EuroVis 2017
Machine Learning Methods in Visualization for Big Data 2017
12 June 2017, Barcelona, Spain

# Methods for text data

Jaakko Peltonen[3]

3 University of Tampere, Aalto University

# Topic models

- Topic models are a prominent way to model the content of documents
- Each document is represesented as a **bag of words**: counts of how many times each different word has appeared
- Topic models represent documents as a mixture of underlying **latent topics**, where each topic has a **probability table** to generate different words over the **vocabulary**.
- Topic models are a kind of **dimensionality reduction** for count data: instead of representing a document by the vector of all counts, the document can be represented as a vector of inferred **topic activities** over a small number of topics.

# Latent Dirichlet Allocation

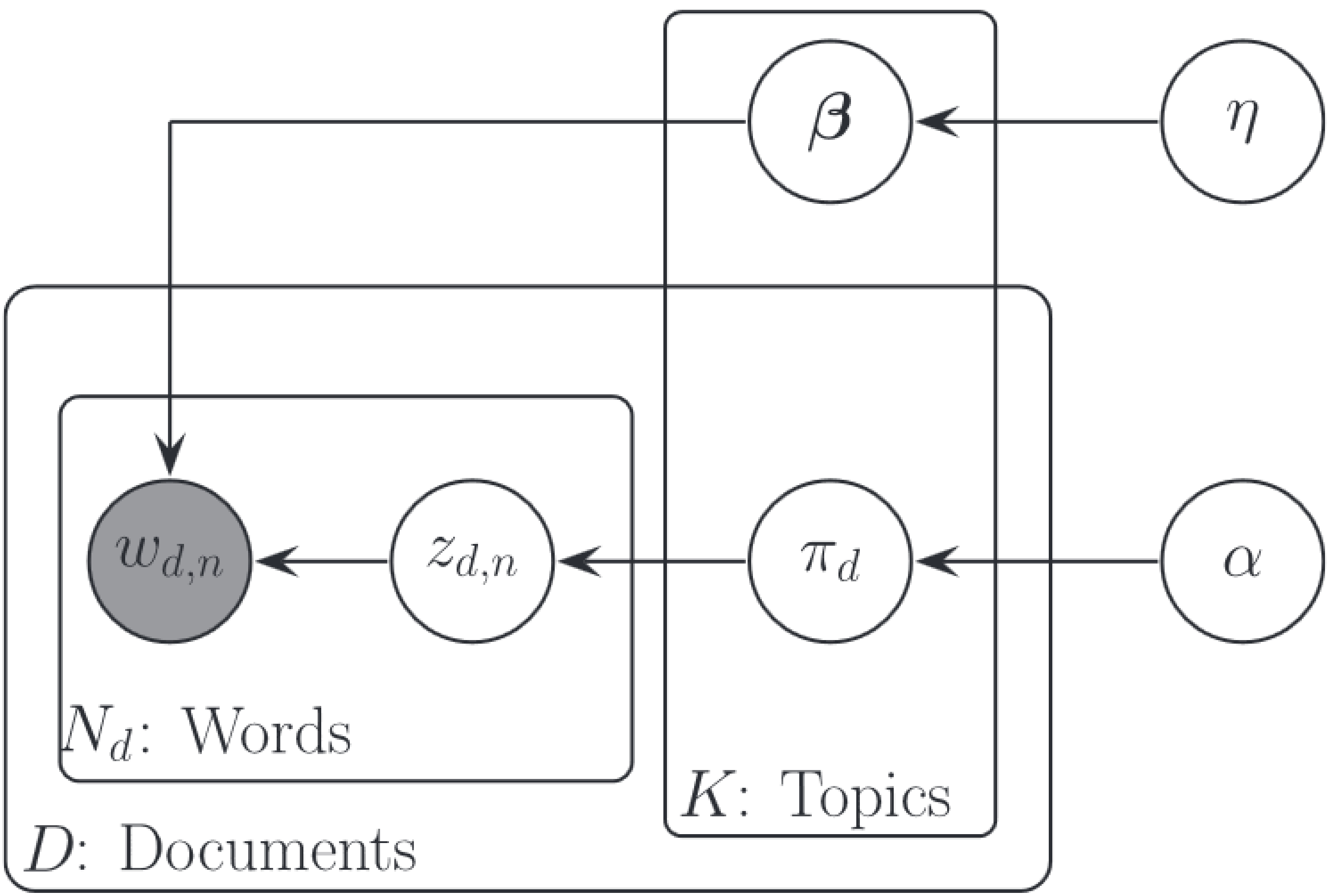
- Latent Dirichlet Allocation (LDA – not to be confused with Linear Discriminant Analysis!) is a simple topic model
- Also called ”discrete PCA”.

- To generate a document d, a **topic distribution** $\pi_d$ is drawn from a prior so that $\boldsymbol{\pi}_d \sim Dirichlet(\boldsymbol{\alpha})$, and then the words are generated one by one.
- To generate the n:th word in the document:
  - a topic index $z_{n,d}$ is drawn from the topic distribution: $z_{n,d} \sim Multinomial(\boldsymbol{\pi}_d)$
  - the word is then drawn from a topic-wise word distribution: $w_{n,d} \sim Multinomial(\boldsymbol{\beta}_{z_{n,d}})$

- $\boldsymbol{\beta}_k = \{\beta_{w|k}\}_w$ are probabilities of each word w in the kth topic. The available topics are the same for all documents.
- Typically the topic-wise word distributions are drawn from a prior $\boldsymbol{\beta}_k \sim Dirichlet(\eta)$, where $\eta$ is the topic hyperparameter.

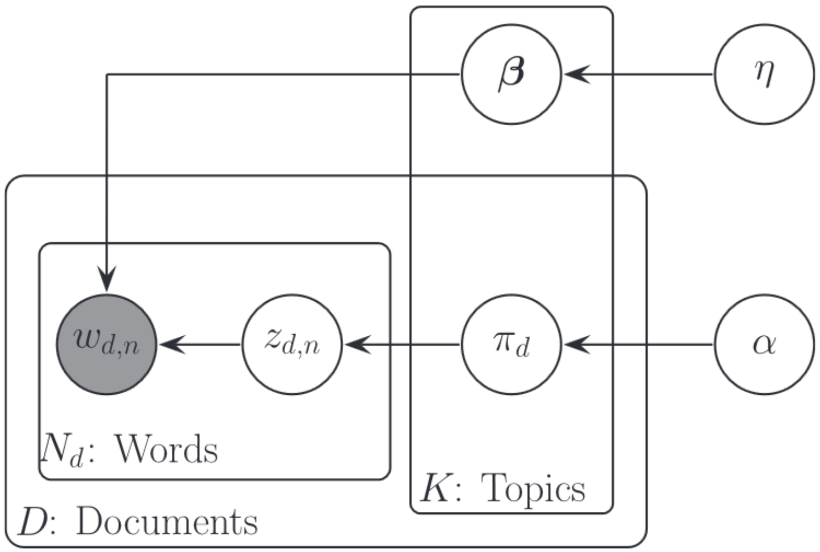
# Latent Dirichlet Allocation

- Plate diagram model of the LDA generative process.




