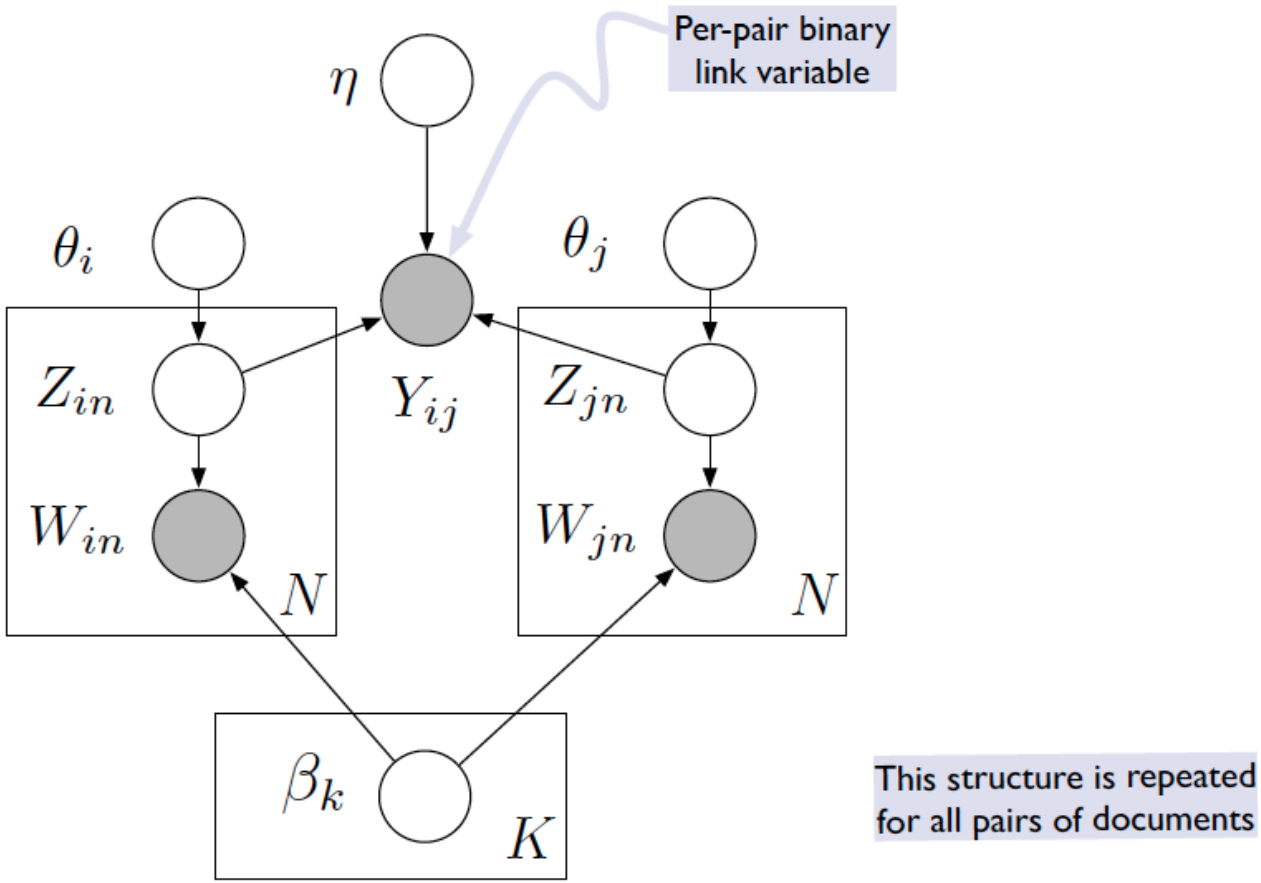
Relational Topic Models

Pawan Goyal

CSE, IITKGP

October 20, 2014

Relational Topic Models



Works in a supervised framework, allowing predictions about new and unlinked data

Supervised settings of LDA

Use data points paired with response variables

- User reviews paired with a number of stars
- Web pages paired with a number of likes
- Documents paired with links to other documents
- Images paired with a category

Supervised settings of LDA

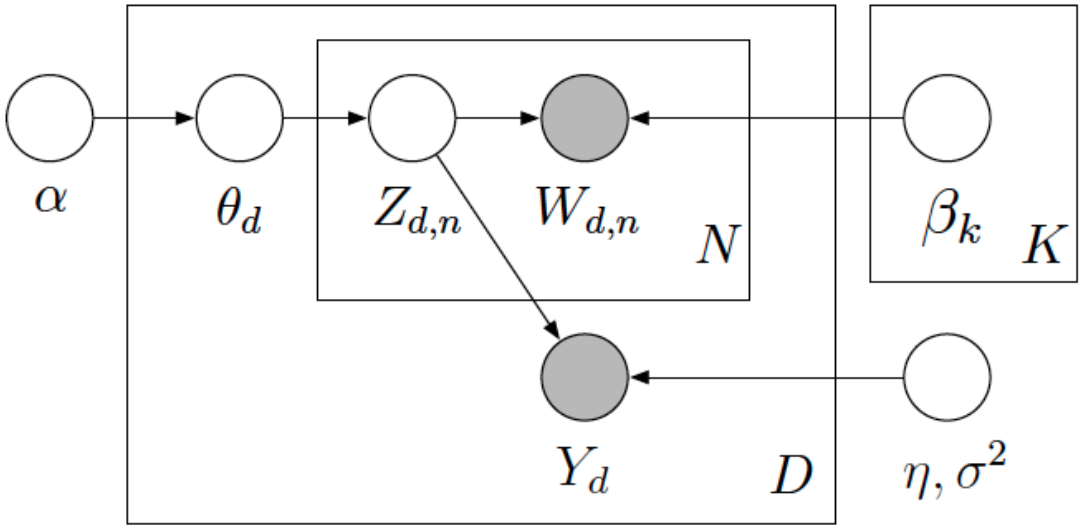
Use data points paired with response variables

