Adding Domain Knowledge to Latent Topic Models

Xiaojin Zhu

Department of Computer Sciences
University of Wisconsin-Madison

2011

Acknowledgments

David M. Andrzejewski
did most work; now postdoc at Livermore National Laboratory

Mark Craven Ben Liblit Ben Recht
University of Wisconsin-Madison

The Wisconsin Alumni Research Foundation, NSF CAREER Award IIS-0953219, AFOSR FA9550-09-1-0313, and NIH/NLM R01 LM07050

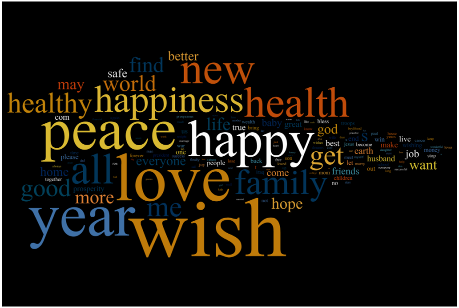
New Year's Eve, Times Square

The Wish Corpus

[Goldberg *et al.*, NAACL 2009]

- Peace on earth
- own a brewery
- I hope I get into Univ. of Penn graduate school.
- The safe return of my friends in Iraq
- find a cure for cancer
- To lose weight and get a boyfriend
- I Hope Barack Obama Wins the Presidency
- To win the lottery!

Corpus-wide word frequencies

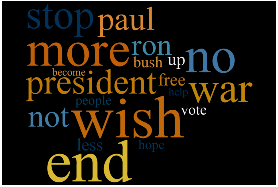


Some Topics by Latent Dirichlet Allocation (LDA)

[Blei *et al.*, JMLR 2003]

$p(\text{word} \mid \text{topic})$

“troops”



“election”

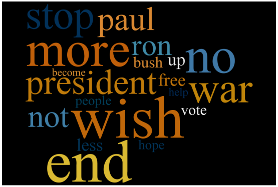
“love”

Some Topics by Latent Dirichlet Allocation (LDA)

[Blei *et al.*, JMLR 2003]

$p(\text{word} \mid \text{topic})$

"troops"



"election"

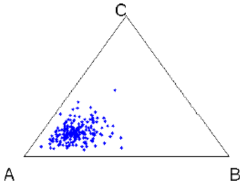
"love"

Major applications: exploratory data analysis

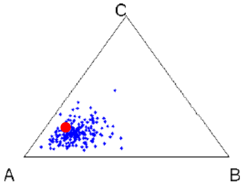
- Research trends [Wang & McCallum, 2006]
- Scientific influence [Gerrish & Blei, 2009]
- Matching papers to reviewers [Mimno & McCallum, 2007]

Quick Statistics Review

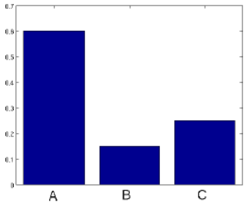
Dirichlet
[20, 5, 5]



Dirichlet
[20, 5, 5]

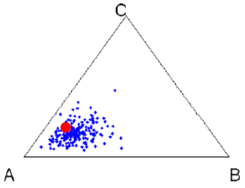


Multinomial
[0.6, 0.15, 0.25]

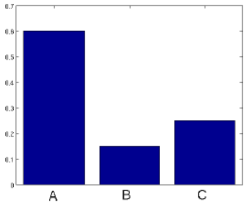


Quick Statistics Review

Dirichlet
[20, 5, 5]



Multinomial
[0.6, 0.15, 0.25]

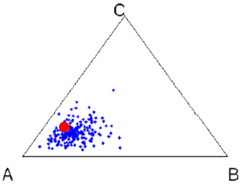


Observed counts
[3, 1, 2]

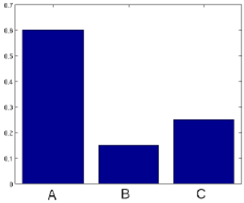
A, A, B, C, A, C

Quick Statistics Review

Dirichlet
[20, 5, 5]



Multinomial
[0.6, 0.15, 0.25]

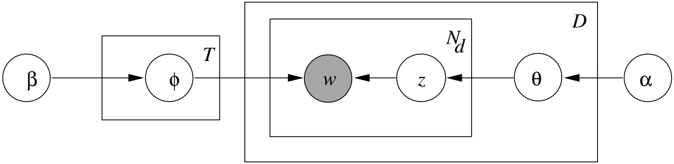


Observed counts
[3, 1, 2]

A, A, B, C, A, C

CONJUGACY!

Latent Dirichlet Allocation (LDA) Review



A generative model for $p(\phi, \theta, z, w \mid \alpha, \beta)$:

For each topic t

$\phi_t \sim$ Dirichlet(β)

For each document d

$\theta \sim$ Dirichlet(α)

For each word position in d

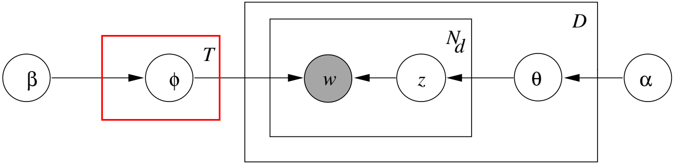
topic $z \sim$ Multinomial(θ)

word $w \sim$ Multinomial(ϕ_z)

Inference goals: $p(z \mid w, \alpha, \beta), \mathrm{argmax}_{\phi,\theta}\, p(\phi, \theta \mid w, \alpha, \beta)$

(reminder on top)

Latent Dirichlet Allocation (LDA) Review



A generative model for $p(\phi, \theta, z, w \mid \alpha, \beta)$:

For each topic t

$\phi_t \sim$ Dirichlet(β)

For each document d

$\theta \sim$ Dirichlet(α)

For each word position in d

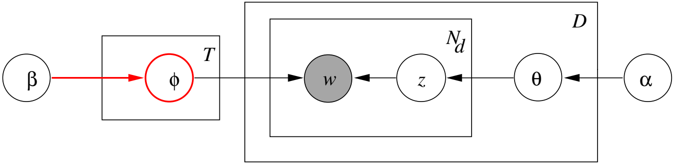
topic $z \sim$ Multinomial(θ)

word $w \sim$ Multinomial(ϕ_z)

Inference goals: $p(z \mid w, \alpha, \beta), \mathrm{argmax}_{\phi,\theta}\, p(\phi, \theta \mid w, \alpha, \beta)$

(reminder on top)

Latent Dirichlet Allocation (LDA) Review



A generative model for $p(\phi, \theta, z, w \mid \alpha, \beta)$:

For each topic t

$\phi_t \sim$ Dirichlet(β)

For each document d

$\theta \sim$ Dirichlet(α)

For each word position in d

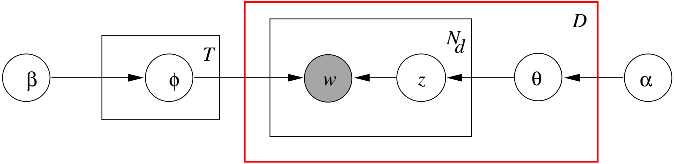
topic $z \sim$ Multinomial(θ)

word $w \sim$ Multinomial(ϕ_z)

Inference goals: $p(z \mid w, \alpha, \beta), \mathrm{argmax}_{\phi,\theta}\, p(\phi, \theta \mid w, \alpha, \beta)$

(reminder on top)