- In LDA each word is generated independently given the topic. The order of the word occurrences does not matter.
- LDA is suitable for count data such as bag-of-words representations of text, where only the overall count of each different word is observed

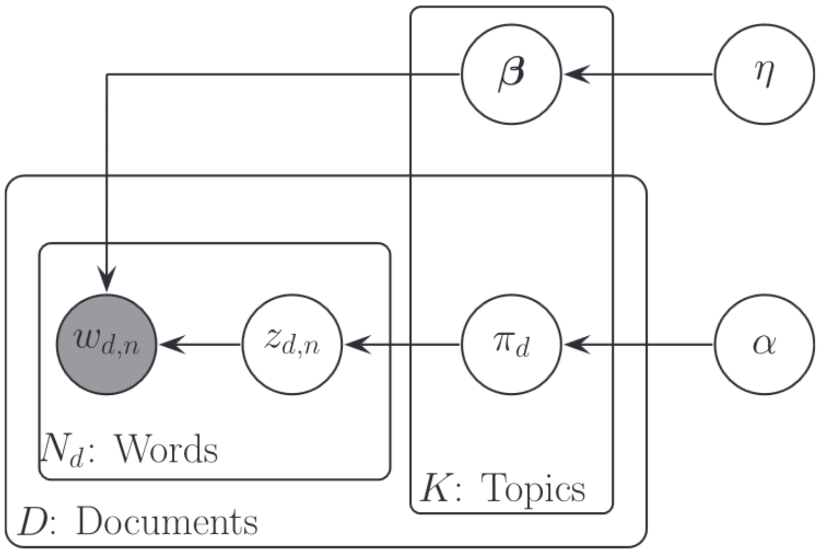
# Latent Dirichlet Allocation




- Given a data set of documents, the LDA model can be fitted to the data by maximum a posteriori methods.

- When LDA is learned from a data set, the Dirichlet priors for the word distribution **mitigate overfitting** with large vocabularies: words that do not appear in the training set still get some probability to appear in future documents. (cf. PLSA: no priors)
- Why use Dirichlet priors? **Convenient properties**: finite dimensional sufficient statistics & conjugate to the multinomial distribution. Allows some parameters to be integrated out **analytically** when fitting an LDA model

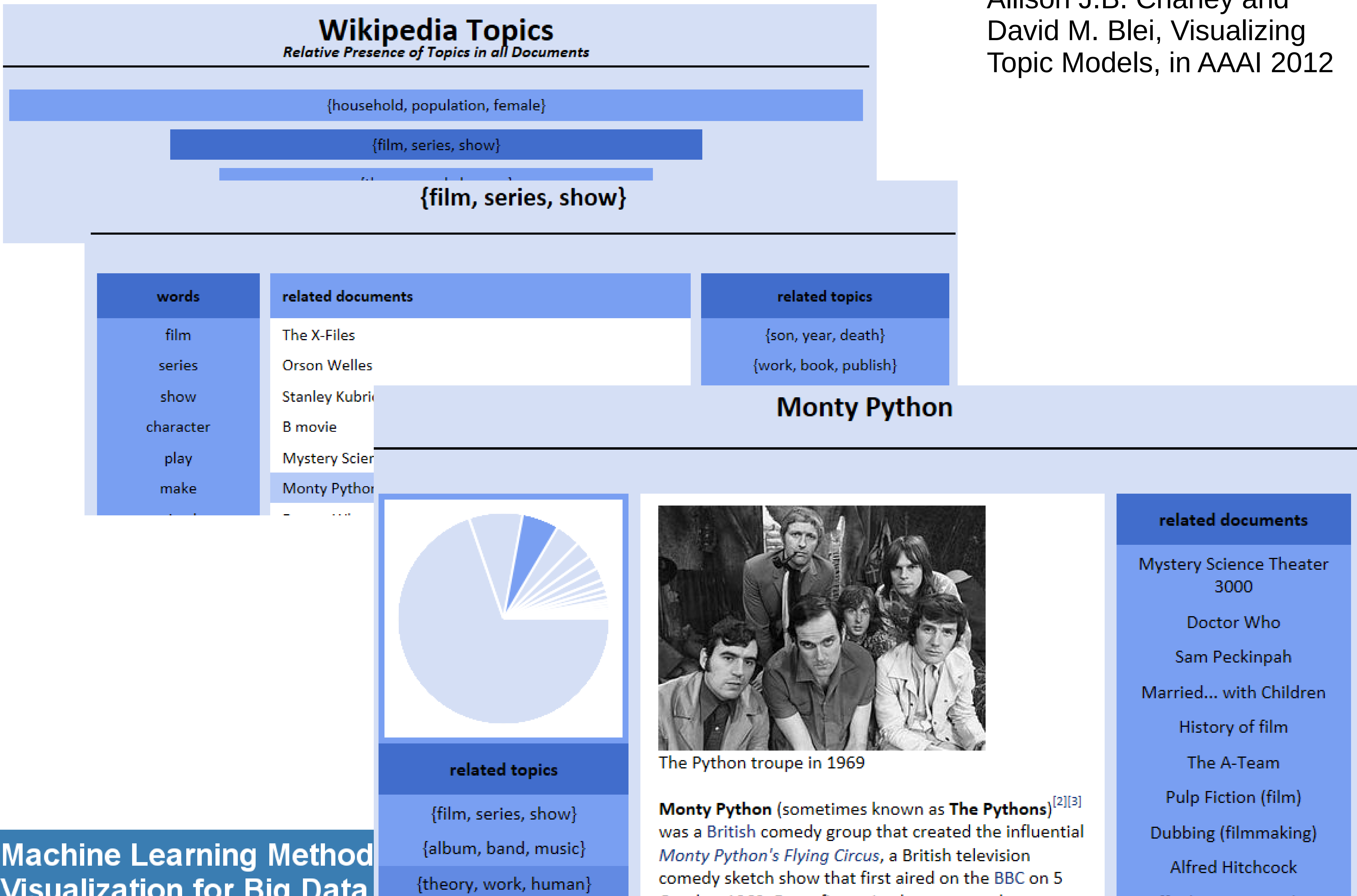
# Latent Dirichlet Allocation




- Given a fitted LDA model, data can be visualized in two ways.
- Each document has a vector $\pi_d$ (probability distribution) whose elements are proportions over the available topics.
- Each topic is an **axis** of a low-dim. topic space,(simplex), and the $\pi_d$ positions the document in this space.
- Each topic k has a word probability table $\beta_k = \{\beta_{w|k}\}_w$ whose top words can be listed.
- Sometimes researchers try to guess semantical meanings to the topics, and give them semantically meaningful names. These meanings and names **do not arise** from the topic model.

