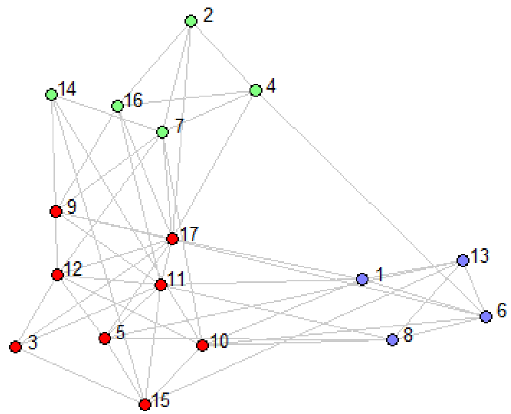
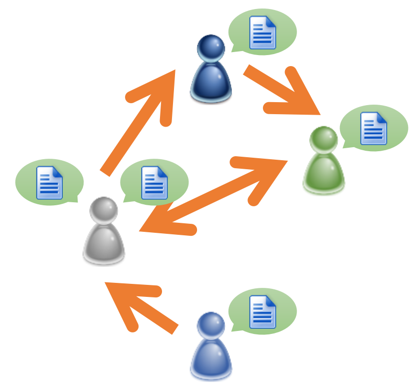




Mining Dynamic Networks with Generative Models

SDM 2021 Tutorial

Kevin S. Xu (University of Toledo)

James R. Foulds (University of Maryland, Baltimore County)

About Us

Kevin S. Xu

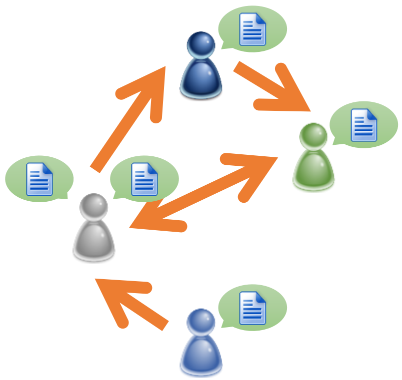
- Assistant professor at University of Toledo
- 3 years research experience in industry
- NSF CAREER award, 2021
- Research interests:
  - Machine learning
  - Network science
  - Statistical signal processing
  - Biomedical informatics

James R. Foulds

- Assistant professor at University of Maryland-Baltimore County
- NSF CAREER award, 2021
- Research interests:
  - Bayesian modeling
  - AI fairness
  - Social networks
  - Text
  - Latent variable models

Generative Models for Networks

- **Generative models** allow the creation of new data which resembles the training data
  - **Probabilistic generative models:**
    - Are **statistical models** which explicitly encode the data distribution
    - Can be used to **study and evaluate plausible mechanisms** that explain the complex system you are trying to model
    - Can **incorporate sociological theories** (etc.), and are useful for validating or invalidating those theories
    - Provide a **principled framework for model extension**, e.g. incorporating text, time, metadata…
  - **Deep generative models (e.g. GANs, VAEs):**
    - Have excellent performance in **generating realistic data** such as images
    - **Do not explicitly encode the data distribution** as a probability model
    - **Do not directly facilitate understanding** of the data
      - Not particularly amenable to encoding theories, metadata…



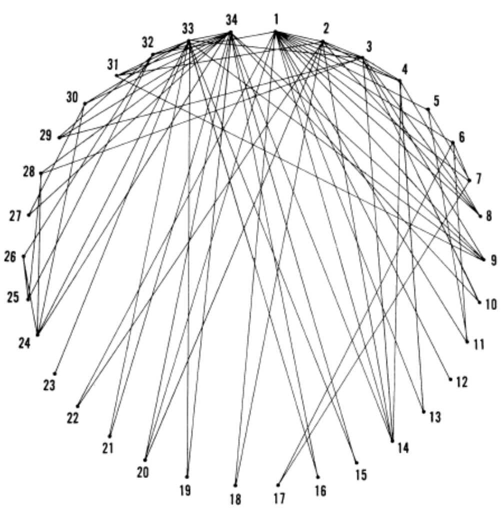
- This tutorial focuses on **probabilistic generative models** for **dynamic networks** such as social networks

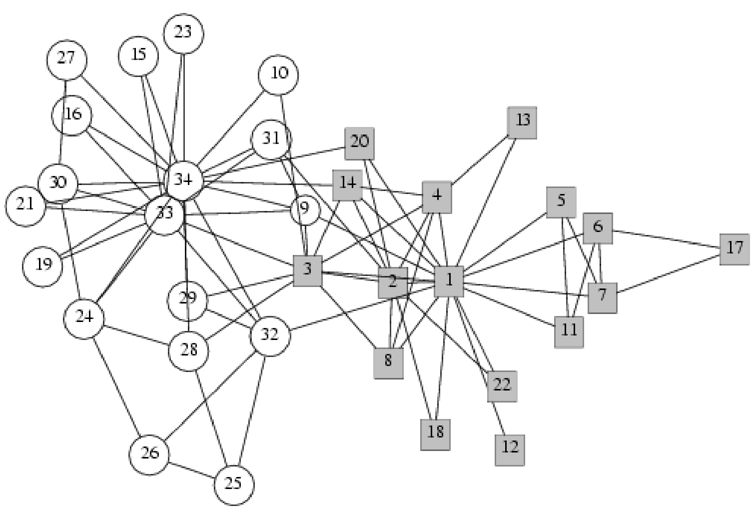
Outline

- **Mathematical representations** and **generative models** for **network data**
  - Introduction to generative approach
  - Connections to sociological principles
  - Fitting generative network models to data
  - Application scenarios with demos
- Rich generative models for **dynamics on networks** and **dynamic networks**
  - Directly-observed networks evolving in discrete and continuous time
  - Indirectly-observed networks estimated through information diffusion cascades

Social networks as graphs

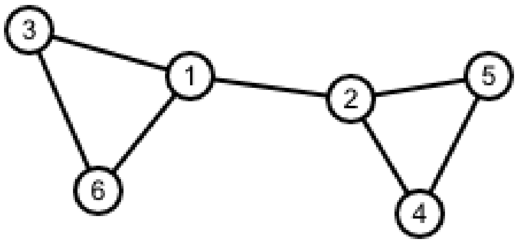
- A social network can be represented by a graph $G = (V, E)$
  - V: vertices, nodes, or actors typically representing people
  - E: edges, links, or ties denoting relationships between nodes
  - Directed graphs used to represent asymmetric relationships
- Graphs have no natural representation in a geometric space
  - Two identical graphs drawn differently
  - Moral: visualization provides very limited analysis ability
  - How do we model and analyze social network data?



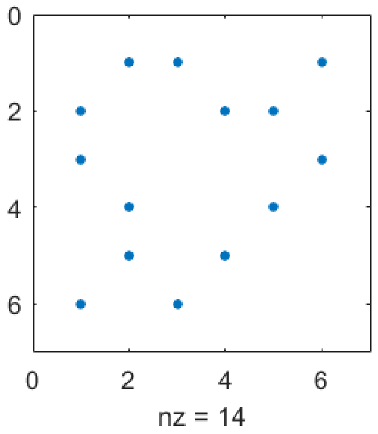


Matrix representation of social networks

- Represent graph by $n \times n$ adjacency matrix or sociomatrix $\mathbf{Y}$
  - $y_{ij} = 1$ if there is an edge between nodes i and j
  - $y_{ij} = 0$ otherwise



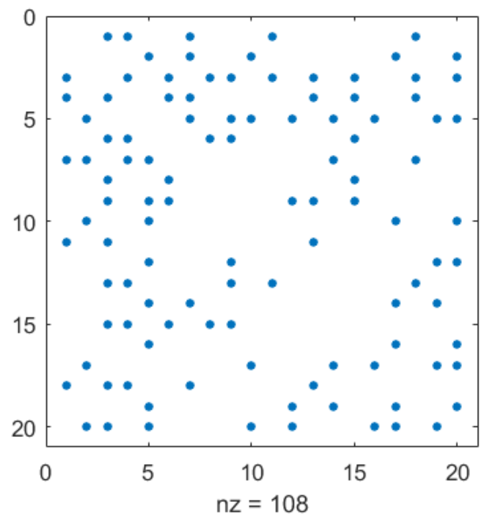
$$\mathbf{Y} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

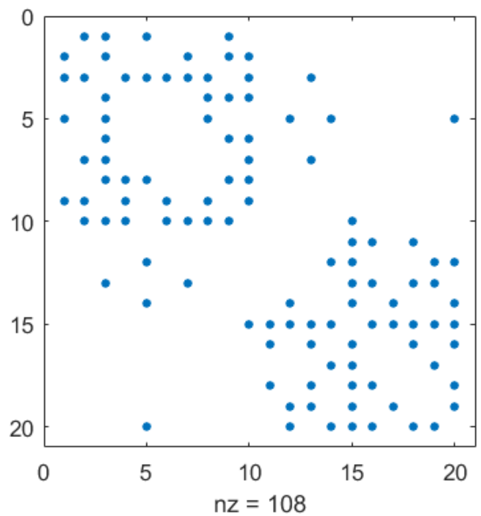


- Easily extended to directed and weighted graphs

Adjacency matrix permutation invariance

- Row and column permutations to adjacency matrix do not change graph
  - Changes only ordering of nodes
  - Provided same permutation is applied to both rows and columns
- Same graph with 2 different orderings of nodes





Sociological principles related to edge formation

- Homophily or assortative mixing
  - Tendency for individuals to bond with similar others
  - Assortative mixing by age, gender, social class, organizational role, node degree, etc.
  - Results in transitivity (triangles) in social networks
    - "My friend of my friend is my friend"
- Equivalence of nodes
  - Two nodes are structurally equivalent if their relations to all other nodes are identical
    - Approximate equivalence recorded by similarity measure
  - Two nodes are regularly equivalent if their neighbors belong to the same groups
  - Stochastic equivalence: nodes in a group have same chance of connecting to other groups

Brief history of probabilistic network models

- 1930s – Graphical depictions of social networks: sociograms (Moreno)
- 1950s – Mathematical (probabilistic) models of social networks (Erdős-Rényi-Gilbert)
- 1960s – Small world / 6-degrees of separation experiment (Milgram)
- 1980s – Introduction of statistical models: stochastic block models and precursors to exponential random graph models (Holland et al., Frank and Strauss)
- 1990s – Statistical physicists weigh in: small-world models (Watts-Strogatz) and preferential attachment (Barabási-Albert)
- 2000s-today – Machine learning approaches, latent variable models, deep neural networks
    - 2010s-today – Increasingly sophisticated **probabilistic latent variable models** for dynamics on networks and dynamic networks. Deep neural networks for graph embeddings and predictive tasks, e.g. link prediction

