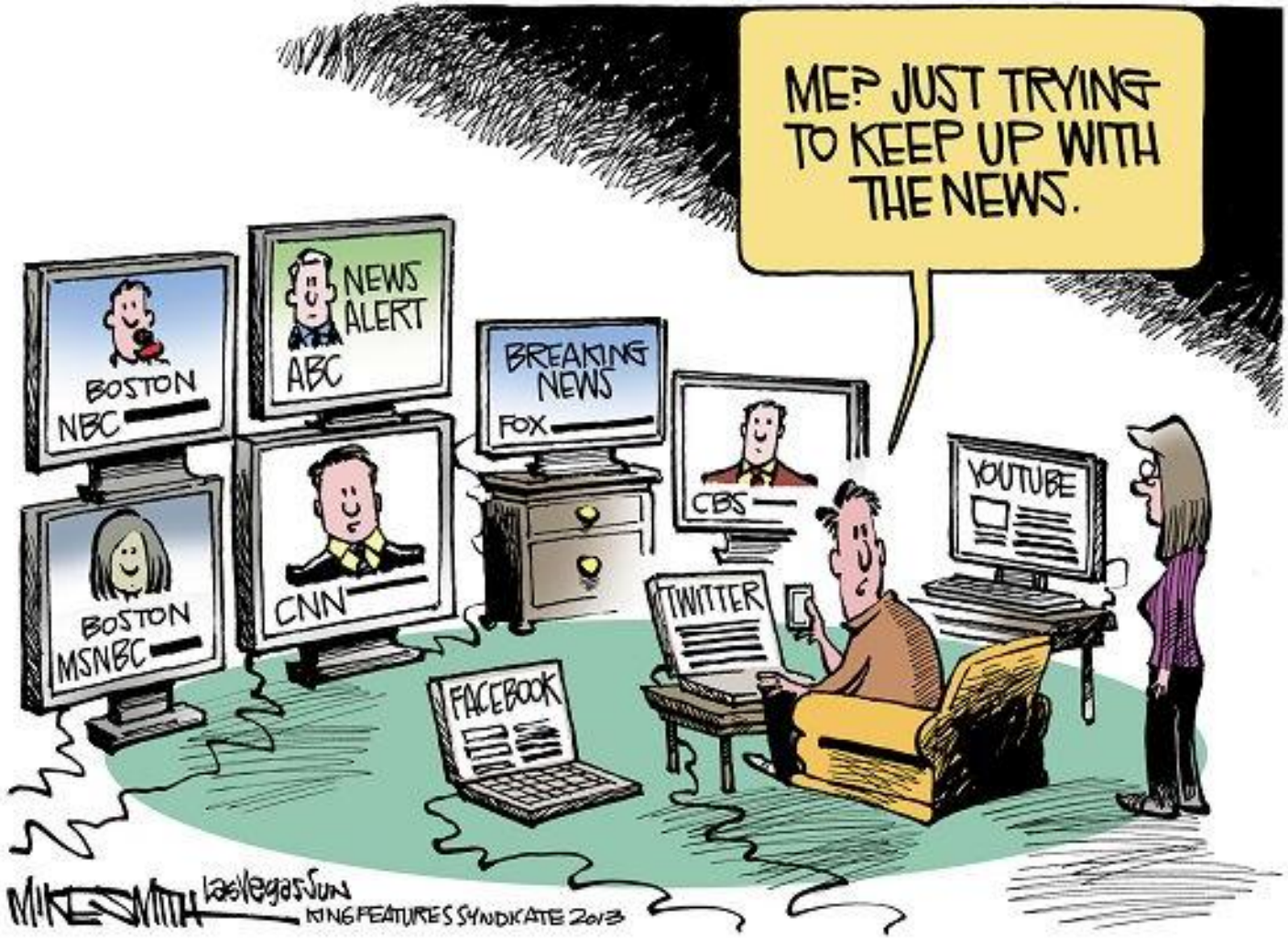
Latent Dirichlet Allocation (LDA)

Based on David Blei's KDD12 Tutorial

Information Overload



[Credits: Mike Smith, Las Vegas Sun 2013]

Topic Modeling

[Credits: IITD Library]

Topic modeling provides methods for automatically organizing, understanding, searching, and summarizing large electronic archives.