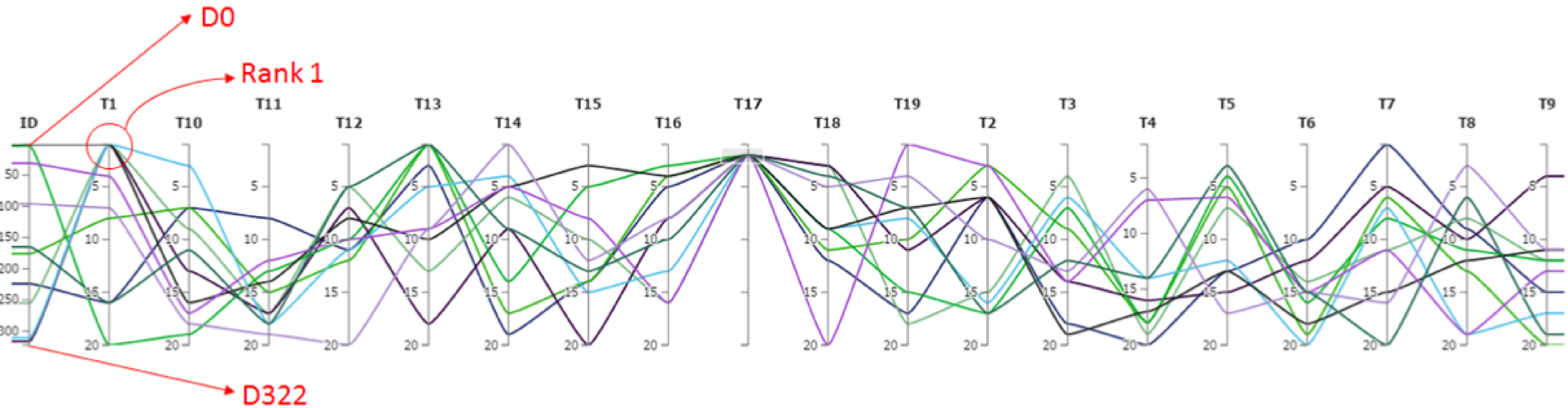
# Latent Dirichlet Allocation

Topic based Wikipedia browsing system from Allison J.B. Chaney and David M. Blei, Visualizing Topic Models, in AAAI 2012



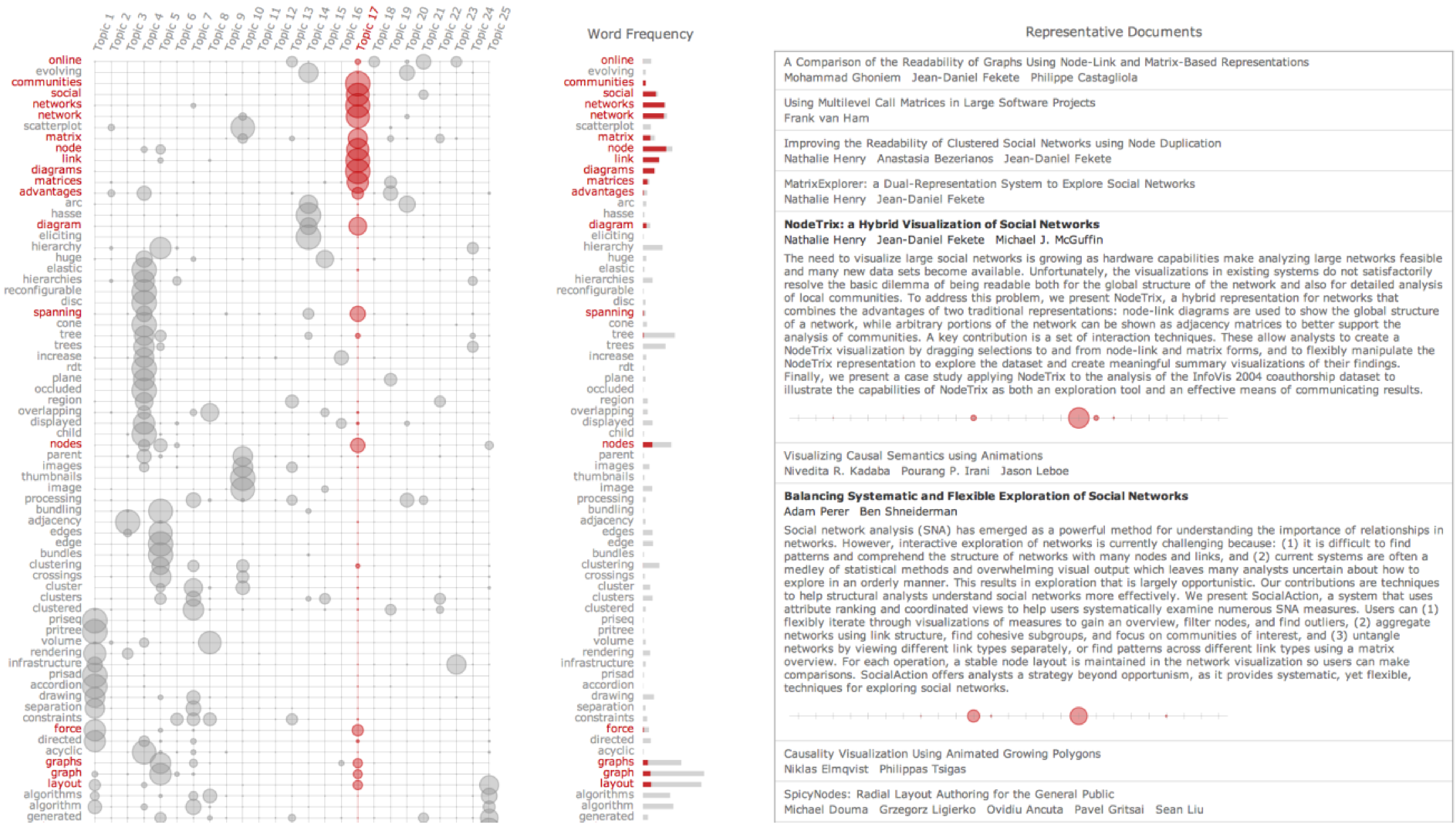
# Latent Dirichlet Allocation

Parallel coordinate plot of several documents versus 20 topics, from Ashwinkumar Ganesan, Kiante Brantley, Shimei Pan and Jian Chen, LDAExplore: Visualizing Topic Models Generated Using Latent Dirichlet Allocation, IUI workshop on text analytics 2015