Generative models for social networks

- A generative model for social networks is one that can simulate new networks
- Two distinct schools of thought:
  - Probability models (non-statistical)
    - Typically simple, 1-2 parameters, not typically learned from data
    - Can be studied analytically
  - Statistical models
    - More parameters, latent variables
    - Learned from data via statistical estimation techniques
- Note: deep generative models such as generative adversarial networks and variational autoencoders are another class of generative models.
  - These models usually require many i.i.d. samples, so are not typically used for modeling networks
- Many other deep neural network models are used for networks, e.g. DeepWalk, Node2Vec, LINE, GNN, but these models are **not generative**

Statistical models for networks

- Statistical models try to represent networks using a larger number of parameters to capture properties of a specific network
- Exponential random graph models
- Latent variable models
    - Latent space models
    - Stochastic block models
    - Mixed-membership stochastic block models
    - Latent feature models

Latent variable models for social networks

- Model where observed variables are dependent on a set of unobserved or latent variables
  - Observed variables assumed to be conditionally independent given latent variables
- Why latent variable models?
  - Adjacency matrix $\mathbf{Y}$ is invariant to row and column permutations
  - Aldous-Hoover theorem implies existence of a latent variable model of form

    $$y_{ij} = h(\theta, z_i, z_j, \epsilon_{ij})$$

    for iid latent variables z_i and some function h

Latent variable models for social networks

- Latent variable models allow for heterogeneity of nodes in social networks
  - Each node (actor) has a latent variable $\mathbf{z}_i$
  - Probability of forming edge between two nodes is independent of all other node pairs given values of latent variables

$$p(\mathbf{Y}|\mathbf{Z},\theta) = \prod_{i \neq j} p\left(y_{ij}\middle|\mathbf{z}_i, \mathbf{z}_j, \theta\right)$$

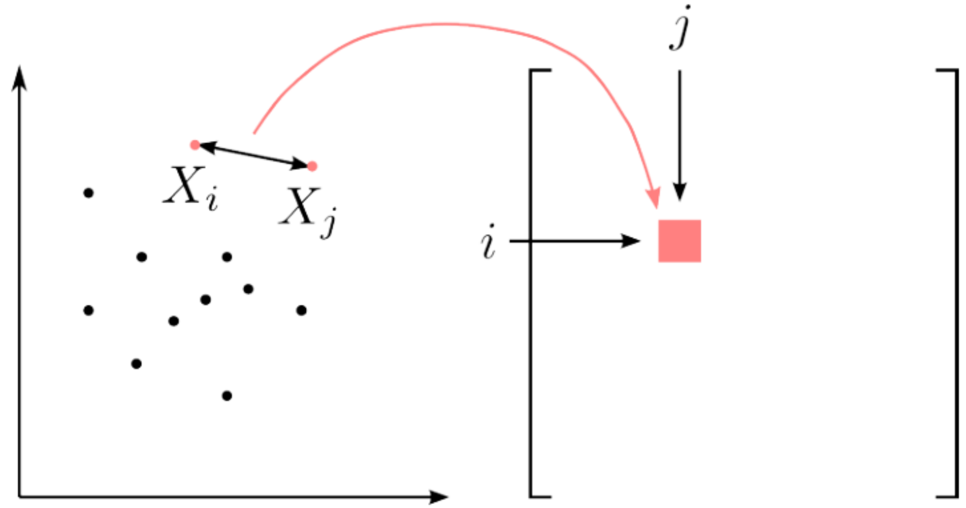
  - Ideally latent variables should provide an interpretable representation

(Continuous) latent space model

- Motivation: homophily or assortative mixing
  - Probability of edge between two nodes increases as characteristics of the nodes become more similar
- Represent nodes in an unobserved (latent) space of characteristics or “social space”
- Small distance between 2 nodes in latent space ➔ high probability of edge between nodes
  - Induces transitivity: observation of edges (i, j) and (j, k) suggests that i and k are not too far apart in latent space ➔ more likely to also have an edge

(Continuous) latent space model

- (Continuous) latent space model (LSM) proposed by Hoff et al. (2002)
  - Each node has a latent position $\mathbf{z}_i \in \mathbb{R}^d$
  - Probabilities of forming edges depend on distances between latent positions
  - Define pairwise affinities $\psi_{ij} = \theta - \|\mathbf{z}_i - \mathbf{z}_j\|_2$



Latent space model: generative process

1. Sample node positions in latent space

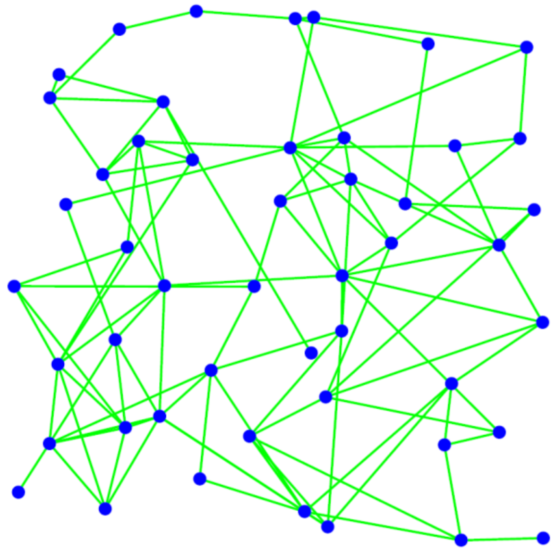
$$\mathbf{z}_i \sim \text{Gaussian}(\mathbf{0}, \kappa\mathbf{I})$$

2. Compute affinities between all pairs of nodes

$$\psi_{ij} = \theta - \|\mathbf{z}_i - \mathbf{z}_j\|_2$$

3. Sample edges between all pairs of nodes

$$P(Y_{ij} = 1|\psi_{ij}) = \sigma(\psi_{ij})$$



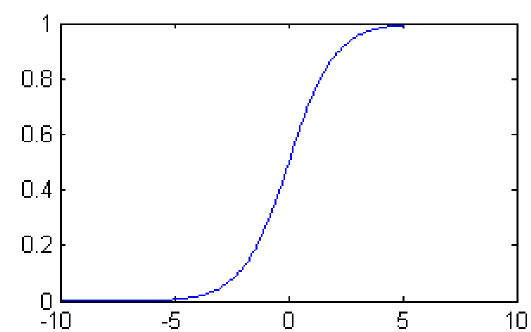


Figure due to P. D. Hoff, Modeling homophily and stochastic equivalence in symmetric relational data, NIPS 2008

Advantages and disadvantages of latent space model

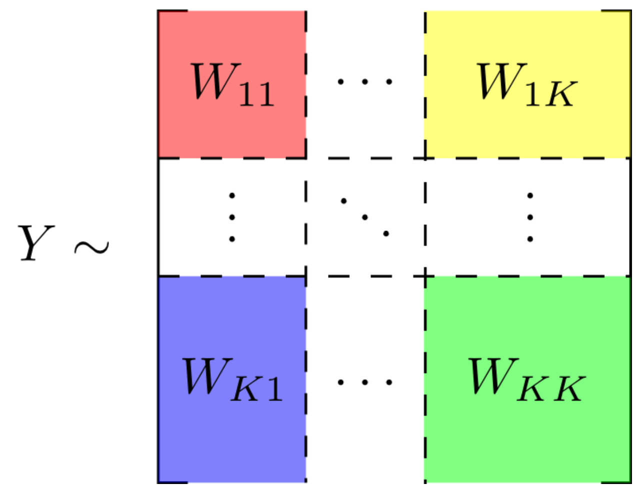
- Advantages of latent space model
  - Visual and interpretable spatial representation of network
  - Models homophily (assortative mixing) well via transitivity
- Disadvantages of latent space model
  - 2-D latent space representation often may not offer enough degrees of freedom
  - Cannot model disassortative mixing (people preferring to associate with people with different characteristics)

Related Models: Node Embeddings

- The neural network community has more recently developed similar models to the latent space model
- **Node embedding:** the task of learning vector space representations for nodes
  - Defined without reference to an explicit fully generative model such as the latent space model
  - Typical models: model nearby nodes encountered via a particular sampling strategy (differs per model)
    - **DeepWalk (Perozzi 2014), LINE (Tang 2015), Node2vec (Grover 2016)**
  - These models were inspired by word2vec from natural language processing
  - See also **Graph Convolutional Networks (Kipf 2017):** extend CNNs to graphs, originally designed for semi-supervised classification but also learns node embeddings
- These models are much more scalable than the latent space model, but lack generative semantics

Stochastic block model (SBM)

- First formalized by Holland et al. (1983)
- Also known as multi-class Erdős-Rényi model
- Each node has categorical latent variable $z_i \in \{1, \dots, K\}$ denoting its class or group
- Probabilities of forming edges depend on class memberships of nodes ($K \times K$ matrix W)
  - Groups often interpreted as functional roles in social networks



Stochastic equivalence and block models

- Stochastic equivalence: generalization of structural equivalence
- Group members have identical probabilities of forming edges to members other groups
  - Can model both assortative and disassortative mixing

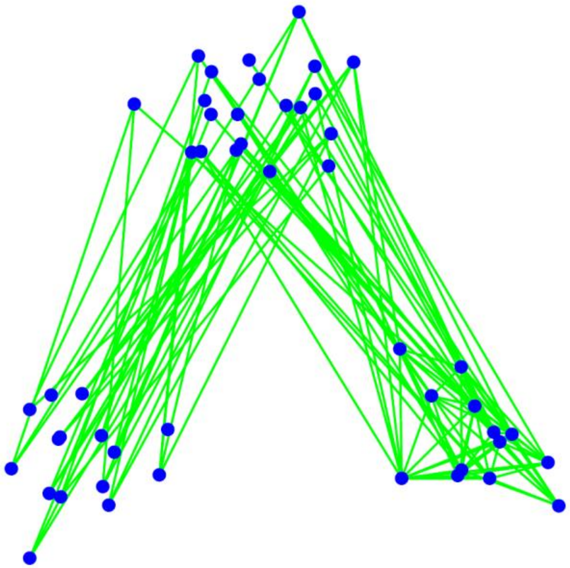
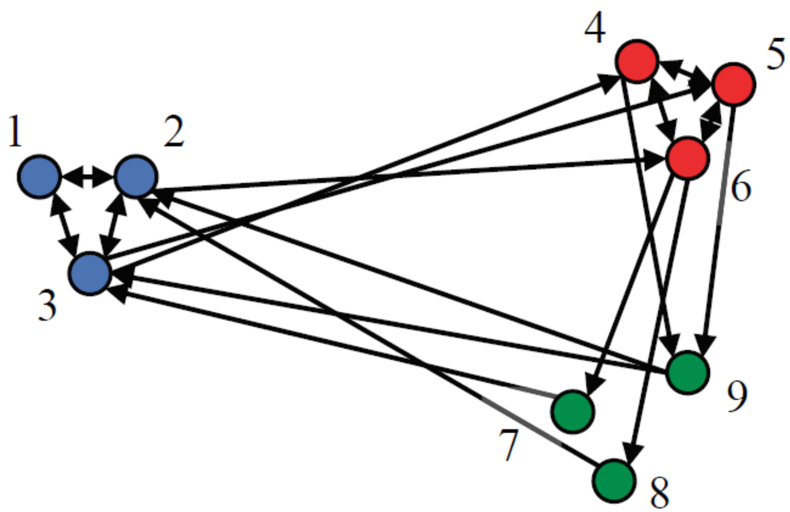