Latent Dirichlet Allocation (LDA) Review



A generative model for $p(\phi, \theta, z, w \mid \alpha, \beta)$:

For each topic t

$\phi_t \sim$ Dirichlet(β)

For each document d

$\theta \sim$ Dirichlet(α)

For each word position in d

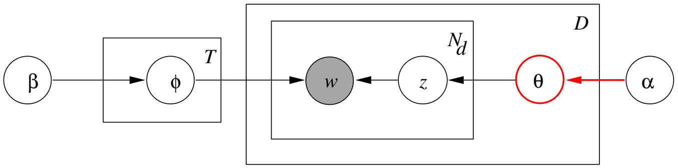
topic $z \sim$ Multinomial(θ)

word $w \sim$ Multinomial(ϕ_z)

Inference goals: $p(z \mid w, \alpha, \beta), \mathrm{argmax}_{\phi,\theta}\, p(\phi, \theta \mid w, \alpha, \beta)$

(reminder on top)

Latent Dirichlet Allocation (LDA) Review



A generative model for $p(\phi, \theta, z, w \mid \alpha, \beta)$:

For each topic t

$\phi_t \sim$ Dirichlet(β)

For each document d

$\theta \sim$ Dirichlet(α)

For each word position in d

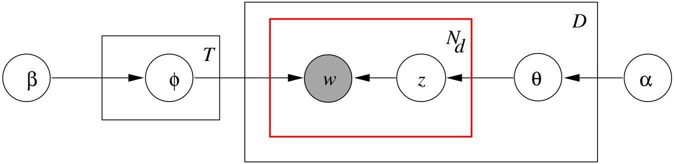
topic $z \sim$ Multinomial(θ)

word $w \sim$ Multinomial(ϕ_z)

Inference goals: $p(z \mid w, \alpha, \beta), \mathrm{argmax}_{\phi,\theta}\, p(\phi, \theta \mid w, \alpha, \beta)$

(reminder on top)

Latent Dirichlet Allocation (LDA) Review



A generative model for $p(\phi, \theta, z, w \mid \alpha, \beta)$:

For each topic t

$\phi_t \sim$ Dirichlet(β)

For each document d

$\theta \sim$ Dirichlet(α)

For each word position in d

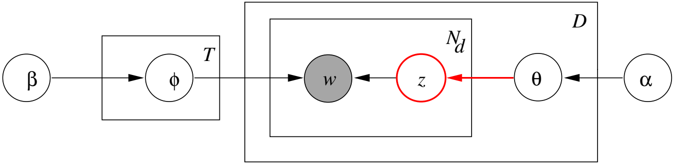
topic $z \sim$ Multinomial(θ)

word $w \sim$ Multinomial(ϕ_z)

Inference goals: $p(z \mid w, \alpha, \beta), \mathrm{argmax}_{\phi,\theta}\, p(\phi, \theta \mid w, \alpha, \beta)$

(reminder on top)

Latent Dirichlet Allocation (LDA) Review



A generative model for $p(\phi, \theta, z, w \mid \alpha, \beta)$:

For each topic t

$\phi_t \sim$ Dirichlet(β)

For each document d

$\theta \sim$ Dirichlet(α)

For each word position in d

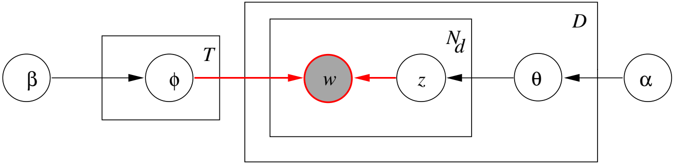
topic $z \sim$ Multinomial(θ)

word $w \sim$ Multinomial(ϕ_z)

Inference goals: $p(z \mid w, \alpha, \beta), \mathrm{argmax}_{\phi,\theta}\, p(\phi, \theta \mid w, \alpha, \beta)$

(reminder on top)

Latent Dirichlet Allocation (LDA) Review



A generative model for $p(\phi, \theta, z, w \mid \alpha, \beta)$:

For each topic t

$\phi_t \sim$ Dirichlet(β)

For each document d

$\theta \sim$ Dirichlet(α)

For each word position in d

topic $z \sim$ Multinomial(θ)

word $w \sim$ Multinomial(ϕ_z)

Inference goals: $p(z \mid w, \alpha, \beta), \mathrm{argmax}_{\phi,\theta}\, p(\phi, \theta \mid w, \alpha, \beta)$

(reminder on top)

When LDA Alone is not Enough

- LDA is unsupervised
- Often domain experts have knowledge in addition to data, want better topics ϕ

When LDA Alone is not Enough

- LDA is unsupervised
- Often domain experts have knowledge in addition to data, want better topics ϕ
- There has been many specialized LDA variants
- This talk: how to do (general) "knowledge + LDA" with 3 models:
  1. topic-in-set
  2. Dirichlet forest
  3. Fold.all

Model 1: Topic-in-Set

Example Application: Statistical Software Debugging

[Andrzejewski *et al.*, ECML 2007], [Zheng *et al.*, ICML 2006]

- Insert predicates into a software:

```
int x = my_func()
if (x > 5) {
    branch_42_true++
    ...
}
else {
    branch_42_false++
    ...
}
```

Example Application: Statistical Software Debugging

[Andrzejewski *et al.*, ECML 2007], [Zheng *et al.*, ICML 2006]

- Insert predicates into a software:

```
int x = my_func()
if (x > 5) {
    branch_42_true++
    ...
}
else {
    branch_42_false++
    ...
}
```

- predicates $\rightarrow$ words w
- a software run $\rightarrow$ doc d
- we know which runs crashed and which didn't (extra knowledge)
- the hope: run LDA on crashed runs, some topics ϕ will correspond to "buggy behaviors"

Hope Crashed

Normal software usage topics dominate. No “bug” topic.

Hope Crashed

Normal software usage topics dominate. No "bug" topic.
Toy example:

- Actual usage (left) and bug (right) topics. Each pixel is a predicate.

Hope Crashed

Normal software usage topics dominate. No “bug” topic.

Toy example:

- Actual usage (left) and bug (right) topics. Each pixel is a predicate.

- Synthetic success (left) and crashed (right) runs

Hope Crashed

Normal software usage topics dominate. No “bug” topic.

Toy example:

- Actual usage (left) and bug (right) topics. Each pixel is a predicate.

- Synthetic success (left) and crashed (right) runs

- LDA topics on crashed runs

ΔLDA

[Andrzejewski *et al.*, ECML 2007]

Model success and crashed runs jointly with T topics:

- fix $t < T$
- for all words in success runs, $z \in \{1 \dots t\}$ (restricted)
- for all words in crashed runs, $z \in \{1 \dots T\}$

New hope: $\phi_1 \dots \phi_t$ usage topics, $\phi_{t+1} \dots \phi_T$ bug topics

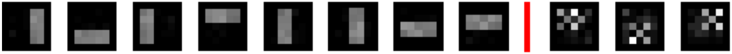
New Hope Succeeds

- Actual usage (left) and bug (right) topics. Each pixel is a predicate.