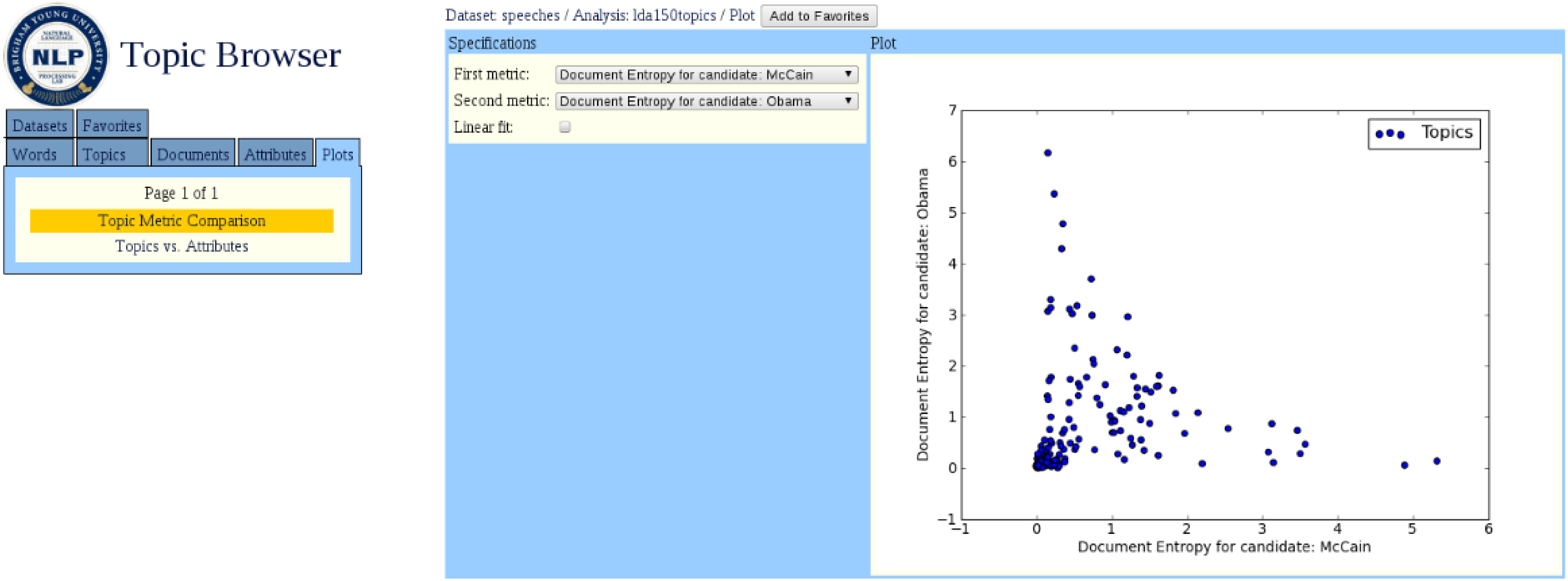
# Latent Dirichlet Allocation

Matrix representation of several documents versus their words, using seriation.
From Jason Chuang, Christopher D. Manning, Jeffrey Heer, Termite: Visualization Techniques for Assessing Textual Topic Models, in proc. AVI ‘12, 2012.



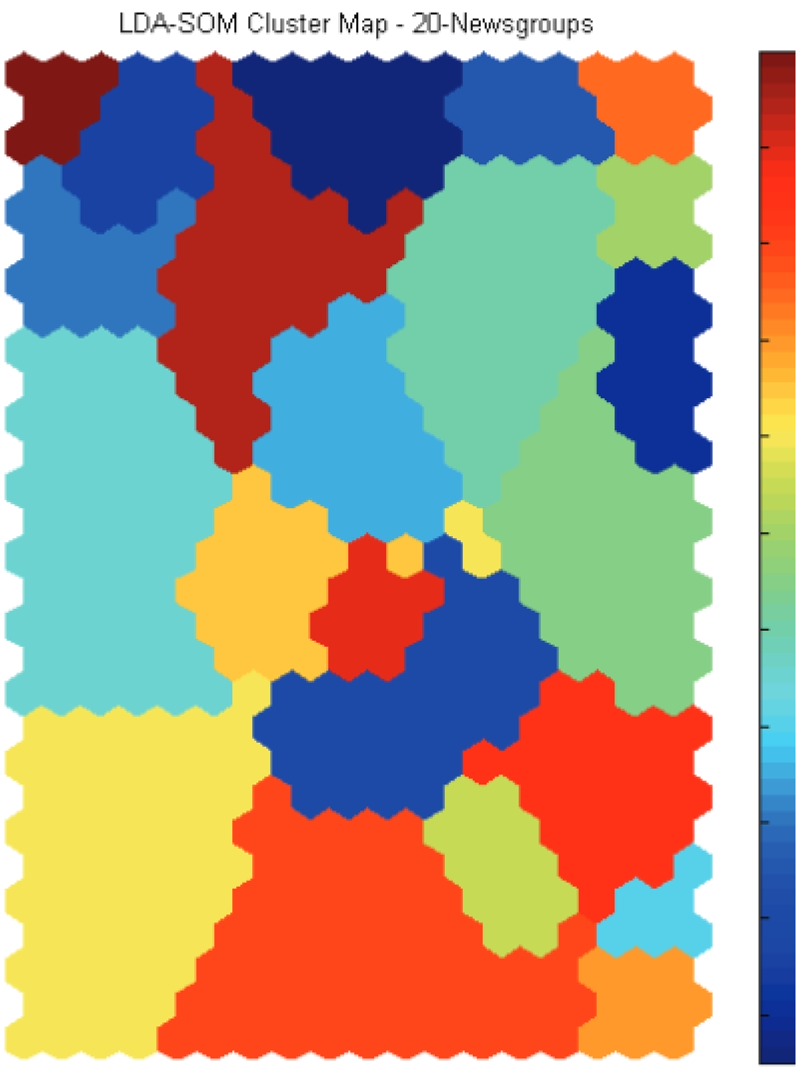
# Latent Dirichlet Allocation

Comparison of topics' activity (summarized by entropy over documents = speeches) between two different document subgroups (here subgroups = authors who are political candidates). From Matthew J. Gardner, Joshua Lutes, Jeff Lund, Josh Hansen, Dan Walker, Eric Ringger, and Kevin Seppi, The Topic Browser: An Interactive Tool for Browsing Topic Models, in proc. NIPS 2010 workshop on Challenges of Data Visualization.