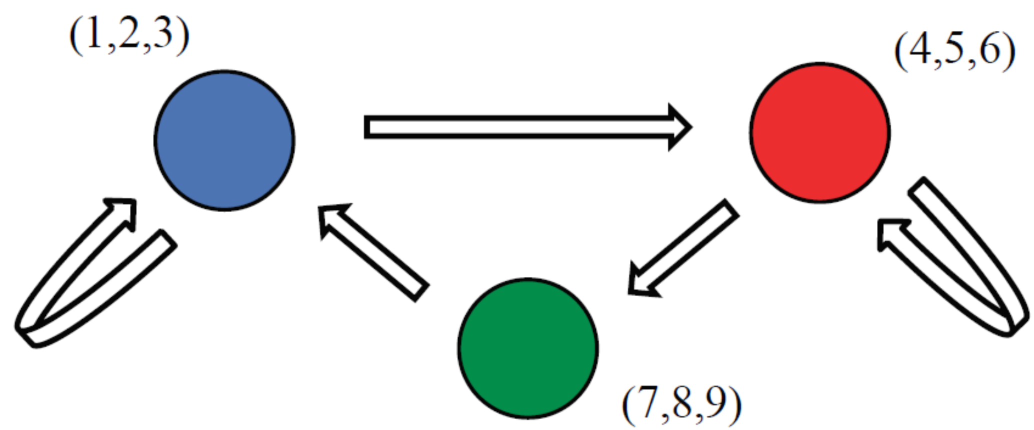
Figure due to P. D. Hoff, Modeling homophily and stochastic equivalence in symmetric relational data, NIPS 2008

Stochastic equivalence vs community detection

Original graph



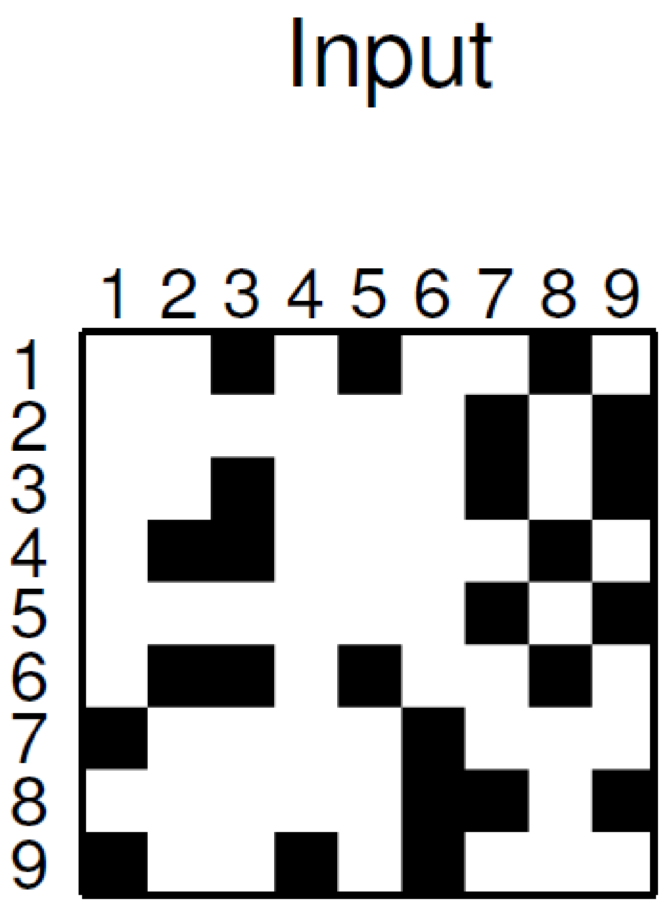
Blockmodel

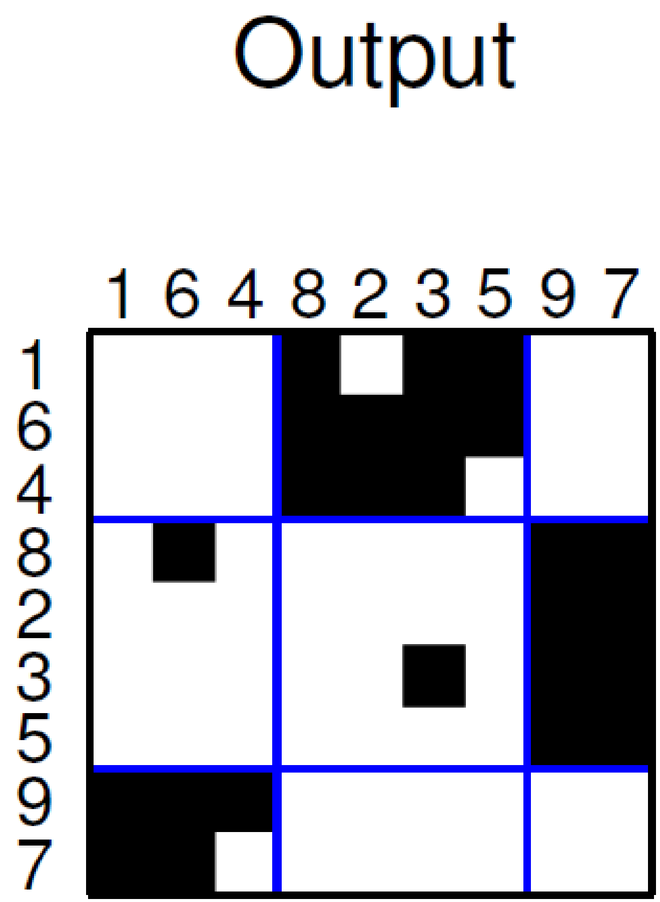


Stochastically equivalent, but
are not densely connected

Figure due to Goldenberg et al. (2009) - Survey of Statistical Network Models, Foundations and Trends

Reordering the matrix to show the inferred block structure

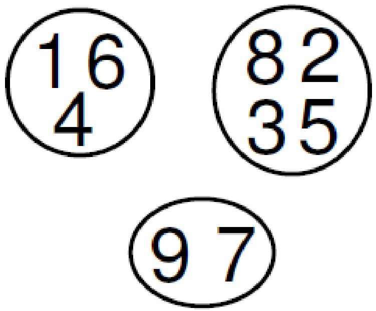




Kemp, Charles, et al. "Learning systems of concepts with an infinite relational model." AAAI. Vol. 3. 2006.

Model structure

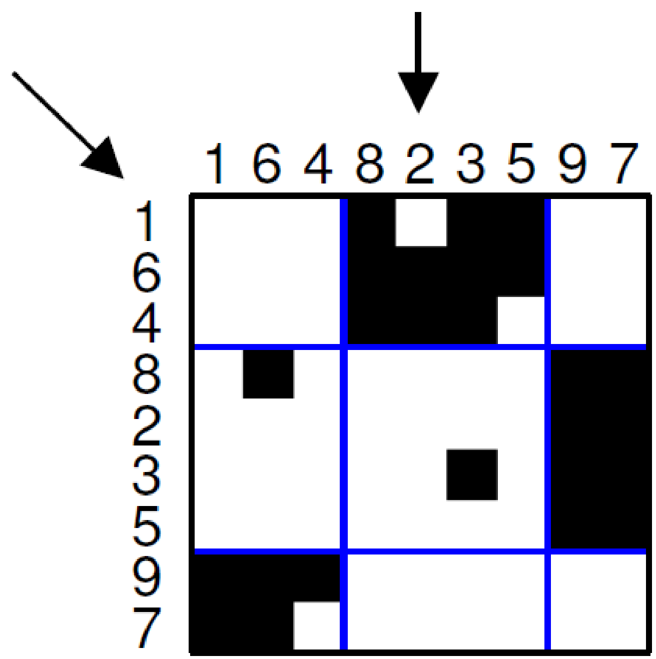
Latent groups Z



0.1	0.9	0.1
0.1	0.1	0.9
0.9	0.1	0.1

Interaction matrix W

(probability of an edge from block k to block k')



Kemp, Charles, et al. "Learning systems of concepts with an infinite relational model." AAAI. Vol. 3. 2006.

Stochastic block model generative process

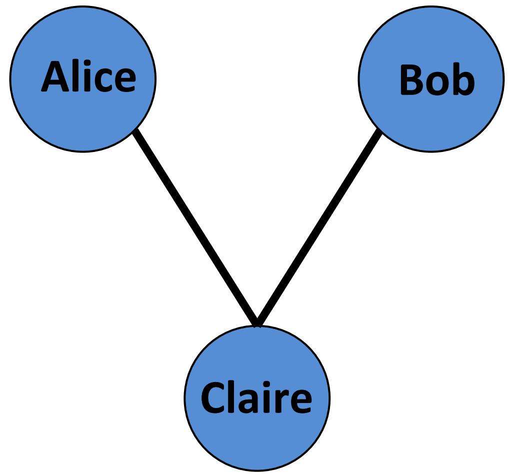
$W_{kk'}$: Probability that a node in group k connects to a node in k'

z_i: Latent group assignment for node i

For each pair of nodes (i, j)

$$Y_{ij} \sim \text{Bernoulli}(W_{z_i, z_j})$$

Stochastic block model
Latent representation



Nodes assigned to only one latent group.

Not always an appropriate assumption

$\mathbf{Z} =$

	Running	Dancing	Fishing
Alice	1		
Bob			1
Claire		1	

Alternative representations to stochastic block model

$\mathbf{Z} =$

	Running	Dancing	Fishing
Alice	1		
Bob			1
Claire		1	

Stochastic Block Model
(1 group per node)

$\mathbf{Z} =$

	Running	Dancing	Fishing
Alice	0.4	0.4	0.2
Bob	0.5	0.5	
Claire	0.1		0.9

Mixed membership stochastic Block Model
(Partial group membership, Sums to 1 per node)

Binary latent feature model

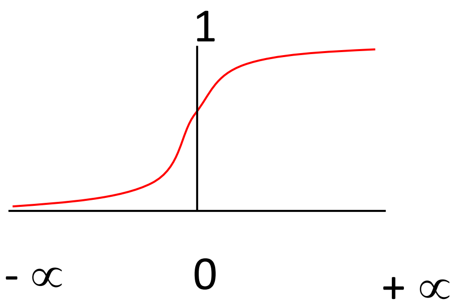
$\mathbf{Z} =$

	Cycling	Fishing	Running	Tango	Salsa	Waltz
Alice						
Bob						
Claire						

Latent feature models

- Latent Feature Relational Model LFRM (Miller, Griffiths, Jordan, 2009) likelihood model:

$$P(Y_{ij} = 1 | \ldots) = \sigma(\mathbf{z}_i \mathbf{W} \mathbf{z}_j^\intercal)$$



- "If I have feature *k*, and you have feature *l*, add W_{kl} to the log-odds of the probability we interact"
- Can include terms for network density, covariates, popularity, etc.