Discover the hidden themes that pervade the collection.

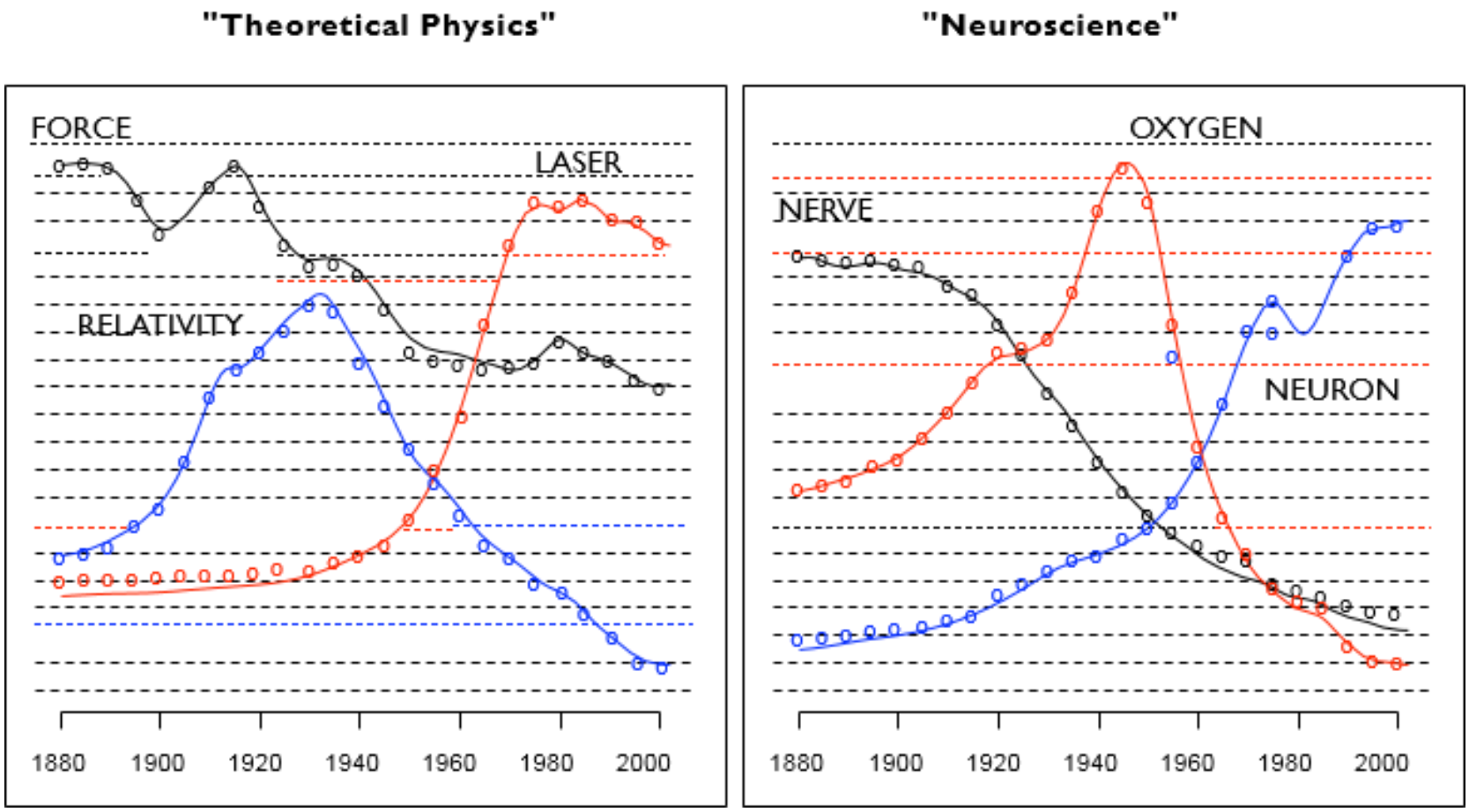
Annotate the documents according to those themes.