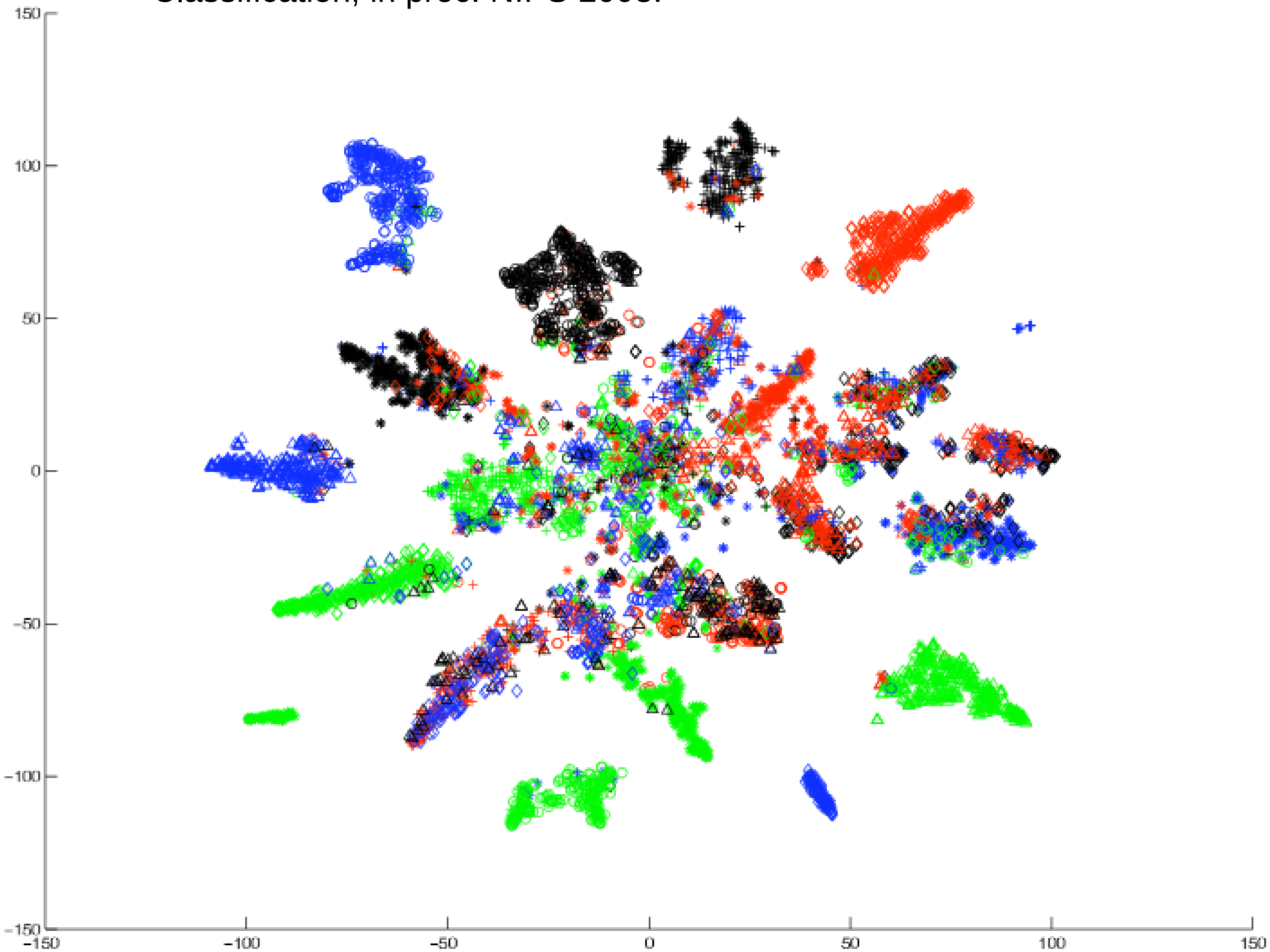


# Latent Dirichlet Allocation

Self-organizing map trained on topic model outputs (topic distributions of each document). From Jeremy R. Millar and Gilbert L. Peterson and Michael J. Mendenhall, Document Clustering and Visualization with Latent Dirichlet Allocation and Self-Organizing Maps, in proc. FLAIRS 2009.

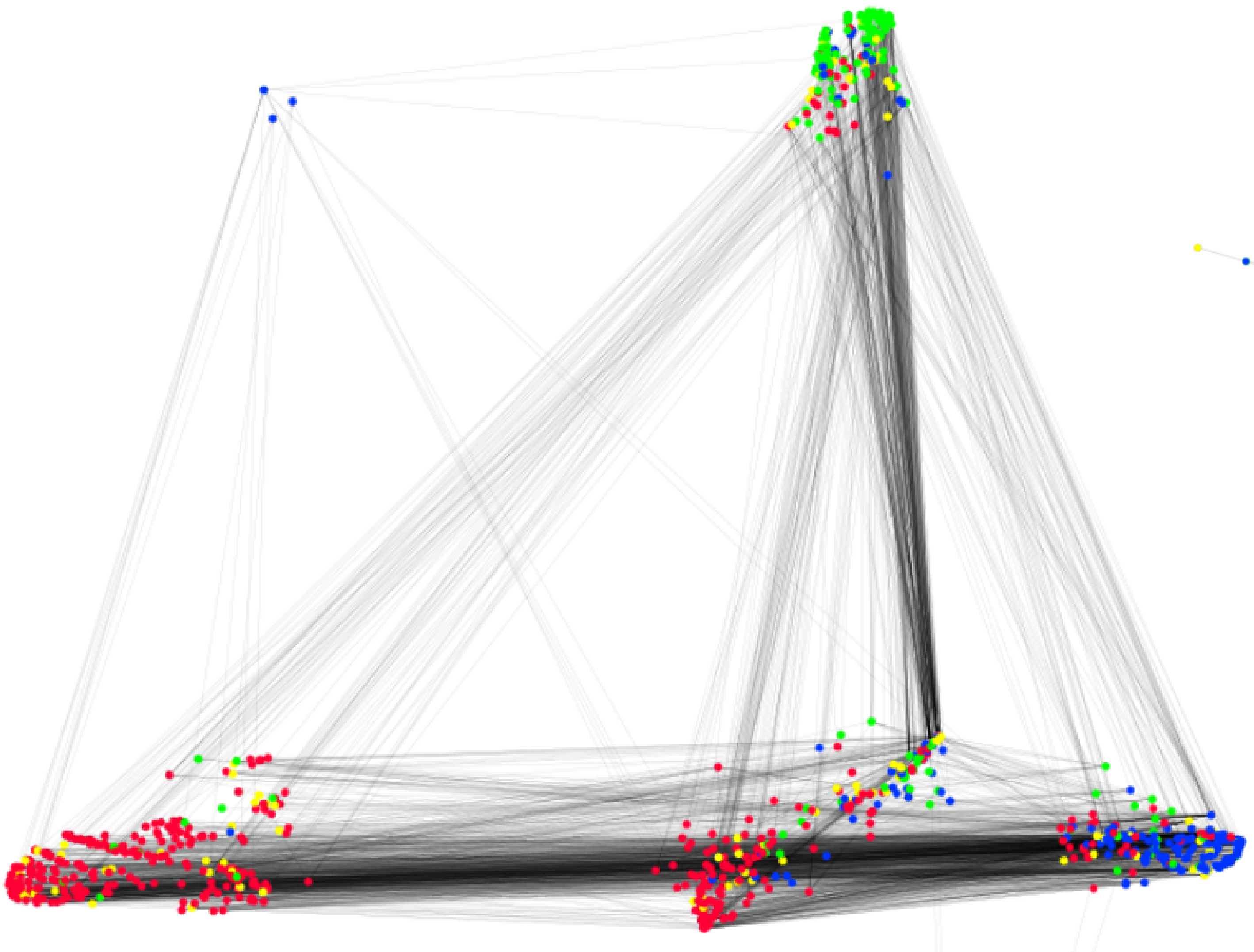


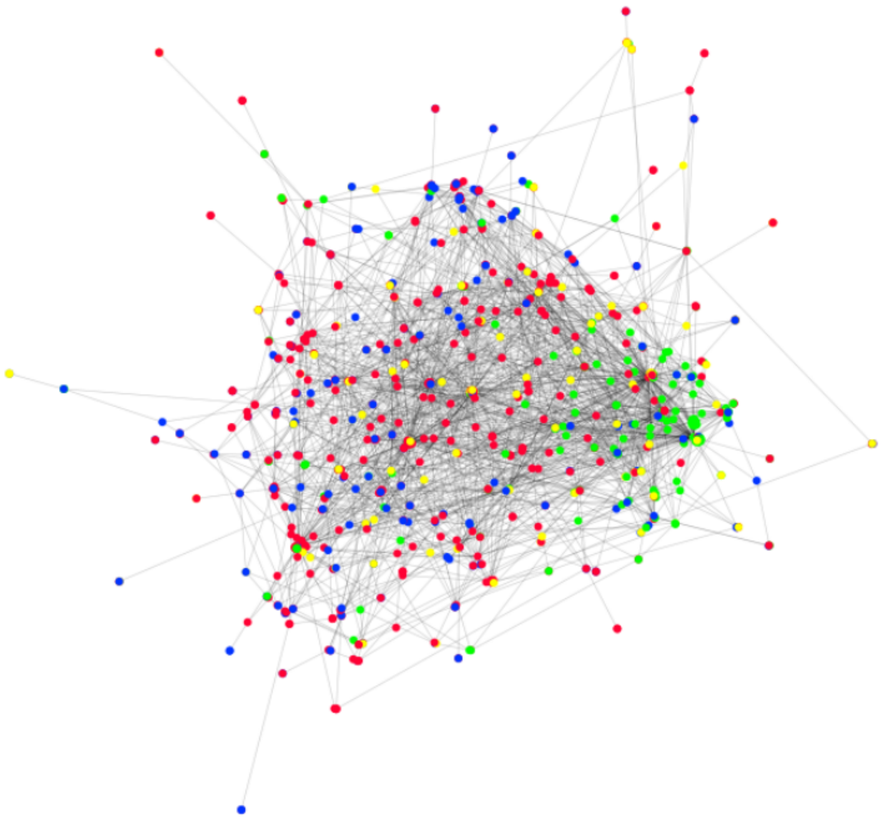

T-distributed Stochastic Neighbor Embedding trained for documents represented by topic model outputs (topic distributions of each document).
From Simon Lacoste-Julien, Fei Sha, and Michael I. Jordan, DiscLDA: Discriminative Learning for Dimensionality Reduction and Classification, in proc. NIPS 2008.