Frequentist inference

- Parameters θ treated as having fixed but unknown values
  - Stochastic block model parameters: class memberships $\mathbf{Z}$ and block-dependent edge probabilities $\mathbf{W}$
  - Latent space model parameters: latent node positions $\mathbf{Z}$ and scalar global bias θ
- Estimate parameters by maximizing likelihood function of the parameters

$$\hat{\theta}_{MLE} = \mathrm{argmax}_{\theta}\, Pr(\mathbf{X}|\theta)$$

Bayesian inference

- Parameters θ treated as random variables. We can then take into account uncertainty over them

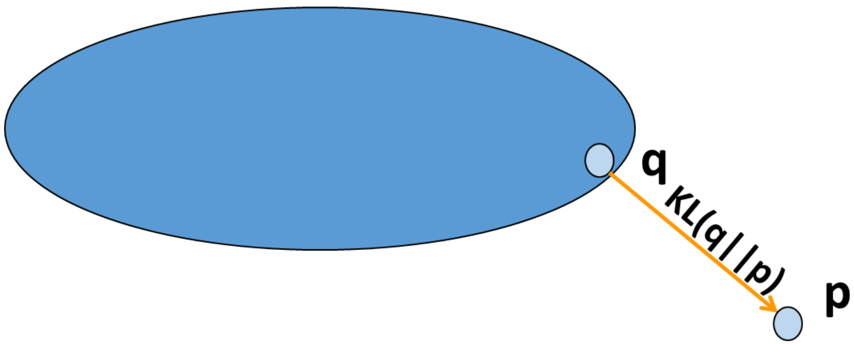
- As a Bayesian, all you have to do is write down your prior beliefs, write down your likelihood, and apply Bayes ' rule,

$$Pr(\theta|\mathbf{X}) = \frac{Pr(\mathbf{X}|\theta)Pr(\theta)}{Pr(\mathbf{X})}$$

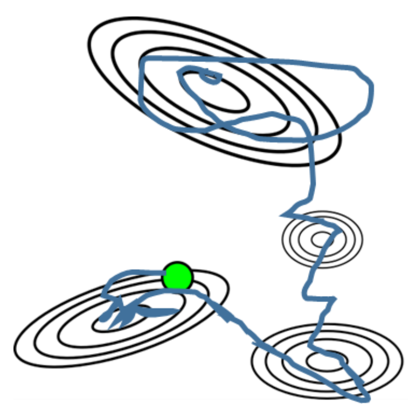
Inference Algorithms

- **Exact inference**
  - Generally intractable

- **Approximate inference**
  - **Optimization approaches**
    - variational inference, EM, variational autoencoders



  - **Simulation approaches**
    - Markov chain Monte Carlo, importance sampling, particle filtering



Variational inference vs MCMC Pros and cons

- **Pros:**
  - Deterministic algorithm, typically converges faster
  - Stochastic algorithms can scale to very large data sets
  - No issues with checking convergence

- **Cons:**
  - Variational assumptions limit possible performance
  - Will never converge to the true distribution, unlike MCMC

Evaluation of unsupervised models

- **Quantitative** evaluation
  - Measurable, quantifiable performance metrics

- **Qualitative** evaluation
  - Exploratory data analysis (EDA) using the model
  - Human evaluation, user studies,…

Evaluation of unsupervised models

- **Intrinsic** evaluation
  - Measure inherently good properties of the model
    - Fit to the data (e.g. link prediction), interpretability,…
- **Extrinsic** evaluation
  - Study usefulness of model for external tasks
    - Classification, retrieval, part of speech tagging,…

Extrinsic evaluation: What will you use your model for?

- If you have a **downstream task** in mind, you should probably evaluate based on it!

- Even if you don't, you could contrive one for evaluation purposes

- E.g. use latent representations for:
  - Classification, regression, retrieval, ranking…

Posterior predictive checks

- Sampling data from the posterior predictive distribution allows us to "look into the mind of the model" – G. Hinton

"This use of the word *mind* is not intended to be metaphorical. We believe that a mental state is the state of a hypothetical, external world in which a high-level internal representation would constitute veridical perception. That hypothetical world is what the figure shows." **Geoff Hinton et al. (2006), A Fast Learning Algorithm for Deep Belief Nets.**

Posterior predictive checks

- Does data drawn from the model differ from the observed data, in ways that we care about?

- PPC:
  - Define a discrepancy function (a.k.a. test statistic) T(**X**).
    - Like a test statistic for a p-value. How extreme is my data set?
  - Simulate new data $\mathbf{X}^{(rep)}$ from the posterior predictive
    - Use MCMC to sample parameters from posterior, then simulate data
  - Compute $T(\mathbf{X}^{(rep)})$ and T(**X**), compare. Repeat, to estimate:

$$\mathrm{PPC} = P(T(\mathbf{X}^{(rep)}) > T(\mathbf{X})|\mathbf{X})$$

Python code for demos available on tutorial website

https://github.com/kevin-s-xu/SDM-2021-Generative-Tutorial

Demo 1: Facebook wall posts with static network model

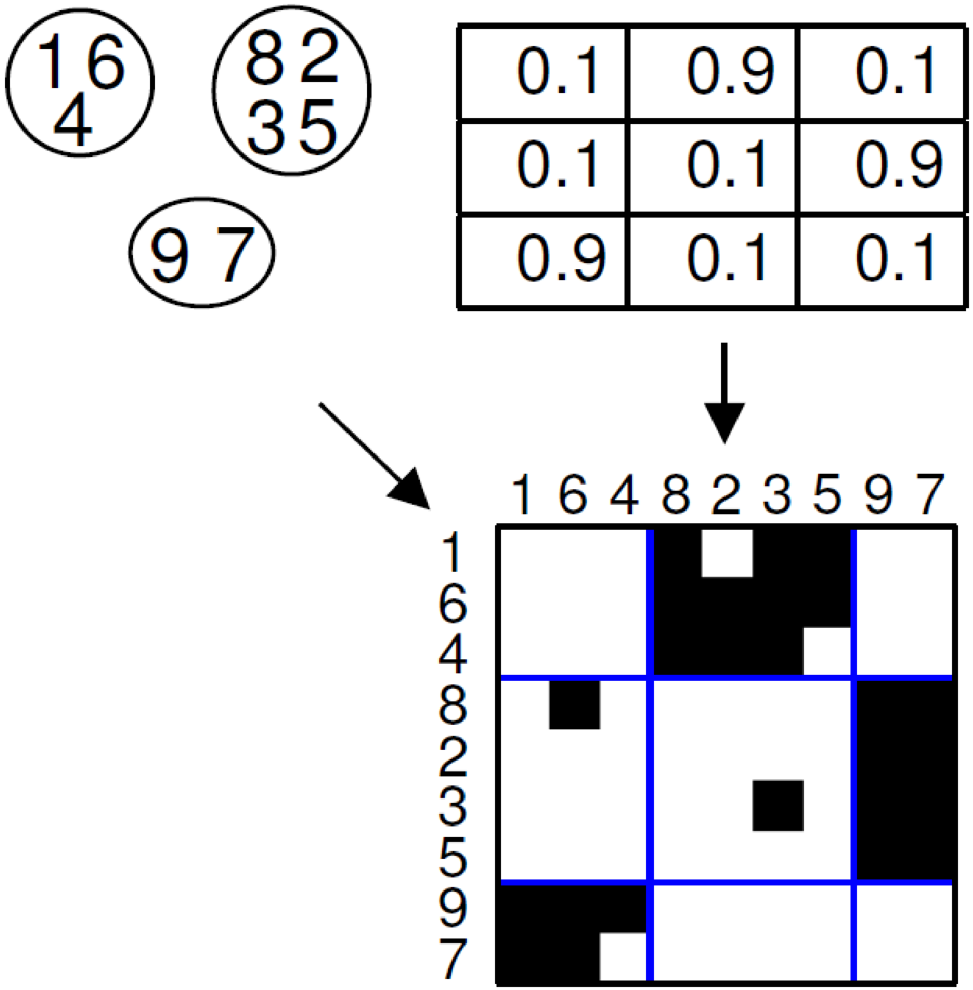
- Network of wall posts on Facebook collected by Viswanath et al. (2009)
  - Nodes: ~44,000 Facebook users
  - Edges: directed edge from i to j if i posts on j's Facebook wall
- We analyze a subset of the network to speed up demo
  - Methods we use scale up to ~100,000 nodes
- We will fit a stochastic block model to the data

Model structure

Latent groups Z

Interaction matrix W

(probability of an edge from block k to block k')



Kemp, Charles, et al. "Learning systems of concepts with an infinite relational model." AAAI. Vol. 3. 2006.