- User reviews paired with a number of stars
- Web pages paired with a number of likes
- Documents paired with links to other documents
- Images paired with a category

Supervised topic modes

are topic models of documents and responses, fit to find topics predictive of the response

Supervised LDA



1. Draw topic proportions $\theta \mid \alpha \sim \text{Dir}(\alpha)$.
2. For each word
   - Draw topic assignment $z_n \mid \theta \sim \text{Mult}(\theta)$.
   - Draw word $w_n \mid z_n, \beta_{1:K} \sim \text{Mult}(\beta_{z_n})$.
3. Draw response variable $y \mid z_{1:N}, \eta, \sigma^2 \sim \text{N}(\eta^\top \bar{z}, \sigma^2)$, where

$$\bar{z} = (1/N) \sum_{n=1}^{N} z_n.$$

Supervised LDA: why a different model is required?

Think of an alternative approach using original settings of LDA

Supervised LDA: why a different model is required?

Think of an alternative approach using original settings of LDA

Formulate a model in which the response variable y is regressed on topic proportions θ

Supervised LDA: why a different model is required?

Think of an alternative approach using original settings of LDA

Formulate a model in which the response variable y is regressed on topic proportions θ

Why then a different model?

Supervised LDA: why a different model is required?

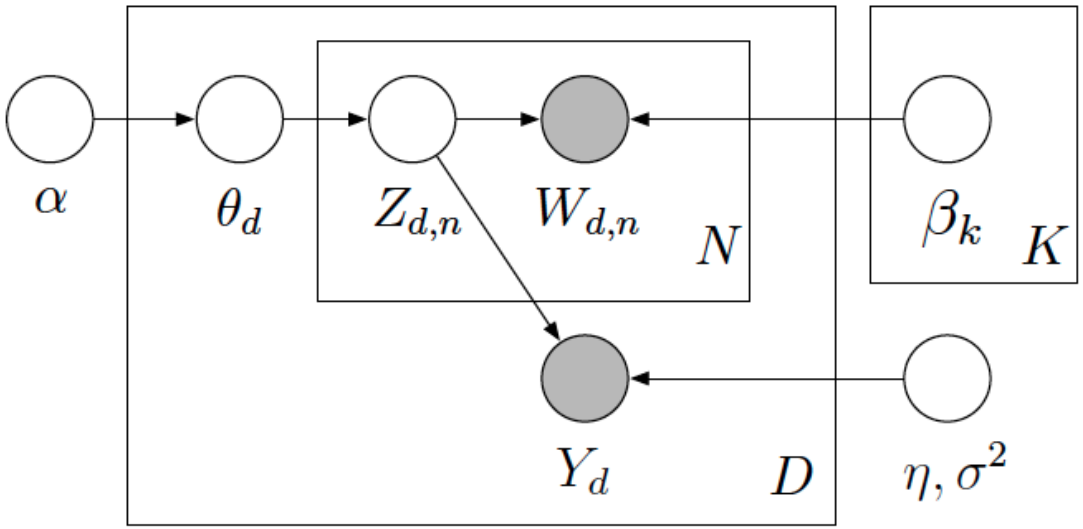
Think of an alternative approach using original settings of LDA

Formulate a model in which the response variable y is regressed on topic proportions θ

Why then a different model?

- The response depends on the topic frequencies, as actually ocurred in the document, rather than on the mean of the distribution generating the topics
- The response variable can be treated as an important observation to infer the topic probabilities in a supervised manner

Supervised LDA



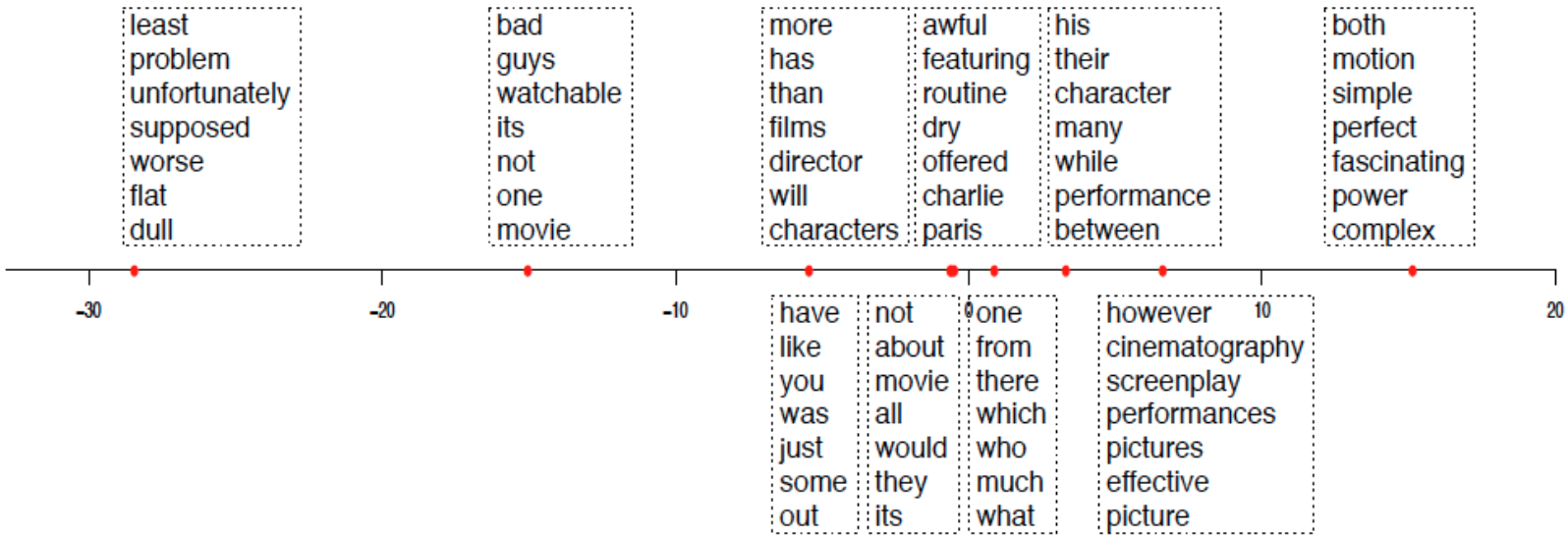
- The response variable y is drawn *after* the document because it depends on $z_{1:N}$, an assumption of **partial exchangeability**.
- Consequently, y is necessarily conditioned on the words.