# Latent Dirichlet Allocation

Graph layout created by Neighbor Retrieval Visualizer applied to topic model outputs (topic distributions of each graph node), when inputs to the topic model are nodes represented as a ”bag of links”. In this case the graph arises from text data: nodes are different words, and links are adjacencies in the novels of Jane Austen. From Juuso Parkkinen, Kristian Nybo, Jaakko Peltonen, and Samuel Kaski. Graph Visualization With Latent Variable Models. In Proceedings of MLG 2010, the Eighth Workshop on Mining and Learning with Graphs, 2010.

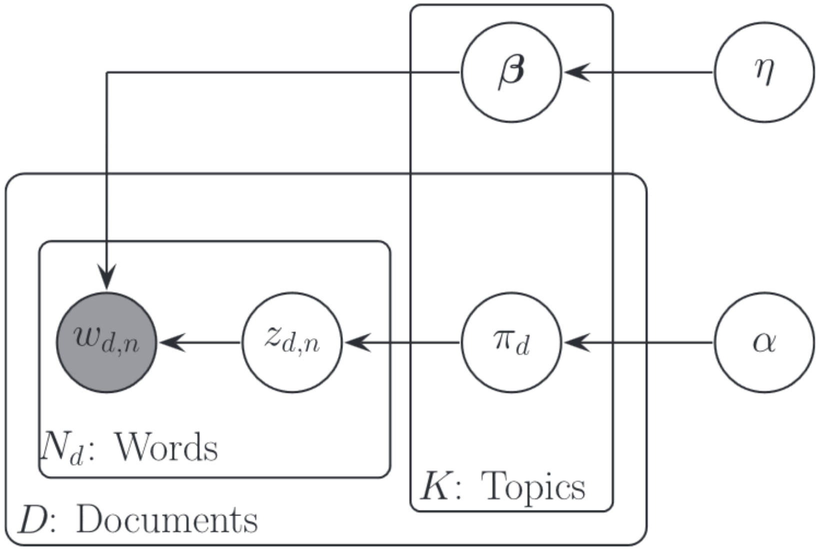




Traditional LinLog layout algorithm creates a hairball for the same graph.

Colors = grammatical classes. Blue nodes are adjectives, red nodes are nouns, green nodes are verbs, yellow nodes have multiple word classes.

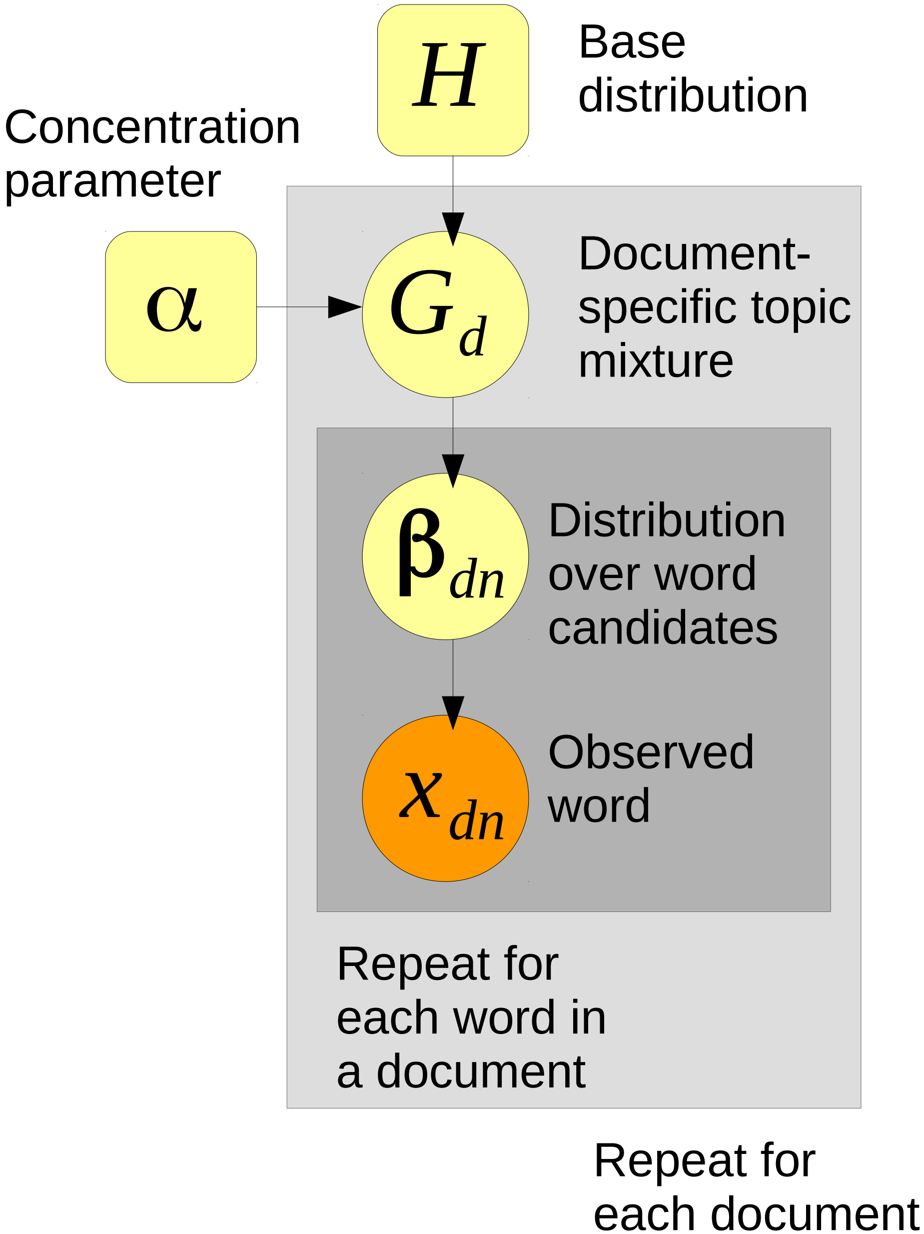
# Latent Dirichlet Allocation




- The LDA model assumes the number K of available topics to be specified in advance.
- Problematic when the number of actual underlying topics can be large, and expert knowledge for choosing the correct number of topics may not be available.