Fitting stochastic block model

- A priori block model: assume that class (role) of each node is given by some other variable
  - Only need to estimate $W_{kk'}$: probability that node in class k connects to node in class k' for all k, k'
- Likelihood given by

Number of actual edges in block (k, k')

Number of possible edges in block (k, k')

$$Pr(\mathbf{Y}|\mathbf{W},\mathbf{Z})$$
$$= \exp\left\{\sum_{k=1}^{K}\sum_{k'=1}^{K}\left[m_{kk'}\log W_{kk'} + (n_{kk'} - m_{kk'})\log(1 - W_{kk'})\right]\right\}$$

- Maximum-likelihood estimate (MLE) given by

$$\hat{W}_{kk'} = \frac{m_{kk'}}{n_{kk'}}$$

Estimating latent classes

- Latent classes (roles) are unknown in this data set
  - First estimate latent classes $\mathbf{Z}$ then use MLE for $\mathbf{W}$
- MLE over latent classes is intractable!
  - $\sim K^N$ possible latent class vectors
- Spectral clustering techniques have been shown to accurately estimate latent classes
  - Use singular vectors of (possibly transformed) adjacency matrix to estimate classes
  - Run K-means clustering on rows of singular vector matrix
  - Many variants with differing theoretical guarantees

Demo 1: SBM on Facebook wall post network

Conclusions from posterior predictive check

- Block densities are well-replicated
- Reciprocity is not replicated
  - Pair-dependent stochastic block model can be used to preserve reciprocity

$$p(\mathbf{Y}|\mathbf{Z},\theta) = \prod_{i \neq j} p\left(y_{ij}, y_{ji} \middle| \mathbf{z}_i, \mathbf{z}_j, \theta\right)$$

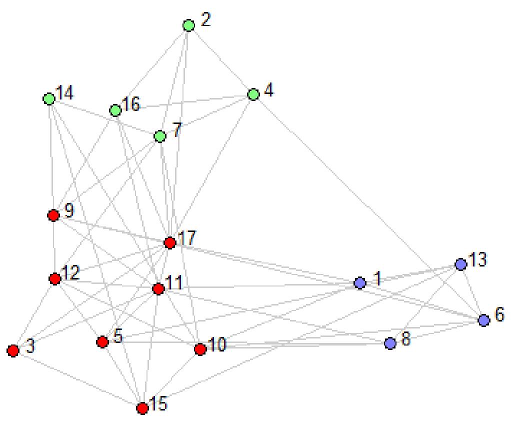
  - 4 choices for pair or dyad: $\left(y_{ij}, y_{ji}\right) \in \{(0,0),(0,1),(1,0),(1,1)\}$
- Transitivity is not replicated
  - No mechanism for transitivity in SBM so this is a limitation of the model
  - Can use latent space models to get transitivity

Outline

- **Mathematical representations** and **generative models** for **network data**
  - Introduction to generative approach
  - Connections to sociological principles
  - Fitting generative network models to data
  - Application scenarios with demos
- Rich generative models for **dynamics on networks** and **dynamic networks**
  - Directly-observed networks evolving in discrete and continuous time
  - Indirectly-observed networks estimated through information diffusion cascades

Dynamic networks

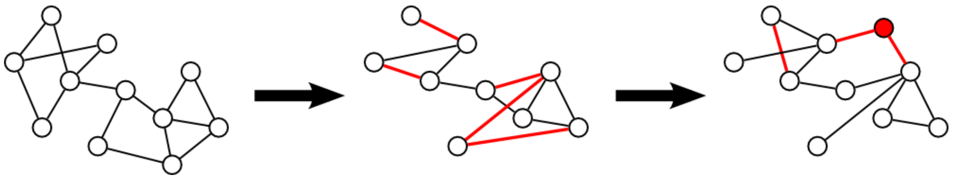
- Directly-observed relations (edges) between nodes that may change over time
- Need to generalize social network models to account for dynamics of edges



Dynamic social network
(Nordlie, 1958; Newcomb, 1961)

Continuous vs. discrete-time network data

- Discrete-time: network snapshots at discrete time steps
  - Nodes and edges can both appear and disappear over time
  - Also known as network panel data
- Continuous-time: relational events with fine-grained timestamps
  - Facebook wall posts (Viswanath et al., 2009)
  - Can be represented by triplets (u, v, t)
  - Also known as timestamped relational event data



Sender	Receiver	Timestamp
1595	1021	1100626783
4581	5626	1100627183
3806	991	1100640075
521	533	1100714520
521	3368	1100716404
8734	527	1100724840
1017	1015	1100828851
17377	1021	1100832283
2926	726	1100838067

Continuous vs. discrete-time network models

- Discrete-time network model: models edge-forming probabilities $p(u, v, t)$ for discrete time steps t, e.g. $t \in \{1, 2, \dots, T\}$
  - Apply to discrete-time network data: choose discrete time steps to match time snapshots of data
  - Apply to continuous-time network data: aggregate event timestamps over time windows to create snapshots
- Continuous-time network model: models edge-forming probabilities $p(u, v, t)$ for arbitrary $t \in \mathbb{R}$
  - Apply to continuous-time network data: treat event timestamps as real-valued (assuming fine-grained)
  - Apply to discrete-time network data: may be possible despite only having information at discrete times
    - Could benefit from having counts of number of events

Continuous vs. discrete-time comparison matrix

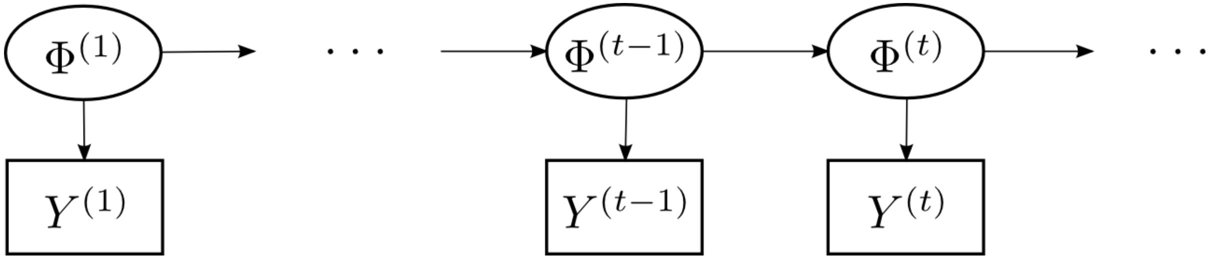
Data \ Model	Discrete	Continuous

Discrete	• Integrate network model with sequence model
	    • Hidden Markov model (HMM) and variants
	• Fit integrated model to discrete-time network data
		• Integrate network model with temporal point process (TPP) model
		• Fit integrated model to discrete-time network data
		    • Event counts can help

Continuous	• Aggregate to form snapshots • Integrate network model with sequence model • Fit integrated model to discrete-time network data	• Integrate network model with temporal point process (TPP) model • Hawkes process and variants • Fit integrated model to continuous-time network data

Discrete-time modeling approach

- Most work on discrete-time network modeling assumes hidden Markov structure
  - Latent variables and/or parameters follow Markov dynamics
  - Graph snapshot at each time generated using static network model, e.g. stochastic block model



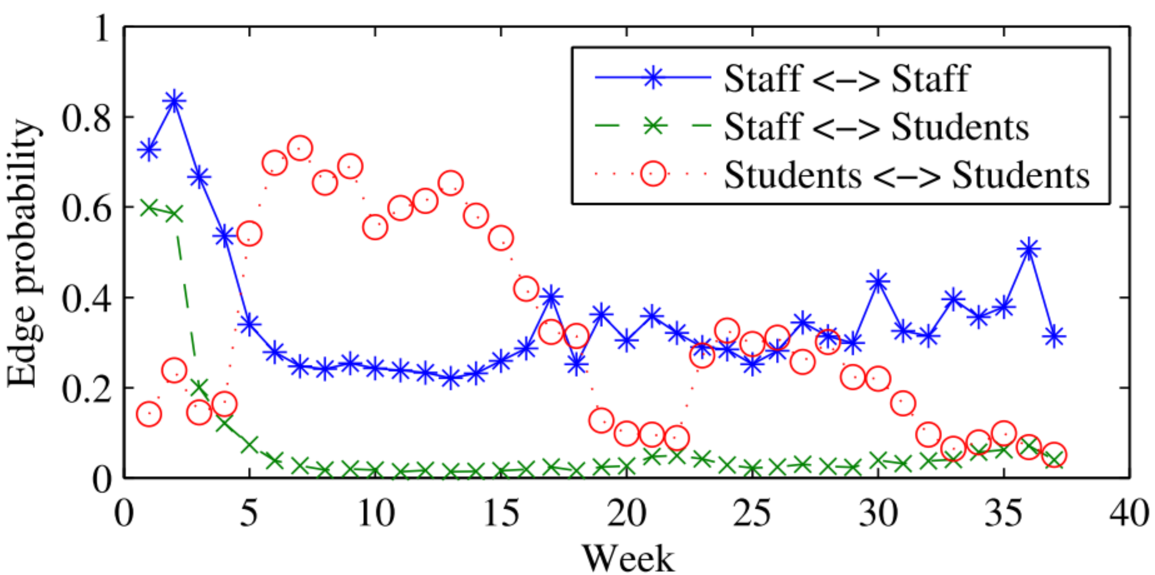
- Can analyze discrete-time network data without generative models
  - Apply static network mining algorithms snapshot by snapshot
- Mining discrete-time networks using generative models can produce more accurate and interpretable insights

Discrete-time network models

- HMM-based structure has been used to extend many static network models to discrete time
  - Stochastic block model
  - Latent space model
  - Mixed-membership stochastic block model
  - Latent feature model
- Time dependence in model further complicates estimation
- Estimation approaches for hidden Markov SBMs
  - Variational inference (Yang et al., 2011; Matias and Miele, 2017)
  - MCMC (Yang et al., 2011)
  - Greedy algorithms, e.g. hill climbing (Xu and Hero, 2014; Corneli et al., 2016)

MIT Reality Mining network

- Dynamic social network of physical proximity using Bluetooth between 94 students and staff at MIT over 2 semesters (Eagle et al., 2009)
- HM-SBM (Xu and Hero, 2014) reveals differences in temporal dynamics of 2 classes



N. Eagle et al. Inferring friendship network structure by using mobile phone data. *PNAS*, 106(36):15274–15278, 2009.

Continuous-time modeling approach

- Combine a latent variable representation for the nodes with a temporal point process for the event times
  - Often called timestamped relational event models
- It is difficult to analyze continuous-time (e.g. timestamped) network data without generative models!
  - Smaller number of analysis tools for direct analysis, e.g. temporal motifs
  - Can use some continuous-time network embeddings
  - Can construct snapshots to revert to discrete time setting
- Continuous-time generative models enable direct analysis of timestamped network data
  - Allows fine-grained analysis of bursty behavior over time

Temporal point processes (TPPs)

- A temporal point process is characterized by an intensity function
  - **Intensity function:** the expected number of events in a small time window (roughly speaking)

$$\lambda_v(t|H) = \lim_{\Delta t \to 0} \frac{\mathbb{E}[N_v(t+\Delta t) - N_v(t)|H_t)]}{\Delta t}$$

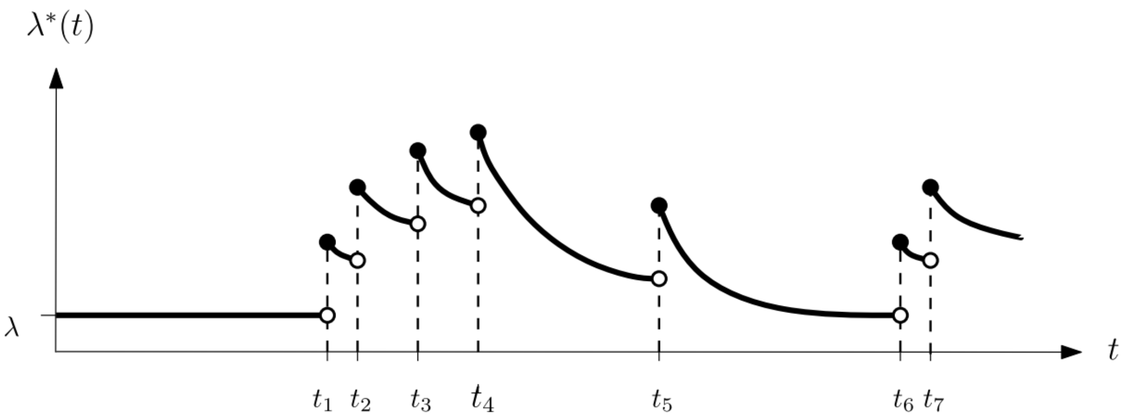
- Common temporal point processes
  - Inhomogeneous Poisson process: process intensity does not change based on generated events
    - Can be used to model diurnal patterns, etc.
    - Can be used to estimate intensity based on observed events – but not fully generative!
  - Hawkes process: process intensity changes based on generated events
    - Intensity function is also a random process!