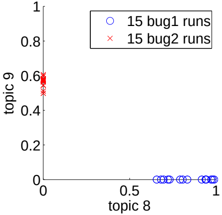
- Synthetic success (left) and crashed (right) runs

- LDA topics on crashed runs

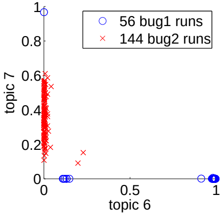
- ΔLDA topics on success and crashed runs



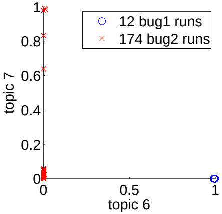
New Hope Succeeds



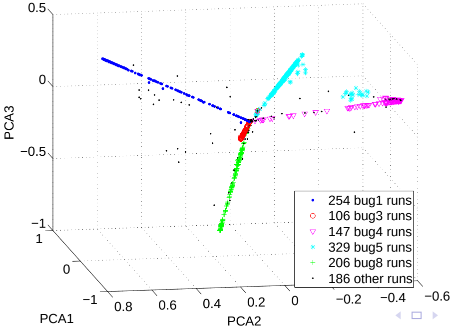
exif



grep



gzip



moss

Generalize ΔLDA to Topic-in-Set

[Andrzejewski & Zhu, NAACL'09 WS]

- The domain knowledge:
  - For each word position i in corpus, we are given a set $C_i \subset \{1 \ldots T\}$, such that $z_i \in C_i$.

Generalize ΔLDA to Topic-in-Set

[Andrzejewski & Zhu, NAACL'09 WS]

- The domain knowledge:
  - For each word position i in corpus, we are given a set $C_i \subset \{1 \ldots T\}$, such that $z_i \in C_i$.
- Very easy to implement in collapsed Gibbs sampling:

$$P(z_i = v | \mathbf{z}_{-i}, \mathbf{w}, \alpha, \beta) \propto$$

$$\left(\frac{n^{(d)}_{-i,v} + \alpha}{\sum_u^T (n^{(d)}_{-i,u} + \alpha)} \right) \left(\frac{n^{(w_i)}_{-i,v} + \beta}{\sum_{w'}^W (\beta + n^{(w')}_{-i,v})} \right) \delta(v \in C_i)$$

  - $n^{(d)}_{-i,v}$ is the number of times topic v is used in document d
  - $n^{(w_i)}_{-i,v}$ is the number of times word w_i is generated by topic v
  - both excluding position i
- Easy to relax the hard constraints

Model 2: Dirichlet Forest

Dirichlet Forest Enables Interactive Topic Modeling

[Andrzejewski *et al.* ICML 2009]

LDA Topics on Wish Corpus:

Topic	Top words sorted by $\phi = p(\text{word}\mid\text{topic})$
0	love i you me and will forever that with hope
1	and health for happiness family good my friends
2	year new happy a this have and everyone years
3	that is it you we be t are as not s will can
4	my to get job a for school husband s that into
5	to more of be and no money stop live people
6	to our the home for of from end safe all come
7	to my be i find want with love life meet man
8	a and healthy my for happy to be have baby
9	a 2008 in for better be to great job president
10	i wish that would for could will my lose can
11	peace and for love all on world earth happiness
12	may god in all your the you s of bless 2008
13	the in to of world best win 2008 go lottery
14	me a com this please at you call 4 if 2 www

Dirichlet Forest Enables Interactive Topic Modeling

[Andrzejewski *et al.* ICML 2009]

LDA Topics on Wish Corpus:

Topic	Top words sorted by $\phi = p(\text{word}\|\text{topic})$
0	love i you me and will forever that with hope
1	and health for happiness family good my friends
2	year new happy a this have and everyone years
3	that is it you we be t are as not s will can
4	my to get job a for school husband s that into
5	to more of be and no money stop live people
6	to our the home for of from end safe all come
7	to my be i find want with love life meet man
8	a and healthy my for happy to be have baby
9	a 2008 in for better be to great job president
10	i wish that would for could will my lose can
11	peace and for love all on world earth happiness
12	may god in all your the you s of bless 2008
13	the in to of world best win 2008 go lottery
14	me a com this please at you call 4 if 2 www

isolate(50 stopwords from existing topics)

Topic	Top words sorted by $\phi = p(\text{word}\|\text{topic})$
0	love forever marry happy together mom back
1	health happiness good family friends prosperity
2	life best live happy long great time ever wonderful
3	out not up do as so what work don was like
4	go school cancer into well free cure college
5	no people stop less day every each take children
6	home safe end troops iraq bring war husband house
7	love peace true happiness hope joy everyone dreams
8	happy healthy family baby safe prosperous everyone
9	better job hope president paul great ron than person
10	make money lose weight meet finally by lots hope married
Isolate	and to for a the year in new all my 2008
12	god bless jesus loved know everyone love who loves
13	peace world earth win lottery around save
14	com call if 4 2 www u visit 1 3 email yahoo
Isolate	i to wish my for and a be that the in

isolate(50 stopwords from existing topics)

Topic	Top words sorted by $\phi = p(\text{word}\mid\text{topic})$
0	love forever marry happy together mom back
1	health happiness good family friends prosperity
2	life best live happy long great time ever wonderful
3	out not up do as so what work don was like
MIXED	go school cancer into well free cure college
5	no people stop less day every each take children
6	home safe end troops iraq bring war husband house
7	love peace true happiness hope joy everyone dreams
8	happy healthy family baby safe prosperous everyone
9	better job hope president paul great ron than person
10	make money lose weight meet finally by lots hope married
Isolate	and to for a the year in new all my 2008
12	god bless jesus loved know everyone love who loves
13	peace world earth win lottery around save
14	com call if 4 2 www u visit 1 3 email yahoo
Isolate	i to wish my for and a be that the in

split([cancer free cure well],[go school into college])

0	love forever happy together marry fall
1	health happiness family good friends
2	life happy best live love long time
3	as not do so what like much don was
4	out make money house up work grow able
5	people no stop less day every each take
6	home safe end troops iraq bring war husband
7	love peace happiness true everyone joy
8	happy healthy family baby safe prosperous
9	better president hope paul ron than person
10	lose meet man hope boyfriend weight finally
Isolate	and to for a the year in new all my 2008
12	god bless jesus loved everyone know loves
13	peace world earth win lottery around save
14	com call if 4 www 2 u visit 1 email yahoo 3
Isolate	i to wish my for and a be that the in me get
Split	job go school great into good college
Split	mom husband cancer hope free son well

split([cancer free cure well],[go school into college])

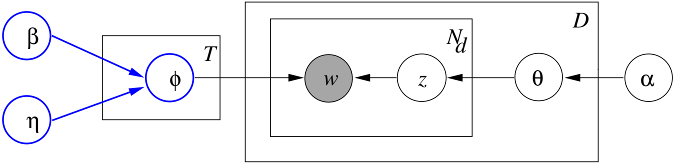
LOVE	love forever happy together marry fall
1	health happiness family good friends
2	life happy best live love long time
3	as not do so what like much don was
4	out make money house up work grow able
5	people no stop less day every each take
6	home safe end troops iraq bring war husband
7	love peace happiness true everyone joy
8	happy healthy family baby safe prosperous
9	better president hope paul ron than person
LOVE	lose meet man hope boyfriend weight finally
Isolate	and to for a the year in new all my 2008
12	god bless jesus loved everyone know loves
13	peace world earth win lottery around save
14	com call if 4 www 2 u visit 1 email yahoo 3
Isolate	i to wish my for and a be that the in me get
Split	job go school great into good college
Split	mom husband cancer hope free son well

merge([love ... marry...],[meet ... married...])