Use annotations to organize, summarize, and search the texts.

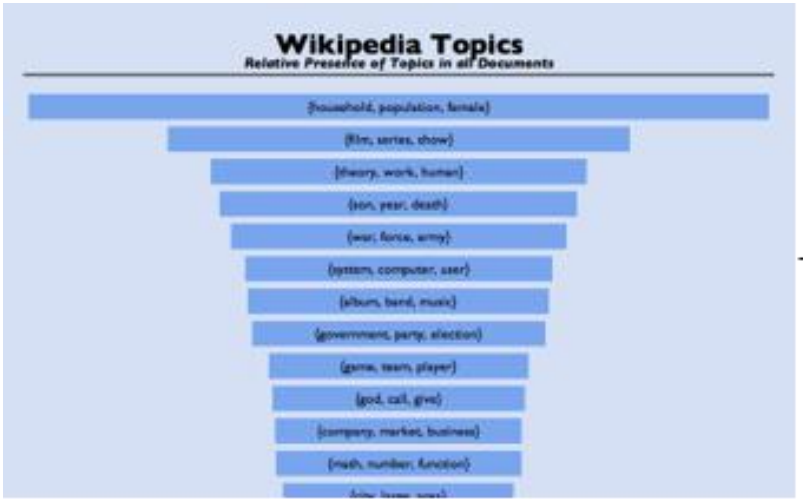
Discovering Topics

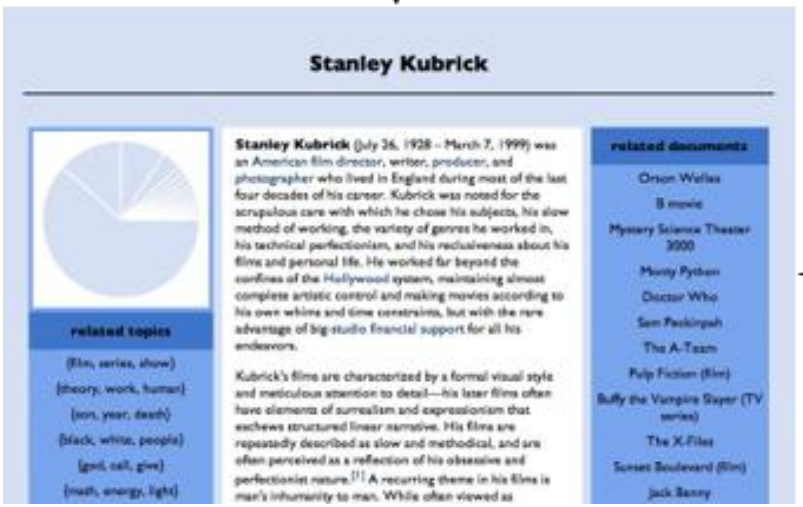
human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	systems
sequencing	phylogenetic	control	model
map	living	infectious	parallel
information	diversity	malaria	methods
genetics	group	parasite	networks
mapping	new	parasites	software
project	two	united	new
sequences	common	tuberculosis	simulations