Hawkes process

- Self-exciting temporal point process
  - Arrival of an event increases the chance of next event arrival immediately after
  - Fully generative inhomogeneous Poisson process
- Conditional intensity function (given event times)

$$\lambda(t) = \mu + \sum_{t_i<t} \alpha\gamma(t - t_i)$$

  - Kernel or excitation function $\gamma(\cdot)$ controls decay rate
  - Exponential kernel is mathematically convenient and has been found to be a good fit on many data types



$$\lambda(t) = \mu + \sum_{t_i<t} \alpha e^{-\beta(t - t_i)}$$

α: Jump size

μ: baseline intensity

β: exponential decay rate

Continuous-time network models

- Latent variable model + Hawkes process has been used to extend many static network models to continuous time relational events
  - Latent space models
  - Stochastic block models
- Timestamped relational event models: directly observe network through events between all $O(n^2)$ node pairs
  - Use univariate or bivariate Hawkes process to model events between each node pair (u, v)

$$\lambda_{uv}(t) = \mu_{uv} + \sum_{e:t_e^{uv}<t} \alpha_{uv} f_\Delta(t - t_e^{uv}) + \sum_{e:t_e^{vu}<t} \alpha_{vu} f_\Delta(t - t_e^{vu})$$

  - Use static network model to obtain a lower-dimensional representation of the $n \times n$ matrices for $\mu_{uv}, \alpha_{uv}, \alpha_{vu}$

Continuous-time network models

- Latent space model (Yang et al., 2017)
  - Bivariate Hawkes process model for each node pair (u, v)
  - Use separate latent spaces for homophily (base intensity) and reciprocity (excitation)
  - Conditional intensity function

$$\lambda_{uv}(t) = \gamma\, e^{-\|\mathbf{z}_u - \mathbf{z}_v\|_2^2} + \sum_{k:\, t_k^{vu} < t} \sum_{b=1}^{B} \beta\, e^{-\|\mathbf{x}_u^{(b)} - \mathbf{x}_v^{(b)}\|_2^2}\, \phi_b(t - t_k^{vu})$$

  - Fit model using MAP estimation
- Stochastic block model
  - Different block pairs have different Hawkes process parameters
  - Estimation approaches
    - Bayesian: MCMC, variational inference
    - Frequentist: spectral clustering, greedy algorithms

Yang, J., Rao, V., & Neville, J. (2017). *UAI.*

Hawkes process block models

- Block Hawkes model (Junuthula et al., 2019)
  1. Assign each node i to block $a \in \{1, \dots, K\}$ with probability π_a
  2. For each block pair (a, b), generate event timestamps from an independent univariate Hawkes process with parameters $(\mu_{ab}, \alpha_{ab}, \beta_{ab})$
     - Randomly choose nodes in blocks to "receive" edges (thinning)
  - Can replace independent univariate processes in step 2 with a K^2-variate Hawkes process to model reciprocity between blocks: Hawkes IRM (Blundell et al., 2012)
- Community Hawkes Independent Pairs (CHIP) model (Arastuie et al., 2020)
  1. Assign each node i to block $a \in \{1, \dots, K\}$ with probability π_a
  2. For each node pair (i, j) in community pair (c_i, c_j), generate event timestamps from an independent Hawkes process with shared parameters $\left(\mu_{c_i c_j}, \alpha_{c_i c_j}, \beta_{c_i c_j}\right)$
  - Can replace independent univariate processes with bivariate processes to model reciprocity between nodes (Miscouridou et al., 2018)

Junuthula, R. R., Haghdan, M., Xu, K. S., & Devabhaktuni, V. K. (2019). *TheWebConf.*
Blundell, C., Heller, K. A., & Beck, J. (2012). *NeurIPS.*
Arastuie, M., Paul, S., & Xu, K. S. (2020). *NeurIPS.*
Miscouridou, X., Caron, F., & Teh, Y. W. (2018). *NeurIPS.*

Demo 2: Facebook wall posts with continuous-time network model

- Network of wall posts on Facebook collected by Viswanath et al. (2009)
  - Nodes: ~44,000 Facebook users
  - **Timestamped edges:** directed edge from i to j at time t when i posts on j's Facebook wall
- We analyze a subset of the network to speed up demo
  - Methods we use scale up to ~100,000 nodes
- We will fit the to the Community Hawkes Independent Pairs (CHIP) model to the data

CHIP estimation procedure

1. Community detection: run spectral clustering on weighted adjacency matrix of event counts N
2. Estimate Hawkes process parameters μ_{ab} and $m_{ab} = \alpha_{ab}/\beta_{ab}$ for all community pairs a, b

$$\hat{\mu}_{ab} = \frac{1}{T}\sqrt{\frac{(\bar{N}_{ab})^3}{S_{ab}^2}}, \qquad \hat{m}_{ab} = 1 - \sqrt{\frac{\bar{N}_{ab}}{S_{ab}^2}}$$

Sample mean

Unbiased sample variance

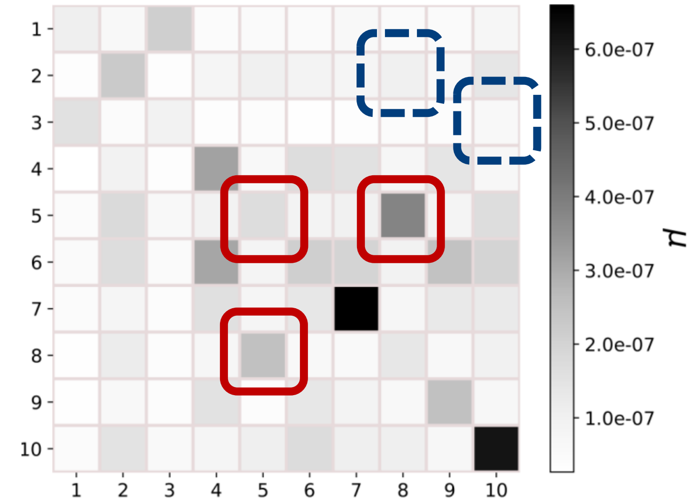
3. (Optional) To recover α_{ab}, β_{ab} separately, substitute $\hat{\mu}_{ab}, \hat{m}_{ab}$ into standard Hawkes process log-likelihood to isolate β_{ab}
   - Run a scalar optimization (line search) on β_{ab}

- Steps 1 & 2 use only counts of events!
  - Could fit a model parameterized by just μ, m from event counts!
- Step 3 requires event timestamps
- Estimation procedure scales to networks with ~100,000 nodes!
  - Estimated community assignments and Hawkes process parameters μ, m are consistent as $n \to \infty, T \to \infty$

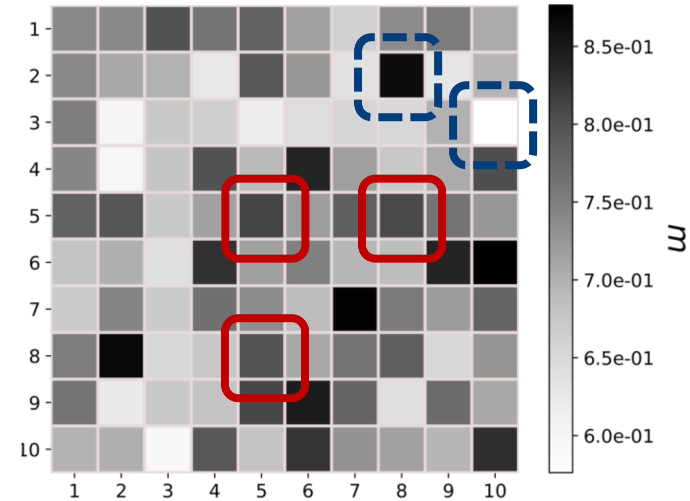
Demo 2: Facebook wall posts with CHIP continuous-time model

Results on full Facebook wall post network

- ~140 seconds to fit CHIP model with 10 blocks on ~44,000 nodes and ~850,000 events
- Not all blocks have higher rates of within-block posts, e.g. $\mu_{5,8} > \mu_{5,5}$ and $\mu_{8,5} > \mu_{5,5}$ with comparable α, β
- Some block pairs behave like a Poisson process, e.g. (3,10)
- Others are extremely bursty, e.g. (2, 8)



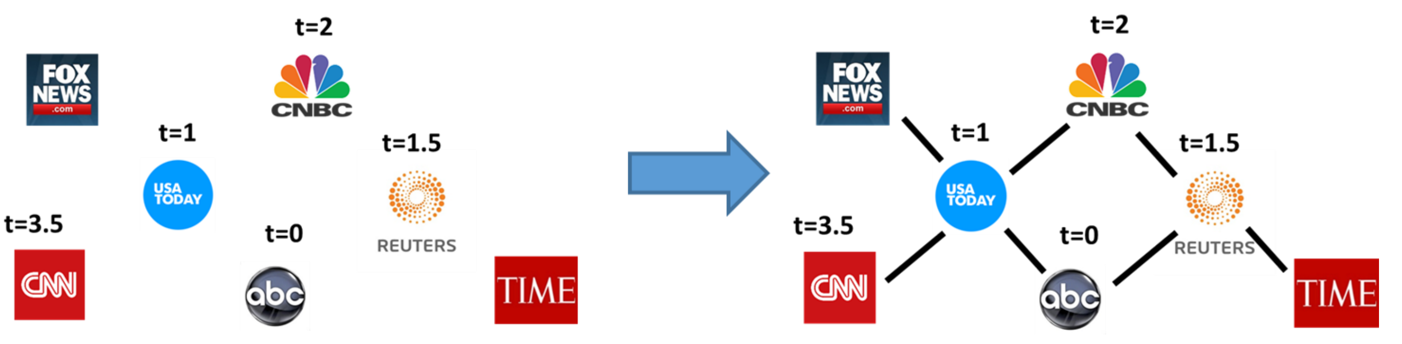
(a) μ: base intensity



(b) m: α to β ratio

Dynamics on networks: diffusion and cascades

- Information diffusion is analogous to virus propagation, a spread along a social network
- Information cascades:
  - **Given:** the times of “infection” with information (news, blogs, memes, misinformation)
  - **Infer:** underlying **latent social network** – who “infects” or influences
  - Applications: viral marketing, propaganda / misinformation detection, citation networks whom?



