sim (\psi_{z_i}^{(t_{ij})})$, $j = 1, \ldots, h_{k(C(_i))}$;

General case: estimation

- Sample latent components by

$$p(z, D_i) \propto p(z|\alpha) \prod_t \frac{\prod_{l=1}^{L_t} \prod_{n=1}^{\#\{u \in D_i | u=(t,l)\}} (g_{tzl} + 1 - n)}{\prod_{m=1}^{\#\{u \in D_i | u=(t,\text{-})\}} (g_{tz\cdot} + 1 - m)}$$

  - Gamma functions of counters are "rolled back"
  - In practice often simple
    - Compare to the document content-citation model

$$p(z, D_i| \dots) \propto \frac{\{n_z, \alpha\}}{N + \alpha} \times \begin{cases} g^{(1)}_{z,l_1} g^{(1)}_{z,l_2} / (g^{(1)}_{z,\cdot} (g^{(1)}_{z,\cdot} + 1)) & \text{for } D_i \in C_1 \\ g^{(1)}_{z,l_1} g^{(2)}_{z,l_2} / (g^{(1)}_{z,\cdot} (g^{(2)}_{z,\cdot})) & \text{for } D_i \in C_2 \end{cases}$$

Implementation

- *g*: counts *(atomic event type, comp., value)*
  - Constantly updated in sampling
  - Mostly zeroes, at least for large collections
  - E.g., a hash table or a set of hash tables
- n_z: number of objects from a component
  - Constantly updated in sampling
  - With the DP prior, components come and go
  - Self-balancing binary tree

$$p(z, D_i| \dots) \propto \frac{\{n_z, \alpha\}}{N+\alpha} \times \begin{cases} g^{(1)}_{z,l_1} g^{(1)}_{z,l_2} / (g^{(1)}_{z,\cdot}(g^{(1)}_{z,\cdot}+1)) & \text{for } D_i \in C_1 \\ g^{(1)}_{z,l_1} g^{(2)}_{z,l_2} / (g^{(1)}_{z,\cdot}(g^{(2)}_{z,\cdot})) & \text{for } D_i \in C_2 \end{cases}$$

Conclusion

- Related works: Volker Tresp's tutorial from Pascal Symposium Meeting 2008
  - IHRM (Xu et al), IRM (Kemp et al)
    HRM (Nowicki et al), stochastic block models (Airoldi et al)
- A simple relational model with wide applicability
  - Global components
    - An application: community finding, enriched graphs
  - Poisson distribution for links and other co-occurrences
    - As opposed to Bernoulli
    - Estimation does not deal explicitly with event types not in the data → sparse, fast
- To do: comparisons, estimation, latent structures