(10 words total)

Topic	Top words sorted by $\phi = p(\text{word}\vert\text{topic})$
Merge	love lose weight together forever marry meet
success	health happiness family good friends prosperity
life	life happy best live time long wishes ever years
-	as do not what someone so like don much he
money	out make money up house work able pay own lots
people	no people stop less day every each other another
iraq	home safe end troops iraq bring war return
joy	love true peace happiness dreams joy everyone
family	happy healthy family baby safe prosperous
vote	better hope president paul ron than person bush
Isolate	and to for a the year in new all my
god	god bless jesus everyone loved know heart christ
peace	peace world earth win lottery around save
spam	com call if u 4 www 2 3 visit 1
Isolate	i to wish my for and a be that the
Split	job go great school into good college hope move
Split	mom hope cancer free husband son well dad cure

From LDA to Dirichlet Forest



For each topic t

$\cancel{\phi_t \sim \text{Dirichlet}(\beta)}$ $\phi_t \sim \text{DirichletForest}(\beta, \eta)$

For each doc d

$\theta \sim \text{Dirichlet}(\alpha)$

For each word w

$z \sim \text{Multinomial}(\theta)$

$w \sim \text{Multinomial}(\phi_z)$

Richer Knowledge Enabled by Dirichlet Forest

Two pairwise relational primitives:

- Must-Link(u, v)
  - e.g., "school" and "college" should be in the same topic
  - encoded as $\phi_t(u) \approx \phi_t(v)$ for $t = 1 \ldots T$
  - can be both large or both small in a given topic

Richer Knowledge Enabled by Dirichlet Forest

Two pairwise relational primitives:

- Must-Link(u, v)
  - e.g., "school" and "college" should be in the same topic
  - encoded as $\phi_t(u) \approx \phi_t(v)$ for $t = 1 \ldots T$
  - can be both large or both small in a given topic
- Cannot-Link(u, v)
  - e.g., "college" and "cure" should not be in the same topic
  - no topic has $\phi_t(u), \phi_t(v)$ both high

Richer Knowledge Enabled by Dirichlet Forest

Two pairwise relational primitives:

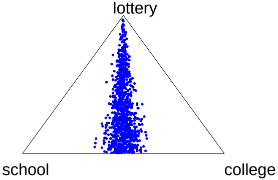
- Must-Link(u, v)
  - e.g., "school" and "college" should be in the same topic
  - encoded as $\phi_t(u) \approx \phi_t(v)$ for $t = 1 \dots T$
  - can be both large or both small in a given topic
- Cannot-Link(u, v)
  - e.g., "college" and "cure" should not be in the same topic
  - no topic has $\phi_t(u), \phi_t(v)$ both high

Complex knowledge:

- split $=$ CL between groups, ML within group
- merge $=$ ML between groups
- isolate $=$ CL to all high prob words in all topics

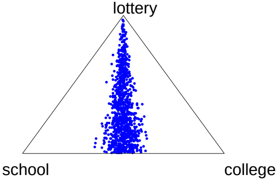
The Dirichlet Distribution is Insufficient for Must-Link

- Must-Link(school, college) with vocabulary {school, college, lottery}
- desired topics ϕ distribution on 3-simplex

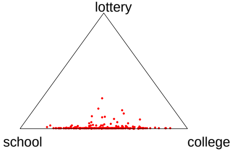


The Dirichlet Distribution is Insufficient for Must-Link

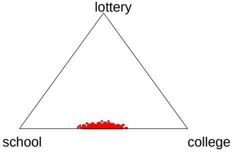
- Must-Link(school, college) with vocabulary {school, college, lottery}
- desired topics ϕ distribution on 3-simplex



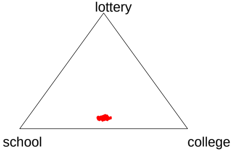
- cannot be represented by a Dirichlet



Dir(5,5,0.1)



Dir(50,50,1)

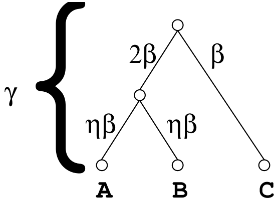


Dir(500,500,100)

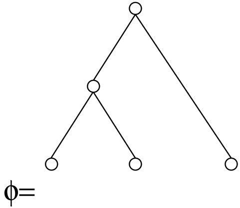
The Dirichlet Tree

[Dennis III 1991], [Minka 1999]

- Dirichlet variance $V(i) = \frac{E(i)(1-E(i))}{1+\sum_j \beta_j}$ tied to mean $E(i) = \frac{\beta_i}{\sum_j \beta_j}$
- More flexible: $\phi \sim$ DirichletTree(γ)
  - Draw multinomial at each internal node
  - Multiply probabilities from leaf to root



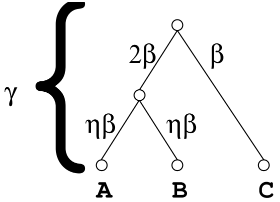
$(\beta = 1, \eta = 50)$



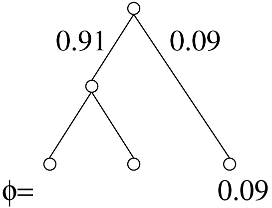
The Dirichlet Tree

[Dennis III 1991], [Minka 1999]

- Dirichlet variance $V(i) = \frac{E(i)(1-E(i))}{1+\sum_j \beta_j}$ tied to mean $E(i) = \frac{\beta_i}{\sum_j \beta_j}$
- More flexible: $\phi \sim \text{DirichletTree}(\gamma)$
  - Draw multinomial at each internal node
  - Multiply probabilities from leaf to root



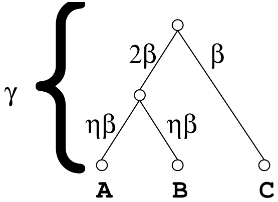
$(\beta = 1, \eta = 50)$



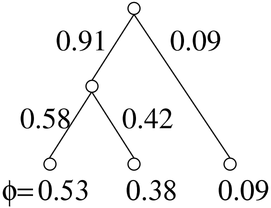
The Dirichlet Tree

[Dennis III 1991], [Minka 1999]

- Dirichlet variance $V(i) = \frac{E(i)(1-E(i))}{1+\sum_j \beta_j}$ tied to mean $E(i) = \frac{\beta_i}{\sum_j \beta_j}$
- More flexible: $\phi \sim$ DirichletTree(γ)
  - Draw multinomial at each internal node
  - Multiply probabilities from leaf to root



$(\beta = 1, \eta = 50)$



The Dirichlet Tree

$$p(\phi|\gamma) = \left(\prod_k^L {\phi^{(k)}}^{\gamma^{(k)}-1} \right) \left(\prod_s^I \frac{\Gamma\left(\sum_k^{C(s)} \gamma^{(k)}\right)}{\prod_k^{C(s)} \Gamma\left(\gamma^{(k)}\right)} \left(\sum_k^{L(s)} \phi^{(k)} \right)^{\Delta(s)} \right)$$

- $C(s) =$ children of s, $L(k) =$ leaves of subtree k
- $\Delta(s) = \mathrm{InDegree}(s) - \mathrm{OutDegree}(s)$ being zero recovers Dirichlet

The Dirichlet Tree

$$p(\phi|\gamma) = \left(\prod_k^L \phi^{(k)\gamma^{(k)}-1}\right)\left(\prod_s^I \frac{\Gamma\left(\sum_k^{C(s)} \gamma^{(k)}\right)}{\prod_k^{C(s)} \Gamma\left(\gamma^{(k)}\right)} \left(\sum_k^{L(s)} \phi^{(k)}\right)^{\Delta(s)}\right)$$