- *We need a generative model, since the network is not observed!*

Dynamics on networks: diffusion and cascades

- A recent survey (Zhou et al., 2021) categorizes information diffusion models into:
  - feature-based models (engineer temporal, content, structural features, etc)
  - deep learning models (recurrent neural networks, representation learning, etc)
  - generative models

- Generative modeling approaches are frequently based on temporal point process models, e.g. inhomogenous Poisson processes, Cox proportional hazard models, Hawkes process models, etc.

- Intensity function typically corresponds to infection rates per node v:

$$\lambda_v(t|H) = \lim_{\Delta t \to 0} \frac{\mathbb{E}[N_v(t+\Delta t) - N_v(t)|H_t)]}{\Delta t}$$

Zhou, F., Xu, X., Trajcevski, G., & Zhang, K. (2021). A survey of information cascade analysis: Models, predictions, and recent advances. *ACM Computing Surveys (CSUR)*, *54*(2), 1-36.

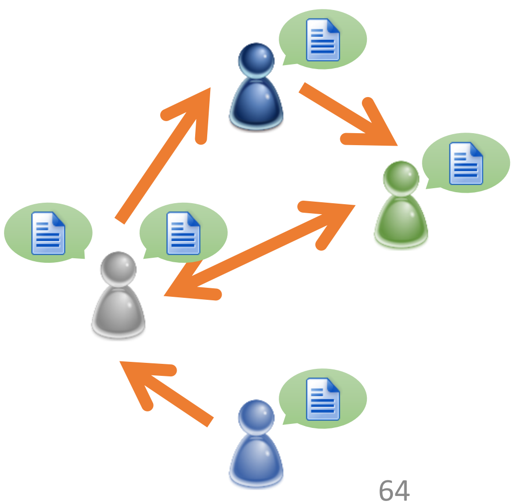
Networks and Text

- Social media data often involve **networks with text** associated
  - Tweets, posts, direct messages/emails,…

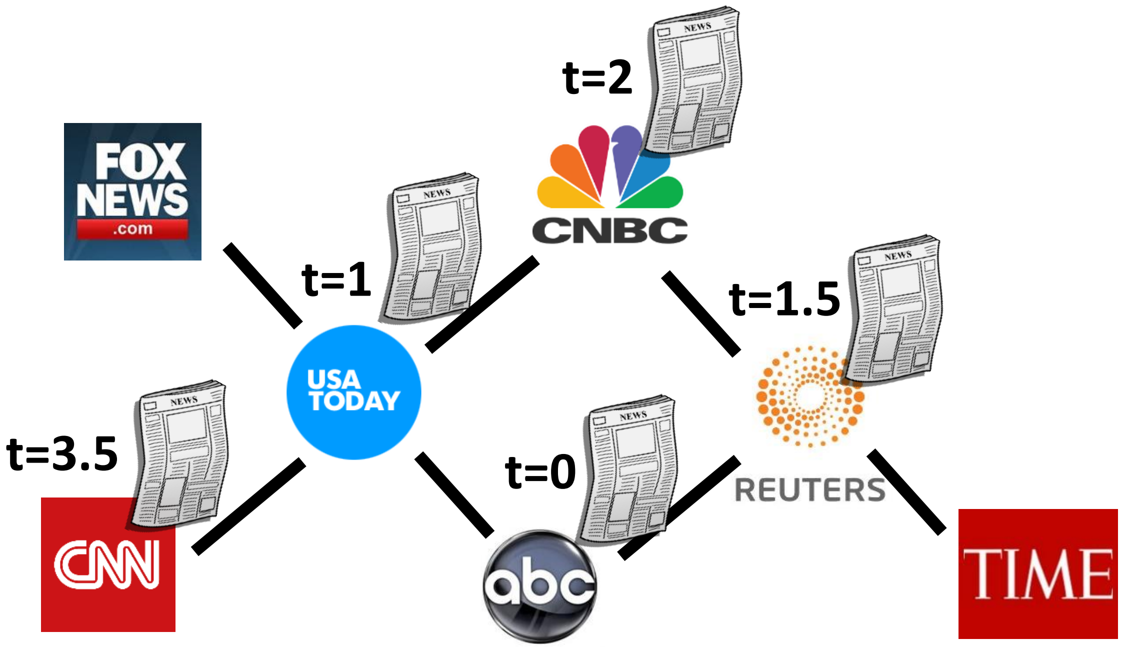
- Leveraging text can help to **improve network modeling**, and to **interpret** the network

- Simple approach: model networks and text **separately**
  - Network model, can determine input for text analysis, e.g. the text for each network community

- More powerful methodology: **joint models of networks and text**
  - Usually combine network and language model components into a single model
  - Generative approach allows straightforward and principled incorporation of multiple data modalities



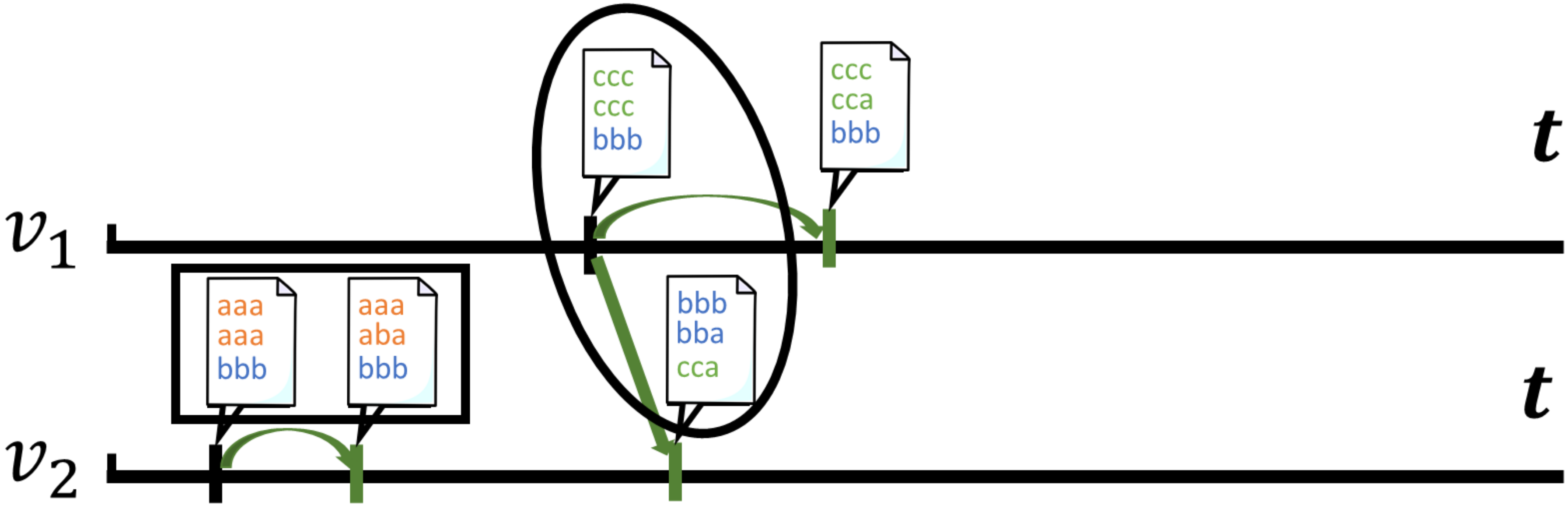
Information diffusion in text-based cascades



- Temporal information
- Content information
- Network is latent

X. He, T. Rekatsinas, J. R. Foulds, L. Getoor, and Y. Liu. HawkesTopic: A joint model for network inference and topic modeling from text-based cascades. ICML 2015.

HawkesTopic model for text-based cascades



Mutual exciting nature: A posting event can trigger future events

Content cascades: The content of a document should be similar to the document that triggers its publication

X. He, T. Rekatsinas, J. R. Foulds, L. Getoor, and Y. Liu. HawkesTopic: A joint model for network inference and topic modeling from text-based cascades. ICML 2015.

Modeling posting times

Mutually exciting nature captured via **Multivariate Hawkes Process** (MHP) [Liniger 09].

For MHP, **intensity process** $\lambda_v(t)$ takes the form:

$$\underbrace{\lambda_v(t)}_{\textbf{Rate}} = \underbrace{\mu_v}_{\textbf{Base intensity}} + \underbrace{\sum_{e:t_e<t} A_{v_e,v} f_\Delta(t - t_e)}_{\textbf{Influence from previous events}}$$

$A_{u,w}$: **influence strength** from u to v

$f_\Delta(\cdot)$: probability density function of the **delay distribution**

Comparison with timestamped relational event models

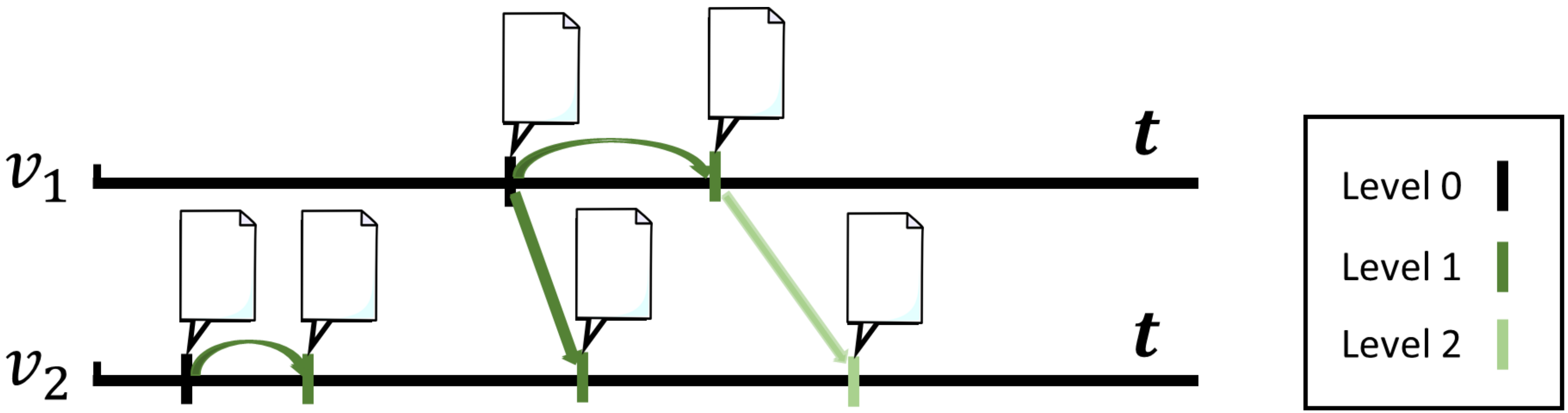
- **Cascade models:** observe events at the n nodes
  - Use n-variate Hawkes process to **infer a latent network** through $n \times n$ excitation matrix A

$$\lambda_v(t) = \mu_v + \sum_{e:t_e<t} A_{v_e,v} f_\Delta(t - t_e)$$

- **Timestamped relational event models: directly observe network** through events between all $O(n^2)$ node pairs
  - Use univariate or bivariate Hawkes process to **model events between each node pair** (u, v)

$$\lambda_{uv}(t) = \mu_{uv} + \sum_{e:t_e^{uv}<t} \alpha_{uv}\, f_\Delta(t - t_e^{uv}) + \sum_{e:t_e^{vu}<t} \alpha_{vu}\, f_\Delta(t - t_e^{vu})$$