Evolution of Topics



Organize Large Corpora

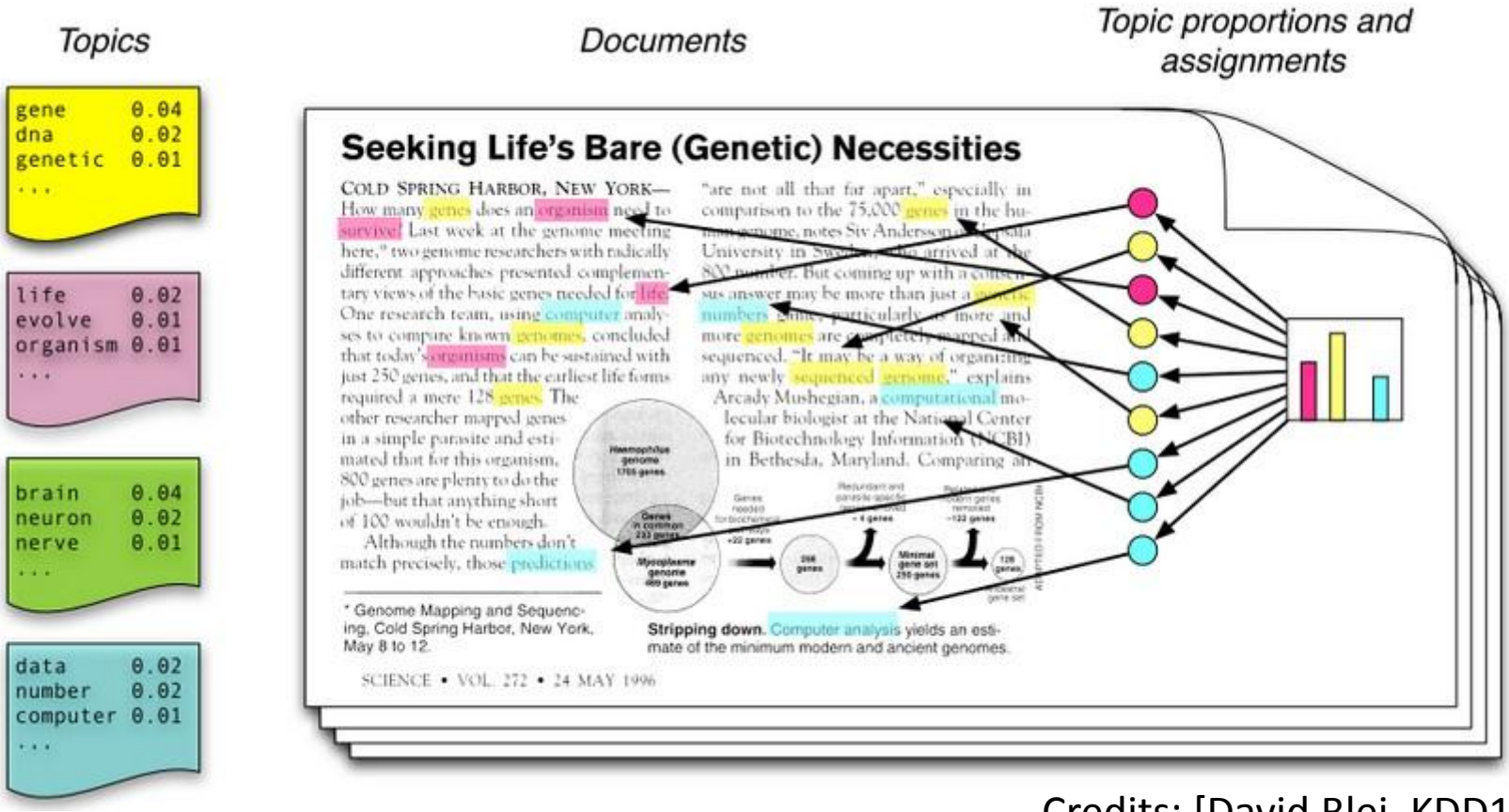




Probabilistic Modeling

- Data are assumed to be observed from a generative probabilistic process that includes hidden variables.
  - In text, the hidden variables are the thematic structure.
- Infer the hidden structure using posterior inference
  - What are the topics that describe this collection?
- Situate new data into the estimated model.
  - How does a new document fit into the topic structure?