- $C(s)$ = children of s, $L(k)$ = leaves of subtree k
- $\Delta(s) = \mathrm{InDegree}(s) - \mathrm{OutDegree}(s)$ being zero recovers Dirichlet

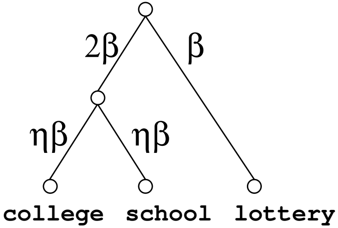
Dirichlet tree is conjugate to multinomial

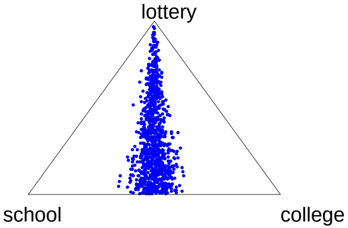
$$p(\mathbf{w}|\gamma) = \prod_s^I \left(\frac{\Gamma\left(\sum_k^{C(s)} \gamma^{(k)}\right)}{\Gamma\left(\sum_k^{C(s)} \left(\gamma^{(k)} + n^{(k)}\right)\right)} \prod_k^{C(s)} \frac{\Gamma\left(\gamma^{(k)} + n^{(k)}\right)}{\Gamma(\gamma^{(k)})}\right)$$

Encode Must-Links with a Dirichlet Tree

[Andrzejewski *et al.* ICML 2009]

1. Compute Must-Link transitive closures
2. Each transitive closures is a subtree with edges $\eta\beta$
3. The root connects to
   - each transitive closure with edge $|\text{closure}|\beta$
   - each individual word not in any closure with edge β

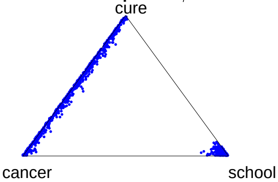




$(\beta = 1, \eta = 50)$

The Dirichlet Distribution is Insufficient for Cannot-Link

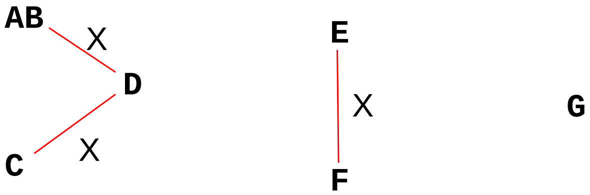
- Cannot-Link(school, cancer) with vocabulary {school, cancer, cure}
- desired topics ϕ distribution on 3-simplex



- cannot be represented by a Dirichlet ($\beta < 1$ not robust)
- cannot be represented by a Dirichlet Tree
- requires mixture of Dirichlet Trees (Dirichlet Forest)

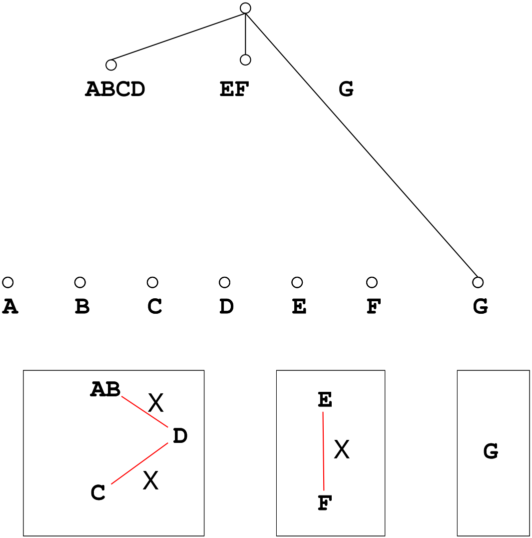
Dirichlet Forest Example

- A toy example:
  - Vocabulary: {A,B,C,D,E,F,G}
  - Must-Link(A,B)
  - Cannot-Link(A,D), Cannot-Link(C,D), Cannot-Link(E,F)
- Cannot-Link graph on Must-Link closures and other words



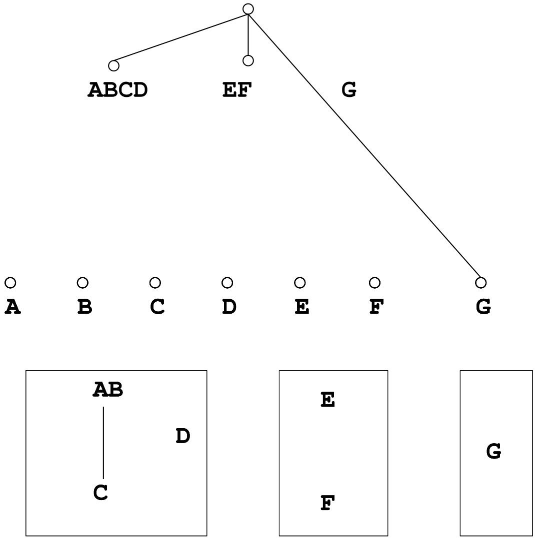
Picking a Dirichlet Tree from the Dirichlet Forest

Identify Cannot-Link connected components (they are independent)



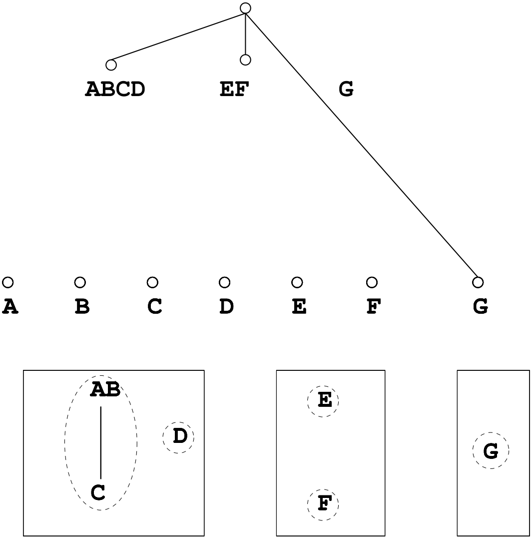
Picking a Dirichlet Tree from the Dirichlet Forest

Flip edges: "Can"-Links within components



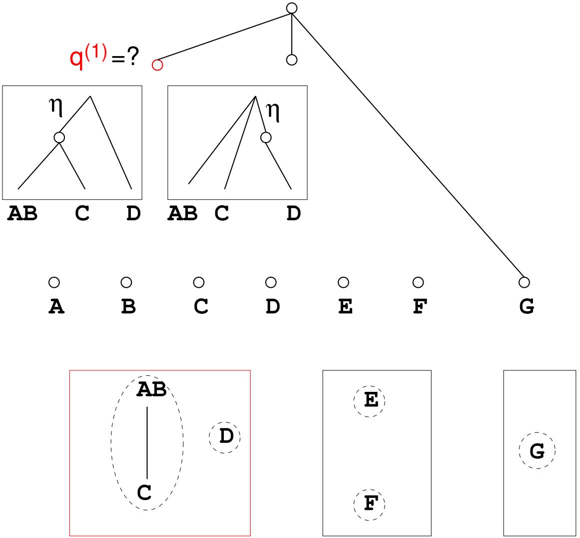
Picking a Dirichlet Tree from the Dirichlet Forest

"Can"-maximal-cliques (no more words can have high prob together)