Prediction

- Fit sLDA parameters to documents and responses. This gives:
  - topics $\beta_{1:K}$
  - coefficients $\eta_{1:K}$
- We have a new document $w_{1:N}$ with unknown response value.
- We predict y using the SLDA expected value:

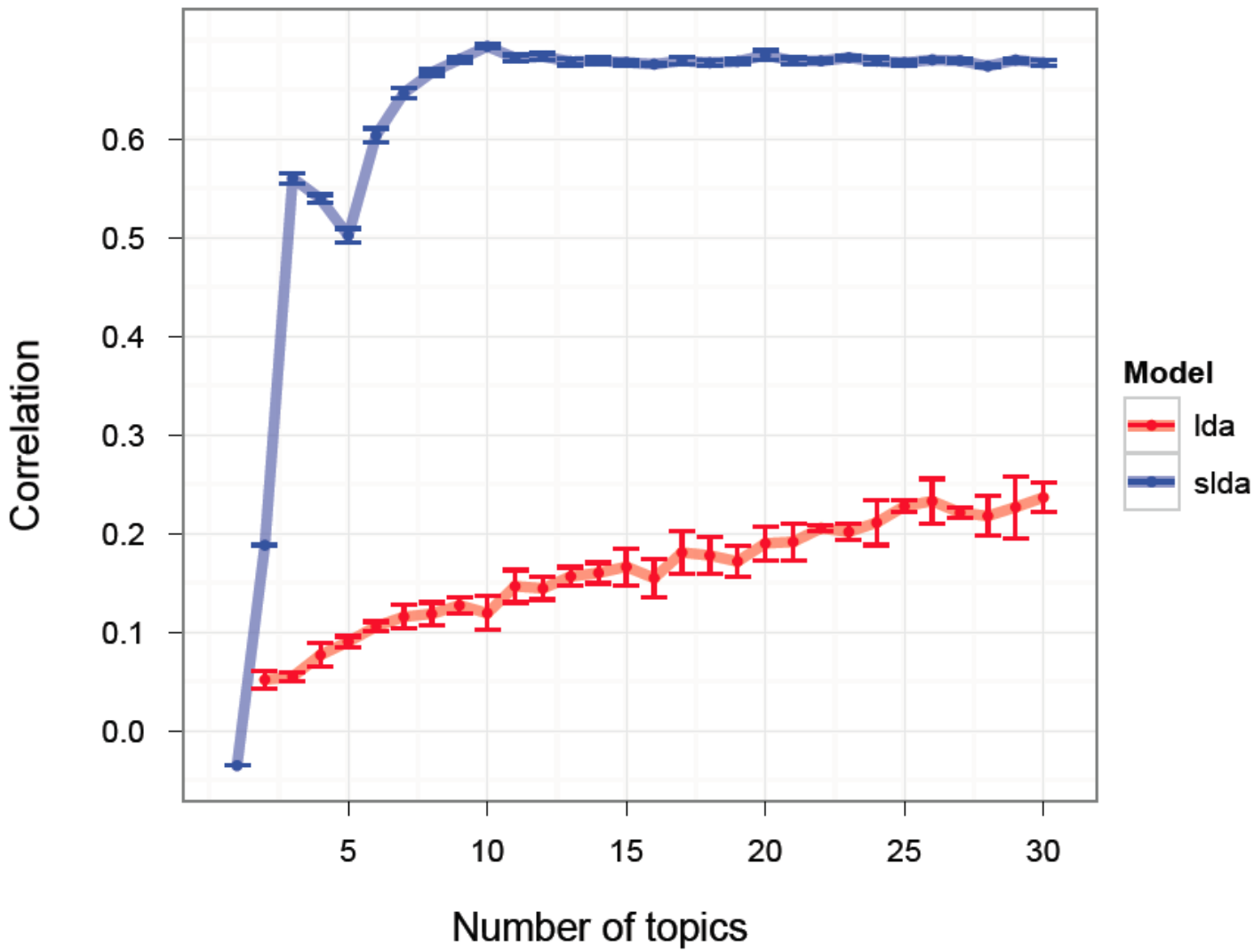
$$\mathrm{E}\left[Y \mid w_{1:N}, \alpha, \beta_{1:K}, \eta, \sigma^2\right] = \eta^\top \mathrm{E}\left[\bar{Z} \mid w_{1:N}\right]$$