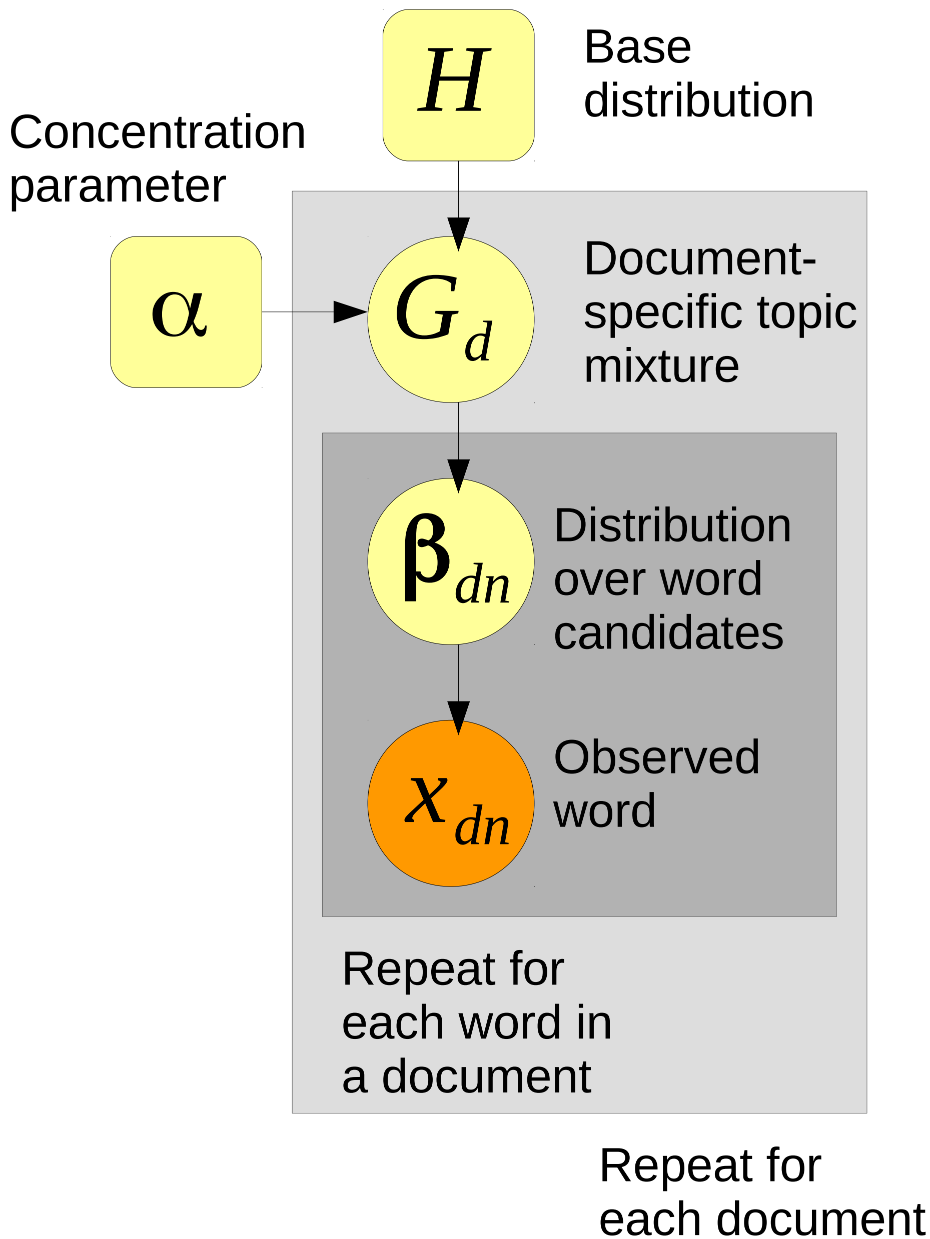
- If the number of topics is set too small, it forces the model to merge some of the real topics
- If the number of topics is set too large, maximum likelihood fitting will overfit and split some real topics according to artifacts in the observed data.

# Dirichlet Process Mixture Topic Model




- ”Nonparametric” topic model where the number of topics does not need to be chosen beforehand
- A document is assumed to come from a mixture of topics.
- The mixture is drawn from a **prior over possible mixtures:** all possible numbers of topics up to infinity, and all possible proportions over topics
- The **Dirichlet process** is a suitable prior
- The model is described in three ways:
  1. A ”theoretic way” to generate data
  2. An iterative process to generate data
  3. Inference equations

# Dirichlet Process Mixture Topic Model




- Theoretic way:
- generate a mixture from the Dirichlet process prior, $G_d \sim DP(H, alpha)$

- For each word n, draw a mixture component (=word distribution = word probability table) from the mixture