Clustered Poisson process interpretation

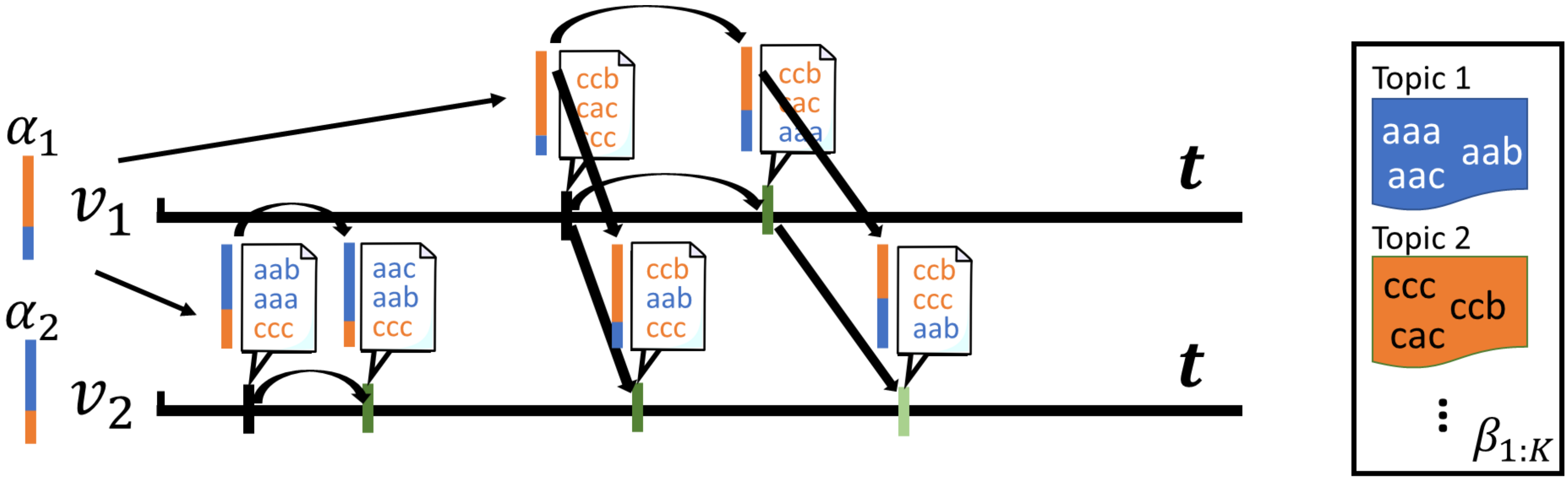


Generate events and their posting times in a **breadth first** order by interpreting the MHP as **clustered Poisson process** [Simma 10]

Provide explicit **parent relationship** for evolution of the content information

X. He, T. Rekatsinas, J. R. Foulds, L. Getoor, and Y. Liu. HawkesTopic: A joint model for network inference and topic modeling from text-based cascades. ICML 2015.

Generating documents



Step 1: Generate the topics $\beta_{1:K}$: $\beta_k \sim Dir(\alpha)$

Step 2: For spontaneous events (level=0): $\eta_e \sim N(\alpha_v, \sigma^2 I)$

Step 3: For triggered events (level>0): $\eta_e \sim N(\eta_{\text{parent}[e]}, \sigma^2 I)$

Step 4: For each word in each document: $z_{e,n} \sim \text{Discrete}\big(\pi(\eta_e)\big), x_{e,n} \sim \text{Discrete}(\beta_{z_{e,n}})$

X. He, T. Rekatsinas, J. R. Foulds, L. Getoor, and Y. Liu. HawkesTopic: A joint model for network inference and topic modeling from text-based cascades. ICML 2015.

Experiments for HawkesTopic

EventRegistry

"Ebola" news articles ~4 months

~9k articles, 330 news media sites

Copying information as ground truth

Cornell University Library

arXiv.org

High-energy physics theory papers ~12 years

Top 50/100/200 researchers

Citation network as ground truth

Evaluation metrics:

-- Topic modeling: document competition likelihood [Wallach et al. 09]

-- Network Inference: AUC against the ground truth network

X. He, T. Rekatsinas, J. R. Foulds, L. Getoor, and Y. Liu. HawkesTopic: A joint model for network inference and topic modeling from text-based cascades. ICML 2015.

Results: ArXiv

Network Inference accuracy: 40% improvement

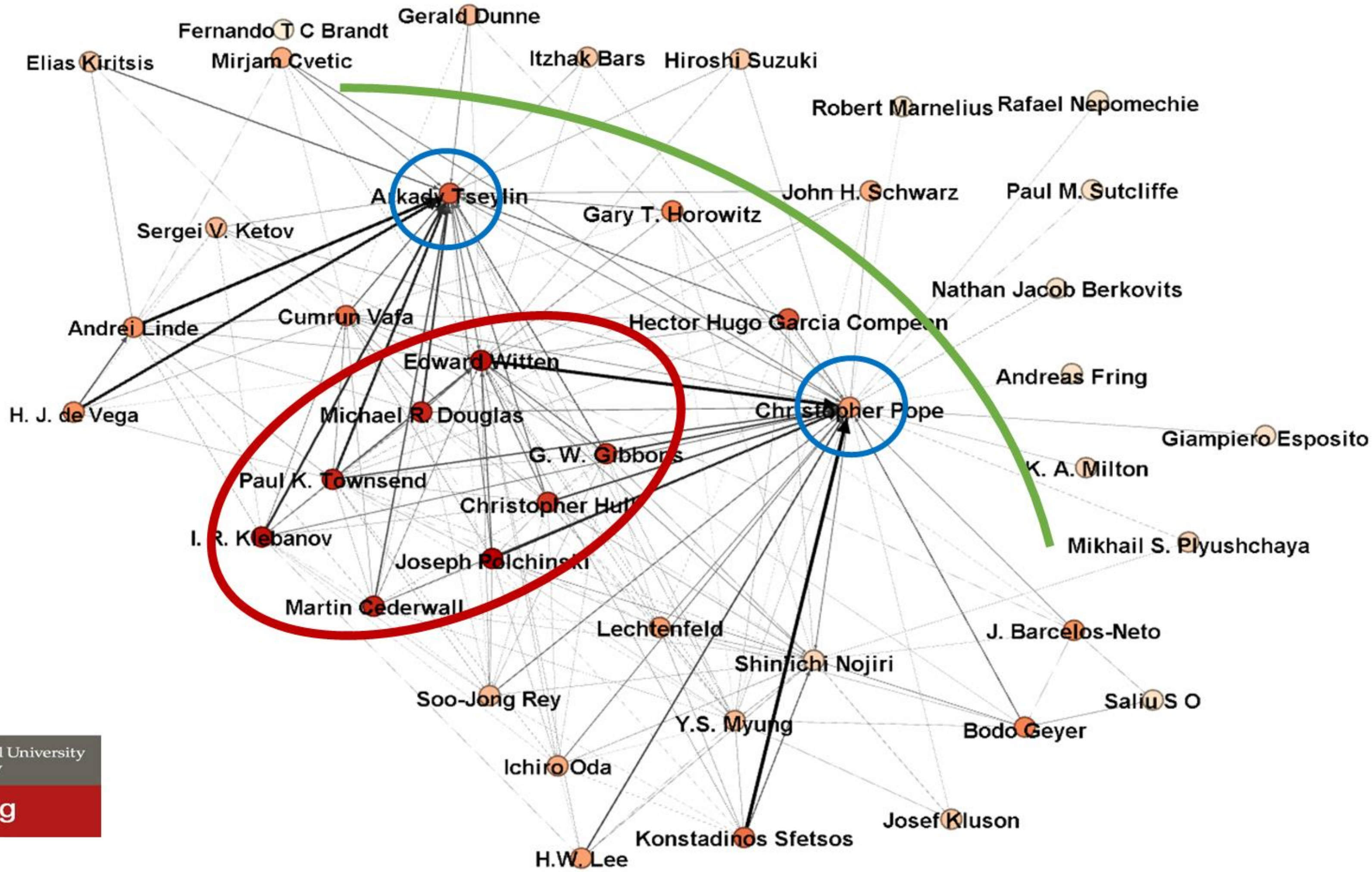
	Hawkes	Hawkes-LDA	Hawkes-CTM	HTM
Top50	0.594	0.656	0.645	**0.807**
Top100	0.588	0.589	0.614	**0.687**
Top200	0.618	0.630	0.629	**0.659**

Topic modeling accuracy:

	LDA	CTM	HTM
Top50	-11074	-10769	**-10708**
Top100	-15711	-15477	**-15252**
Top200	-27758	-27630	**-27443**

X. He, T. Rekatsinas, J. R. Foulds, L. Getoor, and Y. Liu. HawkesTopic: A joint model for network inference and topic modeling from text-based cascades. ICML 2015.

Results: ArXiv



X. He, T. Rekatsinas, J. R. Foulds, L. Getoor, and Y. Liu. HawkesTopic: A joint model for network inference and topic modeling from text-based cascades. ICML 2015.

Summary

- **Generative models** provide a powerful mechanism for modeling and analyzing network data
- **Latent variable models** offer a flexible yet interpretable modeling approach motivated by sociological principles
  - Latent space model
  - Stochastic block model
- Generative models provide a rich mechanism for incorporating **network dynamics**
  - Key building block: **temporal point processes**
  - **Directly-observed networks**: combine **Hawkes process** with static network model to generate timestamped **relational events**
  - **Indirectly-observed networks:** use **multivariate Hawkes process** to **estimate a latent network** from timestamped observations at nodes