Example: Movie Reviews

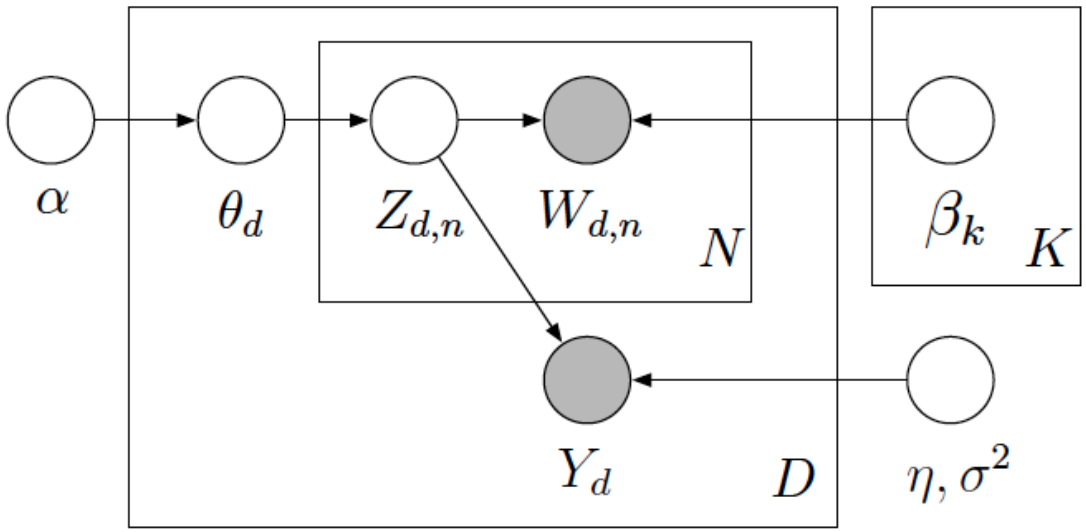


- 10-topic sLDA model on movie reviews (Pang and Lee, 2005).
- Response: number of stars associated with each review
- Each component of coefficient vector η is associated with a topic.

Held out correlation



Supervised Topic Models



- SLDA enables model-based regression where the predictor "variable" is a text document.
- It can easily be used wherever LDA is used in an unsupervised fashion (e.g., images, genes, music).
- SLDA is a supervised dimension-reduction technique, whereas LDA performs unsupervised dimension reduction.

Relational Topic Models

Dealing with Network data

- Such as citation networks, hyperlinked networks of web-pages, social networks of friends
- Useful predictive models can be developed by analyzing this data

Relational Topic Models

Dealing with Network data

- Such as citation networks, hyperlinked networks of web-pages, social networks of friends
- Useful predictive models can be developed by analyzing this data

Relational Topic Models

- LDA needs to be adapted to a model of content and connection
- RTMs find hidden structure in both types of data

Link Prediction Task using RTM

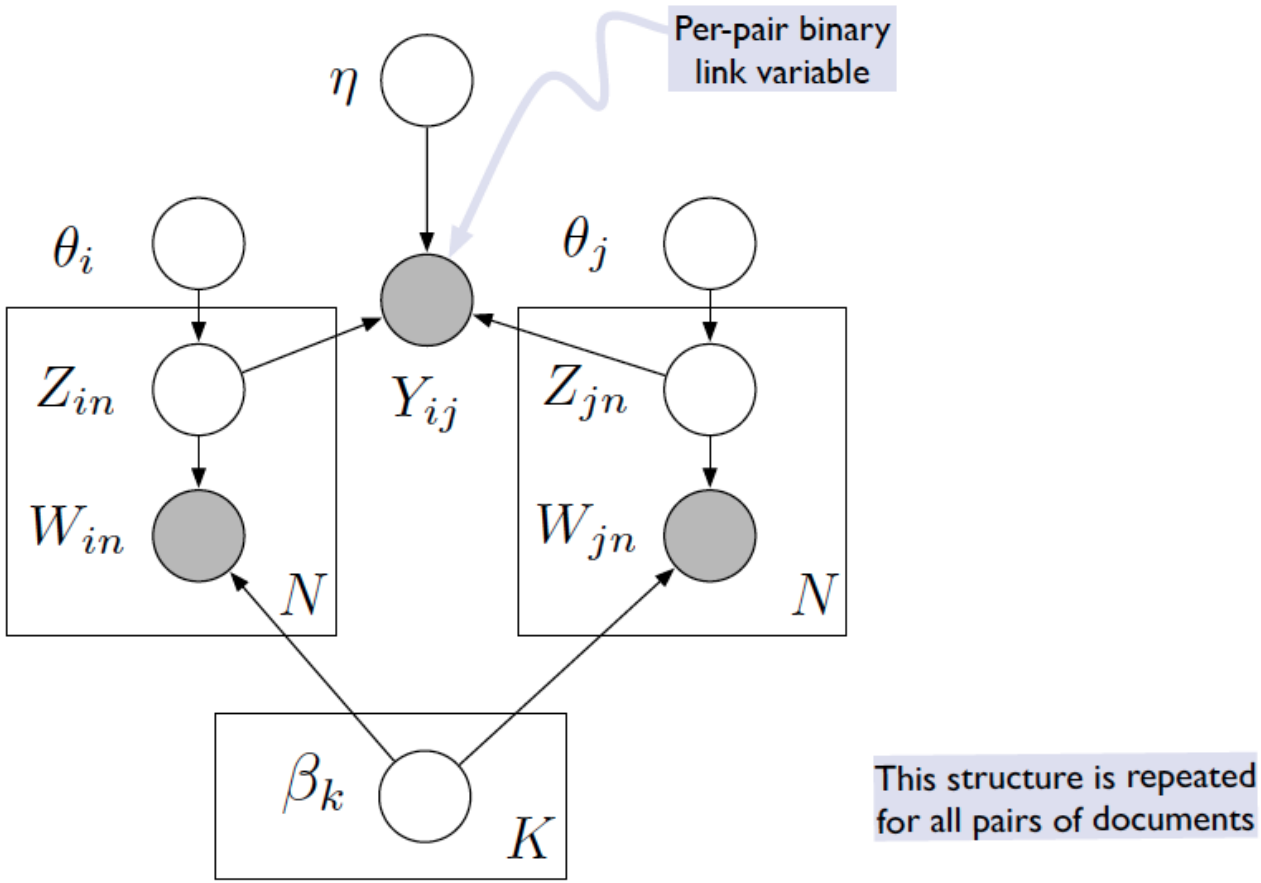
Given a new document, which documents is it likely to link to?