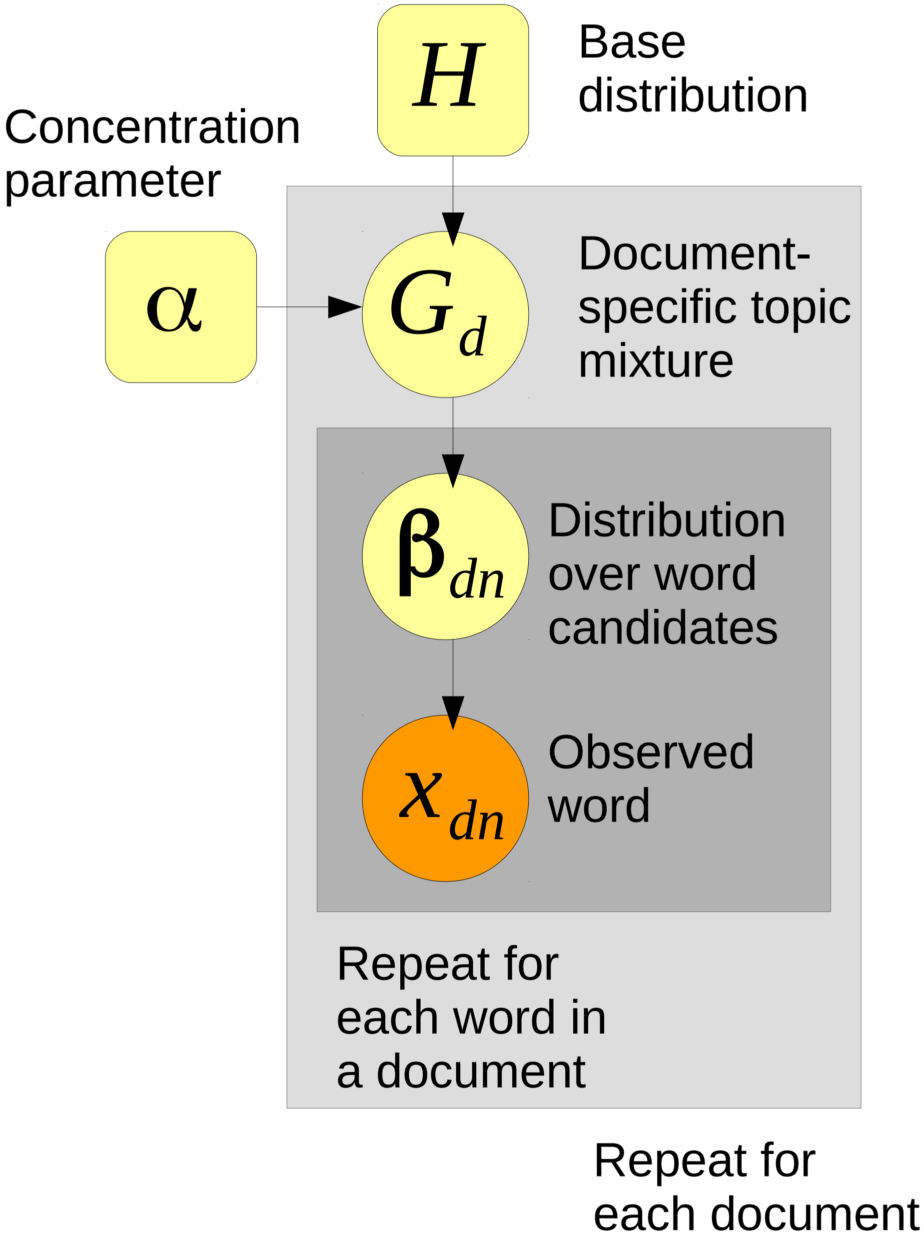
  $\beta_{dn} \sim G_d$

  and draw the observed word from the word distribution

  $x_{dn} \sim Multinomial(\beta_{dn})$

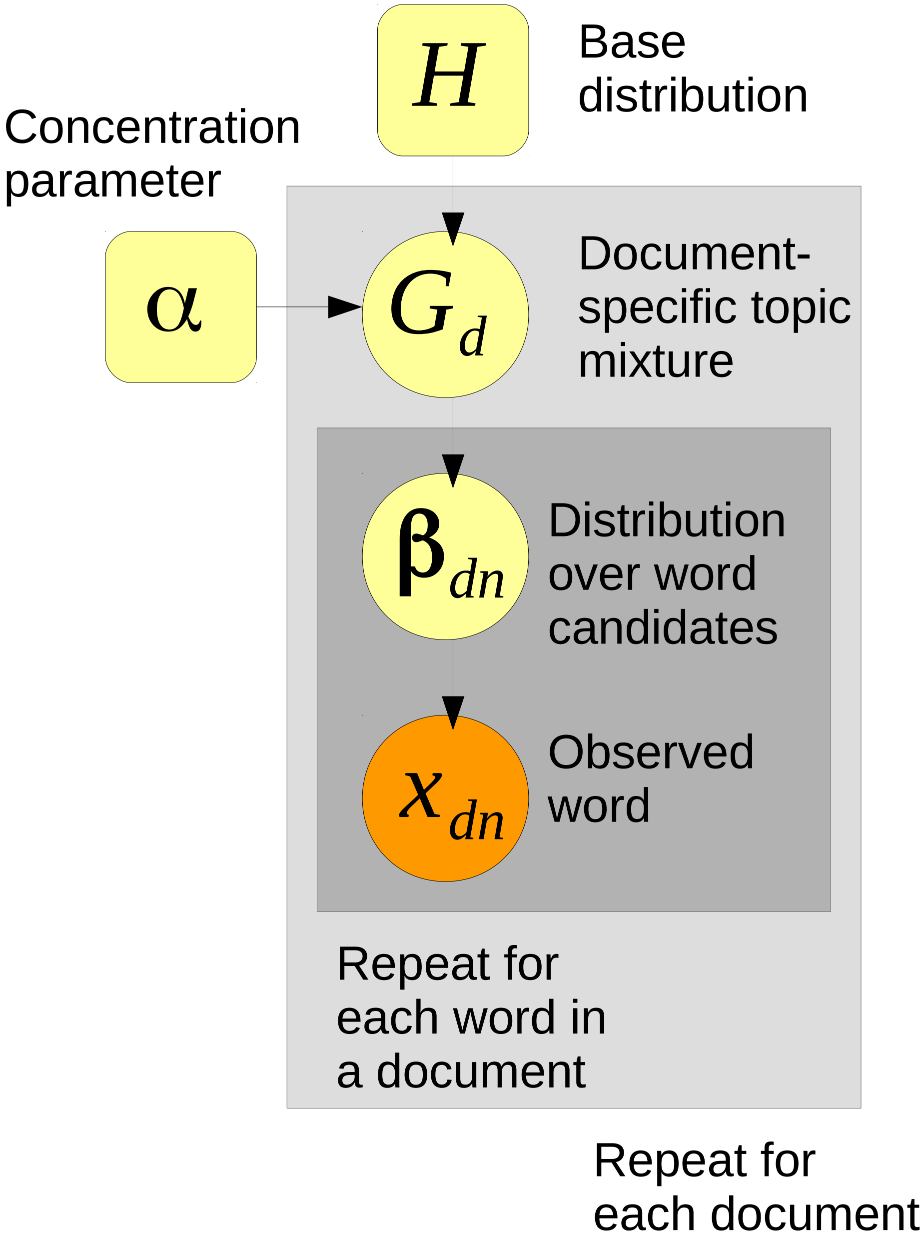
- Problem: it is hard to draw an entire mixture (potentially an infinite-size object) over all possibilities

# Dirichlet Process Mixture Topic Model



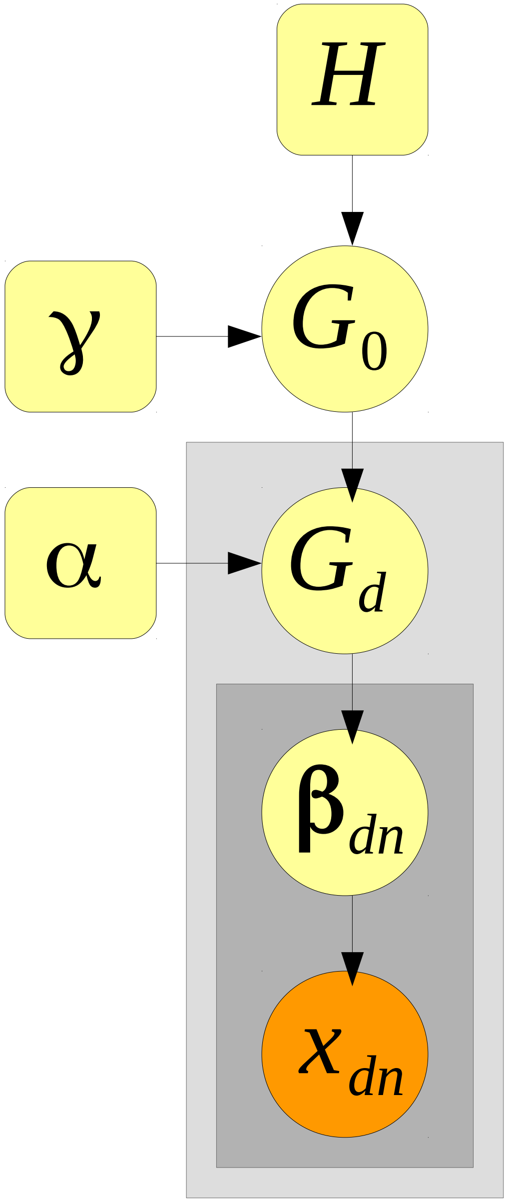

- Iterative way: **chinese restaurant process**

# Dirichlet Process Mixture Topic Model




- Iterative way: **chinese restaurant process**
- Observed words $x_{dn}$ are **food servings** experienced by **customers**
- Each customer comes in and chooses a **table** where to sit.
- Customers prefer to sit at **popular tables** where others already sit, but sometimes choose to take a **new table**
- Topics are **dishes**. Each table serves a particular dish: all customers sitting at the table get servings from that dish.
- Because customers prefer to sit at popular tables, the dish $\beta_{dn}$ of a new customer is likely to be the same as a dish already being served

# Hierarchical Dirichlet Processes




- Hierarchical extension of the Dirichlet Process Mixture Model
- Instead of drawing all documents in the collection from an uninformative generic base distribution:
- Sample a collection-specific base distribution
- Allows the model to use the same topics over multiple documents, helps against overfitting
- Inference is similar as in the DPMM, but slightly more complicated.
- Visualization similar as in DPMM