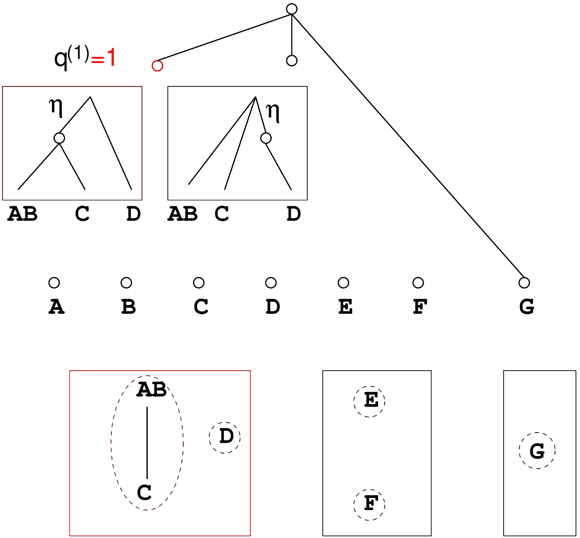
Picking a Dirichlet Tree from the Dirichlet Forest

Pick a Can-clique (subtree) $q^{(1)}$ for the first connected component



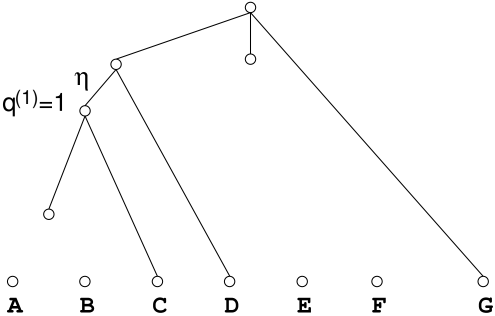
Picking a Dirichlet Tree from the Dirichlet Forest

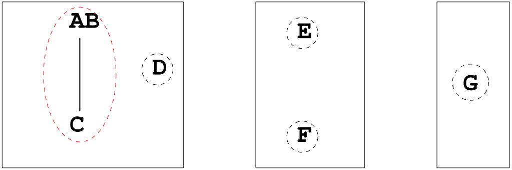
We picked the first Can-clique $q^{(1)} = 1$: give ABC most prob mass



Picking a Dirichlet Tree from the Dirichlet Forest

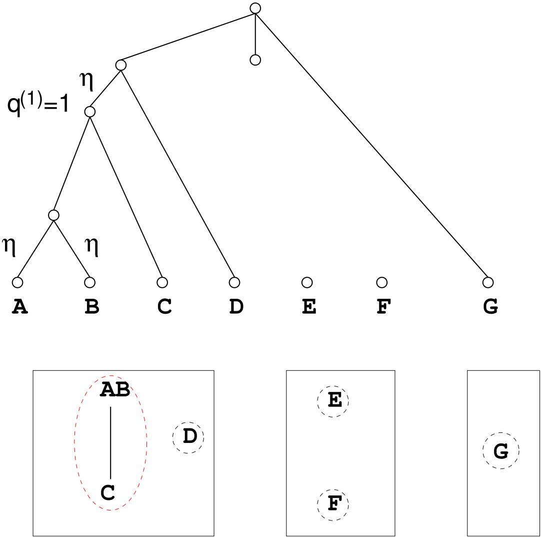
D has little mass, will not occur with any of A, B, C: satisfying Cannot-Links





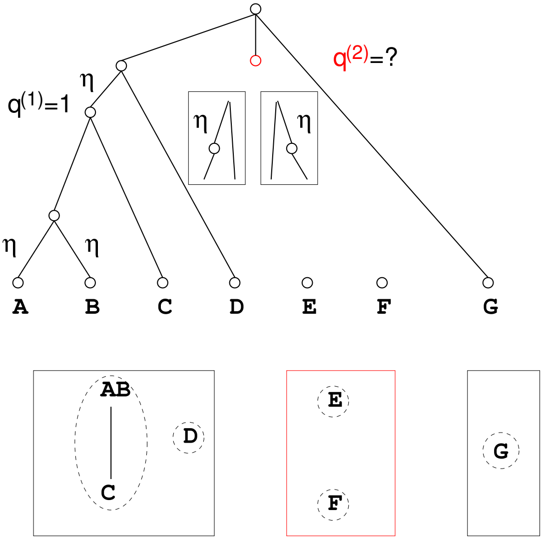
Picking a Dirichlet Tree from the Dirichlet Forest

Mass distribution between (AB) and C flexible; A, B under Must-Link subtree



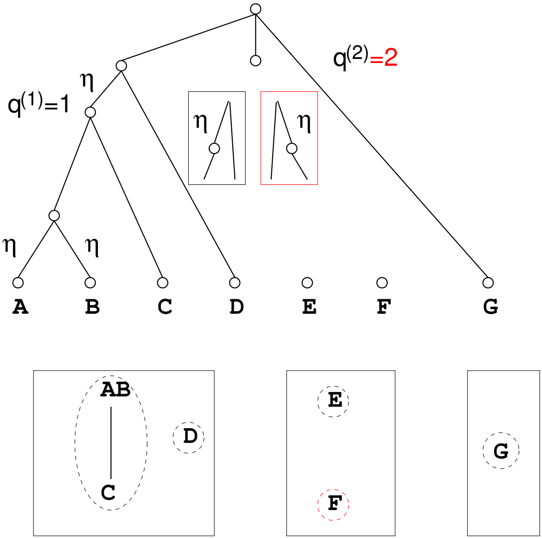
Picking a Dirichlet Tree from the Dirichlet Forest

Pick a Can-clique $q^{(2)}$ for the second connected component



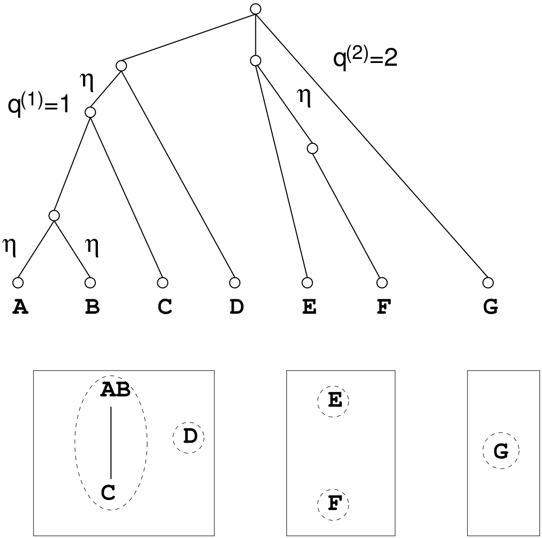
Picking a Dirichlet Tree from the Dirichlet Forest