Markov chain Monte Carlo convergence diagnostics: A comparative review
Minorization conditions and convergence rates for Markov chain Monte Carlo
Rates of convergence of the Hastings and Metropolis algorithms
Possible biases induced by MCMC convergence diagnostics
Bounding convergence time of the Gibbs sampler in Bayesian image restoration
Self regenerative Markov chain Monte Carlo
Auxiliary variable methods for Markov chain Monte Carlo with applications
Rate of Convergence of the Gibbs Sampler by Gaussian Approximation
Diagnosing convergence of Markov chain Monte Carlo algorithms

RTM allows for such predictions

- links given the new words of a document
- words given the links of a new document

Relational Topic Models



Formulation ensures that the same latent topic assignments used to generate the content of the documents also generates their link structure.

RTM: Generative Model

1. For each document d:
   (a) Draw topic proportions $\theta_d|\alpha \sim \text{Dir}(\alpha)$.
   (b) For each word $w_{d,n}$:
      i. Draw assignment $z_{d,n}|\theta_d \sim \text{Mult}(\theta_d)$.
      ii. Draw word $w_{d,n}|z_{d,n}, \beta_{1:K} \sim \text{Mult}(\beta_{z_{d,n}})$.
2. For each pair of documents d, d':
   (a) Draw binary link indicator

$$y|\boldsymbol{z}_d, \boldsymbol{z}_{d'} \sim \psi(\cdot|\boldsymbol{z}_d, \boldsymbol{z}_{d'}).$$

Link Probability Function (ψ)

Dependent on the topic assignments that generated their words, z_d and $z_{d'}$.

$$\psi_e(y = 1) = exp(\eta^T(\overline{z_d} \circ \overline{z_{d'}}) + \nu)$$

- $z_d = \frac{1}{N_d}\sum_n z_{d,n}$
- $\circ$ notation denotes the Hadamard (element-wise) product
- Exponential function is being used, they also tried using sigmoid (ψ_σ)
- Link function models each per-pair binary variable as a logistic regression parameterized by $\eta_{1\times K}$ and intercept ν (in case of sigmoid)
- Covariates are constructed by the Hadamard product of $\overline{z_d}$ and $\overline{z_{d'}}$, capturing similarity between the hidden topic representation of the two documents

Inference: How many links to model

- One can fix $y_{d_1,d_2} = 1$ whnever a link is observed between d_1 and d_2 and set $y_{d_1,d_2} = 0$ otherwise

Inference: How many links to model

- One can fix $y_{d_1,d_2} = 1$ whnever a link is observed between d_1 and d_2 and set $y_{d_1,d_2} = 0$ otherwise
- Problem with that approach is that the absence of a link cannot be construed as evidence for $y_{d_1,d_2} = 0$
- So, in these cases, these links are treated as unobserved variables
- Also provides a significant computational advantage

In large social networks like Facebook, the absence of a link between two people doesn't necessarily mean that they are not friends.

Datasets: Summary Statistics

Data set	# of documents	# of words	Number of links	Lexicon size
Cora	2708	49216	5278	1433
WebKB	877	79365	1388	1703
PNAS	2218	11,9162	1577	2239
LocalNews	51	93765	107	1242

Preprocessing

Stop-words were removed and directed links were converted to undirected links, documents with no links were removed.

Datasets: Summary Statistics

What each dataset is about?