We picked the 2nd Can-clique $q^{(2)} = 2$



Picking a Dirichlet Tree from the Dirichlet Forest

Here is the chosen subtree: F will get prob mass, not E



Inference with Dirichlet Forest

- In theory the number of Can-cliques is exponential, in practice the largest we've seen is 3
- Collapsed Gibbs sampling over topics $z_1 \dots z_N$ and subtree indices $q_j^{(r)}$ for topic $j = 1 \dots T$ and Cannot-Link connected components r

$$p(z_i = v | \mathbf{z}_{-i}, \mathbf{q}_{1:T}, \mathbf{w}) \propto (n_{-i,v}^{(d)} + \alpha) \prod_s^{I_v(\uparrow i)} \frac{\gamma_v^{(C_v(s\downarrow i))} + n_{-i,v}^{(C_v(s\downarrow i))}}{\sum_k^{C_v(s)} \left(\gamma_v^{(k)} + n_{-i,v}^{(k)}\right)}$$

$$p(q_j^{(r)} = q' | \mathbf{z}, \mathbf{q}_{-j}, \mathbf{q}_j^{(-r)}, \mathbf{w}) \propto \left(\sum_k^{M_{rq'}} \beta_k\right) \prod_s^{I_{j,r=q'}} \left(\frac{\Gamma\left(\sum_k^{C_j(s)} \gamma_j^{(k)}\right)}{\Gamma\left(\sum_k^{C_j(s)} (\gamma_j^{(k)} + n_j^{(k)})\right)} \prod_k^{C_j(s)} \frac{\Gamma(\gamma_j^{(k)} + n_j^{(k)})}{\Gamma(\gamma_j^{(k)})} \right)$$

Model 3: Fold.all

Logic can Encode Very General Knowledge

- Topic-in-set and Dirichlet Forest (ML, CL) not general enough

Logic can Encode Very General Knowledge

- Topic-in-set and Dirichlet Forest (ML, CL) not general enough
- Fold.all = First-Order Logic latent Dirichlet ALLocation
  - easy for domain experts to write rules
  - can describe very general domain knowledge
    - can encode many existing LDA variants
  - efficient inference

Logic can Encode Very General Knowledge

- Topic-in-set and Dirichlet Forest (ML, CL) not general enough
- Fold.all = First-Order Logic latent Dirichlet ALLocation
  - easy for domain experts to write rules
  - can describe very general domain knowledge
    - can encode many existing LDA variants
  - efficient inference
- A hybrid Markov Logic Network (MLN) [Wang & Domingos 2008] [Richardson & Domingos 2006], but with fast stochastic optimization

Domain Knowledge in Logic

- Key hidden predicate: $Z(i,t)$ TRUE if topic $z_i = t$
- Observed predicates (anything goes):
  - $W(i,v)$ TRUE if word $w_i = v$
  - $D(i,j)$ TRUE if word position i is in document j
  - $\mathrm{HasLabel}(j,l)$ TRUE if document j has label l
  - $S(i,k)$ TRUE if word position i is in document k
  - $\ldots$

Domain Knowledge in Logic

- Key hidden predicate: $Z(i,t)$ TRUE if topic $z_i = t$
- Observed predicates (anything goes):
  - $W(i,v)$ TRUE if word $w_i = v$
  - $D(i,j)$ TRUE if word position i is in document j
  - $\mathrm{HasLabel}(j,l)$ TRUE if document j has label l
  - $S(i,k)$ TRUE if word position i is in document k
  - $\ldots$
- Domain knowledge-base $(\lambda_1, \psi_1) \ldots (\lambda_L, \psi_L)$
  - rules ψ
  - positive weights λ indicate strength of rule

Example:

$$\lambda_1 = 1, \psi_1 = \text{“}\forall i : W(i, \text{embryo}) \Rightarrow Z(i,3)''$$

$$\lambda_2 = 100, \psi_2 = \text{“}\forall i,j,t : W(i, \text{movie}) \wedge W(j, \text{film}) \Rightarrow \neg(Z(i,t) \wedge Z(j,t))''$$

Propositionalization

- Let $G(\psi)$ be all ground formulas of ψ.
  - $\psi =$ "$\forall i, j, t : W(i, \text{movie}) \wedge W(j, \text{film}) \Rightarrow \neg(Z(i,t) \wedge Z(j,t))$"
  - One ground formula $g \in G(\psi)$ is $W(123, \text{movie}) \wedge W(456, \text{film}) \Rightarrow \neg(Z(123, 9) \wedge Z(456, 9))$

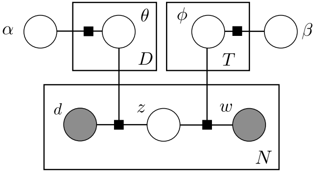
Propositionalization

- Let $G(\psi)$ be all ground formulas of ψ.
  - $\psi =$ "$\forall i, j, t : W(i, \text{movie}) \wedge W(j, \text{film}) \Rightarrow \neg(Z(i, t) \wedge Z(j, t))$"
  - One ground formula $g \in G(\psi)$ is $W(123, \text{movie}) \wedge W(456, \text{film}) \Rightarrow \neg(Z(123, 9) \wedge Z(456, 9))$
- $|G(\psi)|$ combinatorial.
- Let

$$\mathbb{1}_g(\mathbf{z}) = \begin{cases} 1, & \text{if } g \text{ is TRUE under } \mathbf{z} \\ 0, & \text{otherwise.} \end{cases}$$

Fold.all = LDA + MLN

[Andrzejewski *et al.* IJCAI 2011]

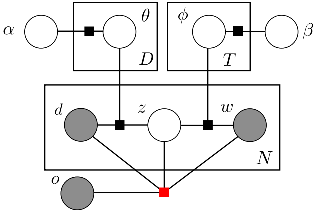


$$p(\mathbf{z}, \phi, \theta \mid \mathbf{w}, \alpha, \beta)$$

$$\propto \left(\prod_t^T p(\phi_t | \beta) \right) \left(\prod_j^D p(\theta_j | \alpha) \right) \left(\prod_i^N \phi_{z_i}(w_i) \theta_{d_i}(z_i) \right)$$

Fold.all = LDA + MLN

[Andrzejewski *et al.* IJCAI 2011]



$$
\begin{aligned}
& p(\mathbf{z}, \phi, \theta \mid \mathbf{w}, \alpha, \beta) \\
\propto \quad & \left(\prod_t^T p(\phi_t|\beta)\right)\left(\prod_j^D p(\theta_j|\alpha)\right)\left(\prod_i^N \phi_{z_i}(w_i)\theta_{d_i}(z_i)\right) \\
& \times \exp\left[\sum_l^L \sum_{g \in G(\psi_l)} \lambda_l \mathbb{1}_g(\mathbf{z})\right]
\end{aligned}
$$

Fold.all Inference

MAP estimate, non-convex objective

$$Q(\mathbf{z}, \phi, \theta) \equiv \sum_t^T \log p(\phi_t|\beta) + \sum_j^D \log p(\theta_j|\alpha) + \sum_i^N \log \phi_{z_i}(w_i)\theta_{d_i}(z_i) + \sum_l^L \sum_{g \in G(\psi_l)} \lambda_l \mathbb{1}_g(\mathbf{z})$$

Alternating optimization. Repeat:

- fixing $\mathbf{z}$, let $(\phi^*, \theta^*) \leftarrow \mathrm{argmax}_{\phi,\theta}\ Q(\mathbf{z}, \phi, \theta)$ (easy)
- fixing ϕ, θ, let $\mathbf{z}^* \leftarrow \mathrm{argmax}_z\ Q(\mathbf{z}, \phi, \theta)$ (integer)

Optimizing $\mathbf{z}$ Step 1: Relax $\mathbb{1}_g(\mathbf{z})$

$g = \mathtt{Z}(i,1) \vee \neg\mathtt{Z}(j,2)$, and $t \in \{1,2,3\}$

1. Take complement $\neg g$ $\qquad \neg\mathtt{Z}(i,1) \wedge \mathtt{Z}(j,2)$

Optimizing $\mathbf{z}$ Step 1: Relax $\mathbb{1}_g(\mathbf{z})$

$g = \mathtt{Z}(i,1) \vee \neg\mathtt{Z}(j,2)$, and $t \in \{1,2,3\}$

1. Take complement $\neg g$ $\qquad \neg\mathtt{Z}(i,1) \wedge \mathtt{Z}(j,2)$
2. Remove negations $(\neg g)_+$ $\qquad (\mathtt{Z}(i,2) \vee \mathtt{Z}(i,3)) \wedge \mathtt{Z}(j,2)$

Optimizing $\mathbf{z}$ Step 1: Relax $\mathbb{1}_g(\mathbf{z})$

$g = \mathtt{Z}(i,1) \vee \neg\mathtt{Z}(j,2)$, and $t \in \{1,2,3\}$

1. Take complement $\neg g$ $\qquad \neg\mathtt{Z}(i,1) \wedge \mathtt{Z}(j,2)$
2. Remove negations $(\neg g)_+$ $\qquad (\mathtt{Z}(i,2) \vee \mathtt{Z}(i,3)) \wedge \mathtt{Z}(j,2)$
3. Numeric $z_{it} \in \{0,1\}$ $\qquad (z_{i2} + z_{i3})z_{j2}$

Optimizing $\mathbf{z}$ Step 1: Relax $\mathbb{1}_g(\mathbf{z})$

$g = \mathtt{Z}(i,1) \vee \neg\mathtt{Z}(j,2)$, and $t \in \{1,2,3\}$

1. Take complement $\neg g$ — $\neg\mathtt{Z}(i,1) \wedge \mathtt{Z}(j,2)$
2. Remove negations $(\neg g)_+$ — $(\mathtt{Z}(i,2) \vee \mathtt{Z}(i,3)) \wedge \mathtt{Z}(j,2)$
3. Numeric $z_{it} \in \{0,1\}$ — $(z_{i2} + z_{i3})z_{j2}$
4. Polynomial $\mathbb{1}_g(\mathbf{z})$ — $1 - (z_{i2} + z_{i3})z_{j2}$

Optimizing $\mathbf{z}$ Step 1: Relax $\mathbb{1}_g(\mathbf{z})$

$g = \mathtt{Z}(i,1) \vee \neg\mathtt{Z}(j,2)$, and $t \in \{1,2,3\}$

1. Take complement $\neg g$ — $\neg\mathtt{Z}(i,1) \wedge \mathtt{Z}(j,2)$
2. Remove negations $(\neg g)_+$ — $(\mathtt{Z}(i,2) \vee \mathtt{Z}(i,3)) \wedge \mathtt{Z}(j,2)$
3. Numeric $z_{it} \in \{0,1\}$ — $(z_{i2} + z_{i3})z_{j2}$
4. Polynomial $\mathbb{1}_g(\mathbf{z})$ — $1 - (z_{i2} + z_{i3})z_{j2}$
5. Relax discrete z_{it} — $z_{it} \in \{0,1\} \rightarrow z_{it} \in [0,1]$

Optimizing $\mathbf{z}$ Step 1: Relax $\mathbb{1}_g(\mathbf{z})$

$g = \mathtt{Z}(i,1) \vee \neg\mathtt{Z}(j,2)$, and $t \in \{1,2,3\}$

1. Take complement $\neg g$ — $\neg\mathtt{Z}(i,1) \wedge \mathtt{Z}(j,2)$
2. Remove negations $(\neg g)_+$ — $(\mathtt{Z}(i,2) \vee \mathtt{Z}(i,3)) \wedge \mathtt{Z}(j,2)$
3. Numeric $z_{it} \in \{0,1\}$ — $(z_{i2} + z_{i3})z_{j2}$
4. Polynomial $\mathbb{1}_g(\mathbf{z})$ — $1 - (z_{i2} + z_{i3})z_{j2}$
5. Relax discrete z_{it} — $z_{it} \in \{0,1\} \rightarrow z_{it} \in [0,1]$

Optimizing $\mathbf{z}$ Step 1: Relax $\mathbb{1}_g(\mathbf{z})$

$g = \mathtt{Z}(i,1) \vee \neg \mathtt{Z}(j,2)$, and $t \in \{1,2,3\}$

1. Take complement $\neg g$ — $\neg \mathtt{Z}(i,1) \wedge \mathtt{Z}(j,2)$
2. Remove negations $(\neg g)_+$ — $(\mathtt{Z}(i,2) \vee \mathtt{Z}(i,3)) \wedge \mathtt{Z}(j,2)$
3. Numeric $z_{it} \in \{0,1\}$ — $(z_{i2} + z_{i3}) z_{j2}$
4. Polynomial $\mathbb{1}_g(\mathbf{z})$ — $1 - (z_{i2} + z_{i3}) z_{j2}$
5. Relax discrete z_{it} — $z_{it} \in \{0,1\} \rightarrow z_{it} \in [0,1]$

$$\mathbb{1}_g(\mathbf{z}) = 1 - \prod_{g_i \neq \emptyset} \left(\sum_{\mathtt{Z}(i,t) \in (\neg g_i)_+} z_{it} \right)$$

Optimizing $\mathbf{z}$ Step 2: Stochastic Optimization

- $\sum_l^L |G(\psi_l)| + NT$ terms in Q related to $\mathbf{z}$:

$$\max_{\mathbf{z}} \quad \sum_l^L \sum_{g \in G(\psi_l)} \lambda_l \mathbb{1}_g(\mathbf{z}) + \sum_i^N \sum_t^T z_{it} \log \phi_t(w_i)\theta_{d_i}(t)$$

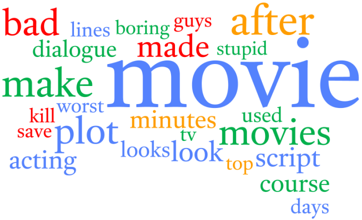
$$\text{s.t.} \quad z_{it} \geq 0, \quad \sum_t^T z_{it} = 1.$$

- Entropic Mirror Descent [Beck & Teboulle, 2003]. Repeat:
  - select a term f at random
  - descent with decreasing step size η

$$z_{it} \leftarrow \frac{z_{it} \exp\left(\eta \nabla_{z_{it}} f\right)}{\sum_{t'} z_{it'} \exp\left(\eta \nabla_{z_{it'}} f\right)}$$

Example: Movie Reviews

$$\forall i, j, t : W(i, \text{movie}) \wedge W(j, \text{film}) \Rightarrow \neg(Z(i,t) \wedge Z(j,t))$$



Generalization and Scalability

Cross-validation

- Training: do Fold.all MAP inference to estimate $\hat{\phi}$
- Testing: use trainset $\hat{\phi}$ to infer testset $\hat{\mathbf{z}}$ (no logic rules)
- Evaluation: testset objective Q
- "-": runs more than 24 hours

Data	Fold.all	LDA	Alchemy	$\sum_l \lvert G(\psi_l)\rvert$
Synth	$\mathbf{9.86}$	-2.18	-1.73	10^5
Comp	$\mathbf{2.40}$	1.19	-	10^4
Con	$\mathbf{2.51}$	1.09	-	10^3
Pol	$\mathbf{5.67}$	$\mathbf{5.67}$	-	10^9
HDG	$\mathbf{10.66}$	3.59	-	10^8

Summary

- "Knowledge + data" for latent topic models
- Increasingly more general models
  - Topic-in-set
  - Dirichlet forest (Must & Cannot links)
  - Fold.all (logic)
- Easy for users
- Scalable inference