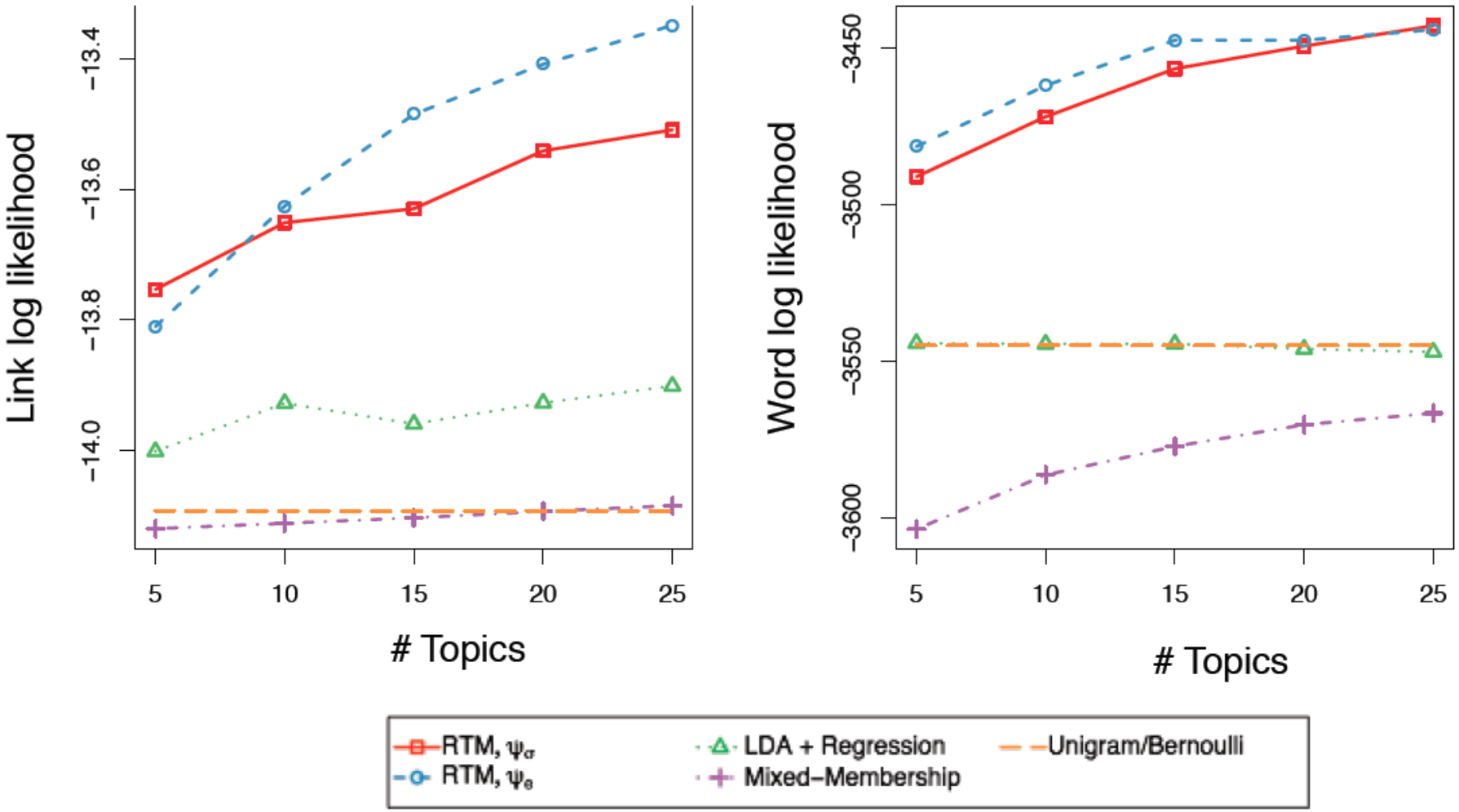
Cora Dataset contains abstracts from the Cora computer science research paper search engine, with links between documents that cite each other

WebKB Dataset contains web pages from the computer science departments of different universities, with links determined from the hyperlinks on each page

PNAS Dataset contains recent abstracts from PNAS. Links between the documents are intra-PNAS citations

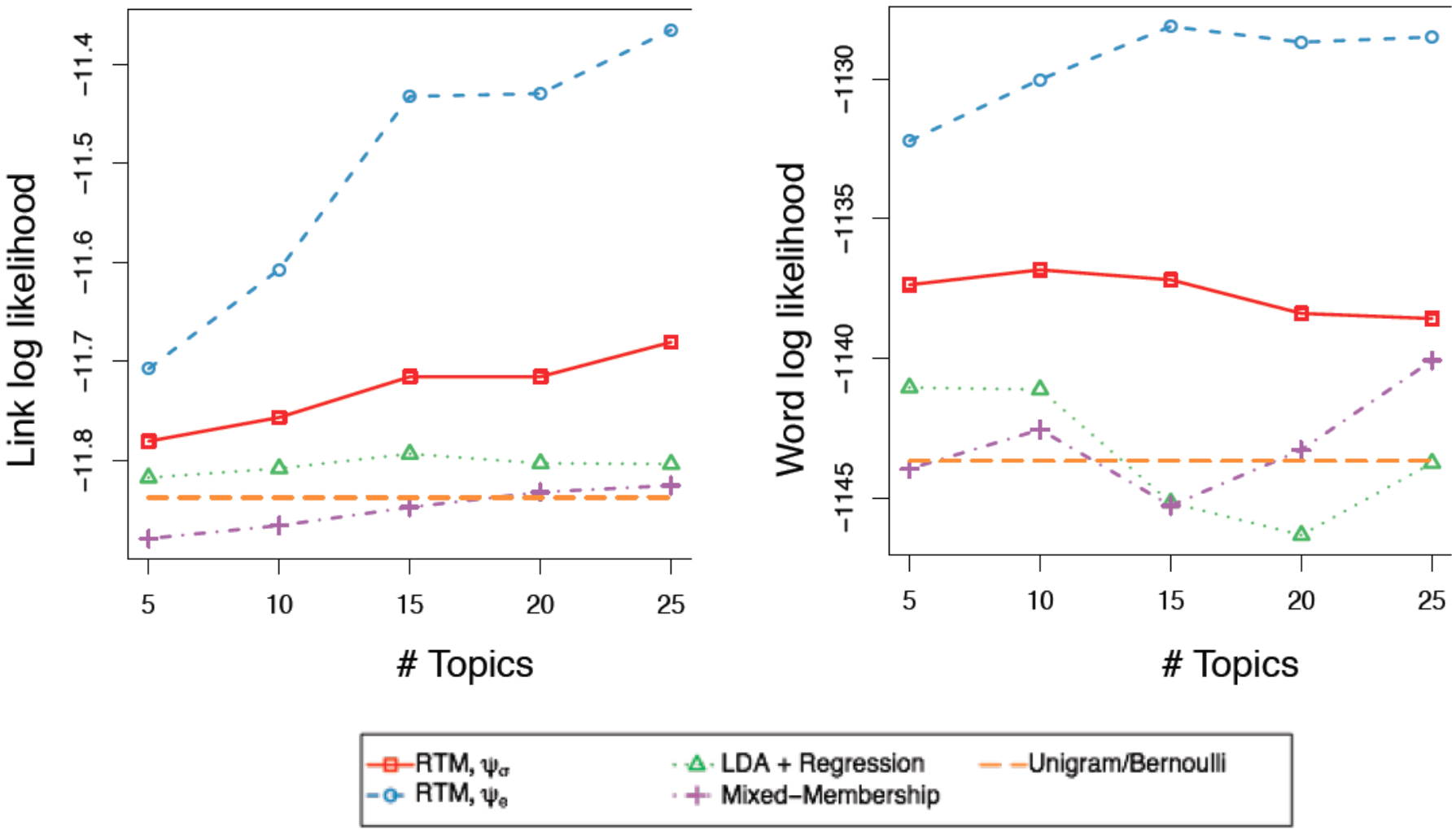
LocalNews Dataset is a corpus of local news culled from various media markets throughout the US. One document for each state, consisting of headlinesa dn summaries from local news. Links determined by geographical adjacency.

Predictve Performance



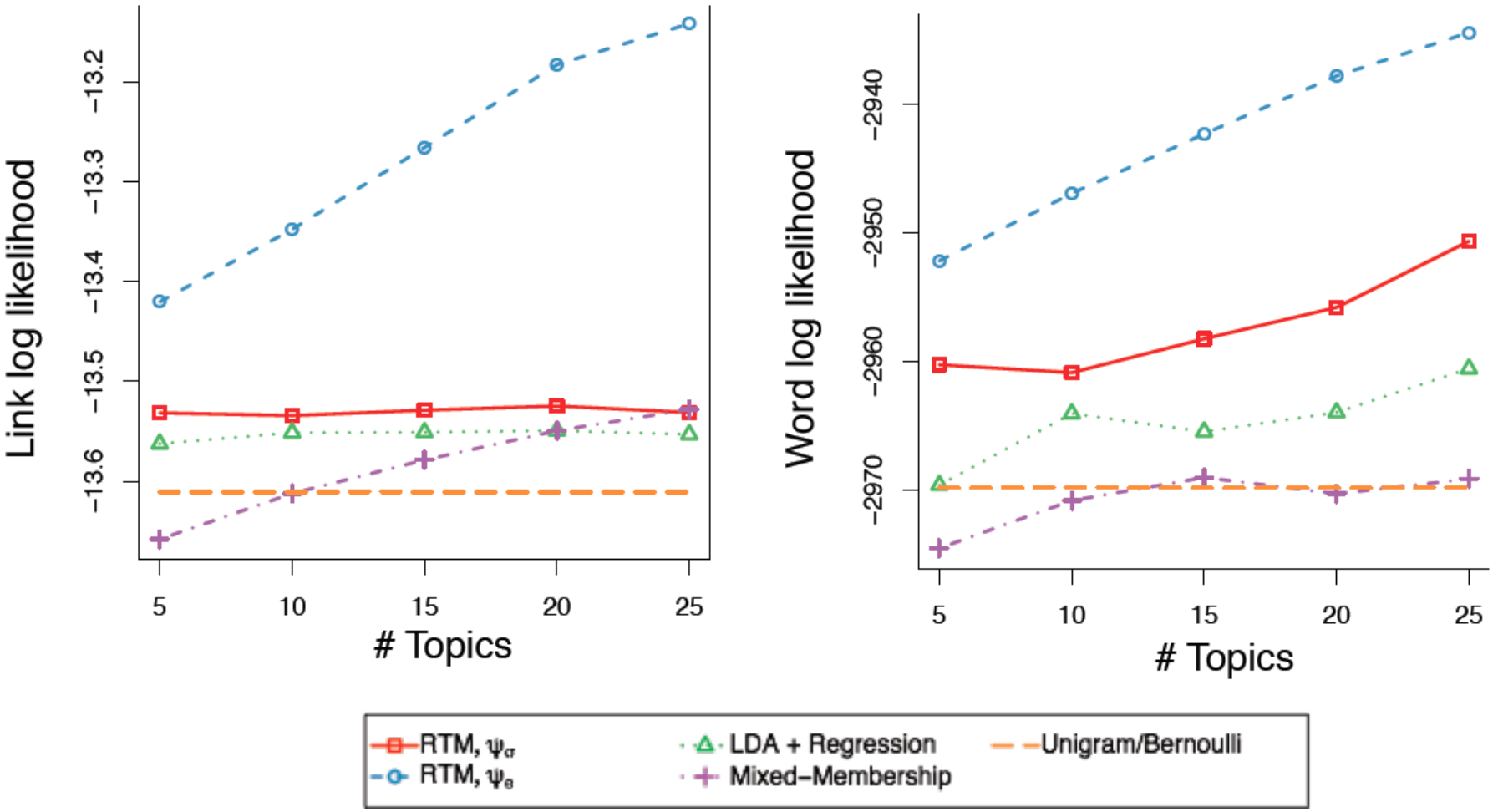
Cora corpus (McCallum et al., 2000)

Predictve Performance



WebKB corpus (Craven et al., 1998)

Predictve Performance



PNAS corpus (courtesy of JSTOR)

Predicting links from documents

Markov chain Monte Carlo convergence diagnostics: A comparative review	
Minorization conditions and convergence rates for Markov chain Monte Carlo Rates of convergence of the Hastings and Metropolis algorithms **Possible biases induced by MCMC convergence diagnostics** Bounding convergence time of the Gibbs sampler in Bayesian image restoration Self regenerative Markov chain Monte Carlo Auxiliary variable methods for Markov chain Monte Carlo with applications **Rate of Convergence of the Gibbs Sampler by Gaussian Approximation** Diagnosing convergence of Markov chain Monte Carlo algorithms	**RTM (ψ_e)**
Exact Bound for the Convergence of Metropolis Chains Self regenerative Markov chain Monte Carlo **Minorization conditions and convergence rates for Markov chain Monte Carlo** Gibbs-markov models Auxiliary variable methods for Markov chain Monte Carlo with applications Markov Chain Monte Carlo Model Determination for Hierarchical and Graphical Models Mediating instrumental variables A qualitative framework for probabilistic inference Adaptation for Self Regenerative MCMC	**LDA + Regression**

Given a new document, which documents is it likely to link to?

Predicting links from documents

Competitive environments evolve better solutions for complex tasks	
Coevolving High Level Representations A Survey of Evolutionary Strategies **Genetic Algorithms in Search, Optimization and Machine Learning** **Strongly typed genetic programming in evolving cooperation strategies** Solving combinatorial problems using evolutionary algorithms A promising genetic algorithm approach to job-shop scheduling... Evolutionary Module Acquisition An Empirical Investigation of Multi-Parent Recombination Operators...	**RTM (ψ_e)**
A New Algorithm for DNA Sequence Assembly Identification of protein coding regions in genomic DNA Solving combinatorial problems using evolutionary algorithms A promising genetic algorithm approach to job-shop scheduling... A genetic algorithm for passive management The Performance of a Genetic Algorithm on a Chaotic Objective Function Adaptive global optimization with local search Mutation rates as adaptations	**LDA + Regression**

Given a new document, which documents is it likely to link to?

Using 'lda' package in R

```
> example <- c("I am the very model of a modern major
+ general", "I have a major headache")
> corpus <- lexicalize(example, lower=TRUE)
> corpus$documents
[[1]]
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
[1,]    0    1    2    3    4    5    6    7    8     9
[2,]    1    1    1    1    1    1    1    1    1     1

[[2]]
     [,1] [,2] [,3] [,4] [,5]
[1,]    0   10    6    8   11
[2,]    1    1    1    1    1
```

Using R package

```
> corpus$vocab
 [1] "i"        "am"       "the"      "very"     "model"    "of"
 [7] "a"  "modern""major"    "general"  "have""headache"
##Take the citations for the first few documents of Cora
> data(cora.cites)
> links<- cora.cites[1:5]
> links
[[1]]
[1] 484 389 ## Papers 484 and 389 cite paper 1
[[2]]
integer(0) ## Paper 2 is not cited by any paper
[[3]]
integer(0)
[[4]]
[1] 177 416 533
[[5]]
[1] 153
```

Using R package

```
> links.as.edgelist(links)
##cited paper, citing paper
      [,1] [,2]
[1,]    0  484
[2,]    0  389
[3,]    3  177
[4,]    3  416
[5,]    3  533
[6,]    4  153
```

Using R package for Relational Topic Models

```
data(cora.documents)
data(cora.vocab)
data(cora.titles)
data(cora.cites)

## Fit an RTM model.
rtm.model <- rtm.collapsed.gibbs.sampler(
cora.documents,##A collection of docs in lda format
cora.cites,##link vector
8,##number of latent topics
cora.vocab,##vocabulary words associated with word indices
35,##number of sweeps of Gibbs sampling
0.1, 0.1, 3##alpha, beta, eta)
```

Using R package for Relational Topic Models

```
## Fit an LDA model by setting beta to zero.
lda.model <- rtm.collapsed.gibbs.sampler(
cora.documents,cora.cites,8,cora.vocab,
35, 0.1, 0.1, 0)

## Randomly sample 100 edges.
edges <- links.as.edgelist(cora.cites)
sampled.edges <- edges[sample
(dim(edges)[1], 100),]

##How sampling works...
> dim(edges)
[1] 4356    2
> sample(4356,100)
  [1] 3803 2668  283 2840 3523 2211 4111  ...
```

Using R package for Relational Topic Models

```
rtm.similarity <- predictive.link.probability
(sampled.edges,##100 samples
rtm.model$document_sums,##KxD matrix topic proportions
0.1, 3##alpha, eta)

lda.similarity <- predictive.link.probability
(sampled.edges,lda.model$document_sums,
0.1, 3)

## Compute how many times each document was cited.
cite.counts <- table(factor(edges[,1],
levels=1:dim(rtm.model$document_sums)[2]))

## And which topic is most expressed by the cited document.
max.topic <- apply(rtm.model$document_sums, 2, which.max)
```

Using R package for Relational Topic Models

```
qplot(lda.similarity,
      rtm.similarity,
      size = log(cite.counts[sampled.edges[,1]]),
      colour = factor(max.topic[sampled.edges[,2]]),
      xlab = "LDA predicted link probability",
      ylab = "RTM predicted link probability",
      xlim=c(0,1), ylim=c(0,1)) +
  scale_size(name="log(Number of citations)") +
  scale_colour_hue(name="Max RTM topic of citing document")
```

Final Output