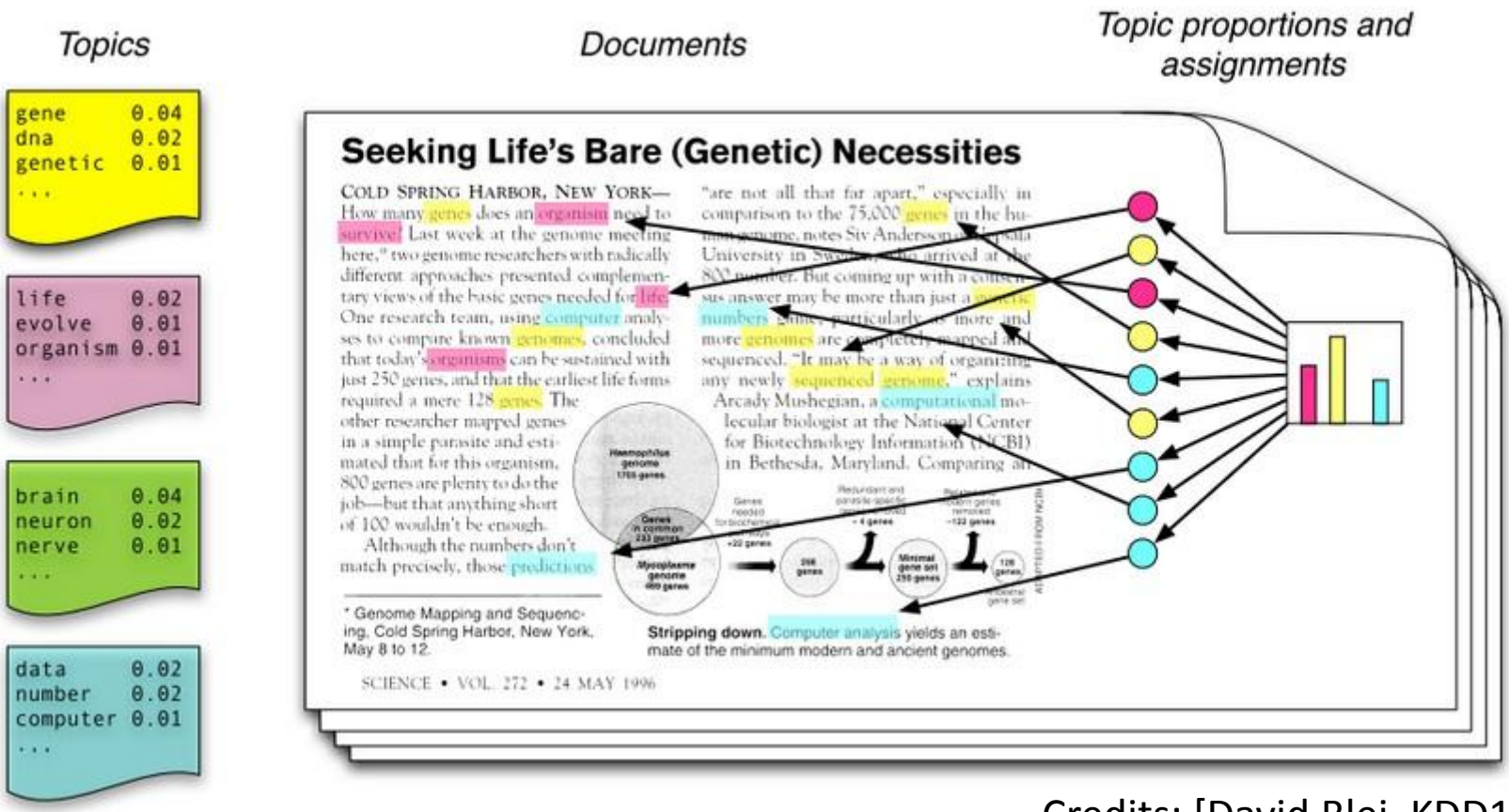
Latent Dirichlet Allocation



Credits: [David Blei, KDD12]

Intuition: Documents exhibit multiple topics

Generative Model



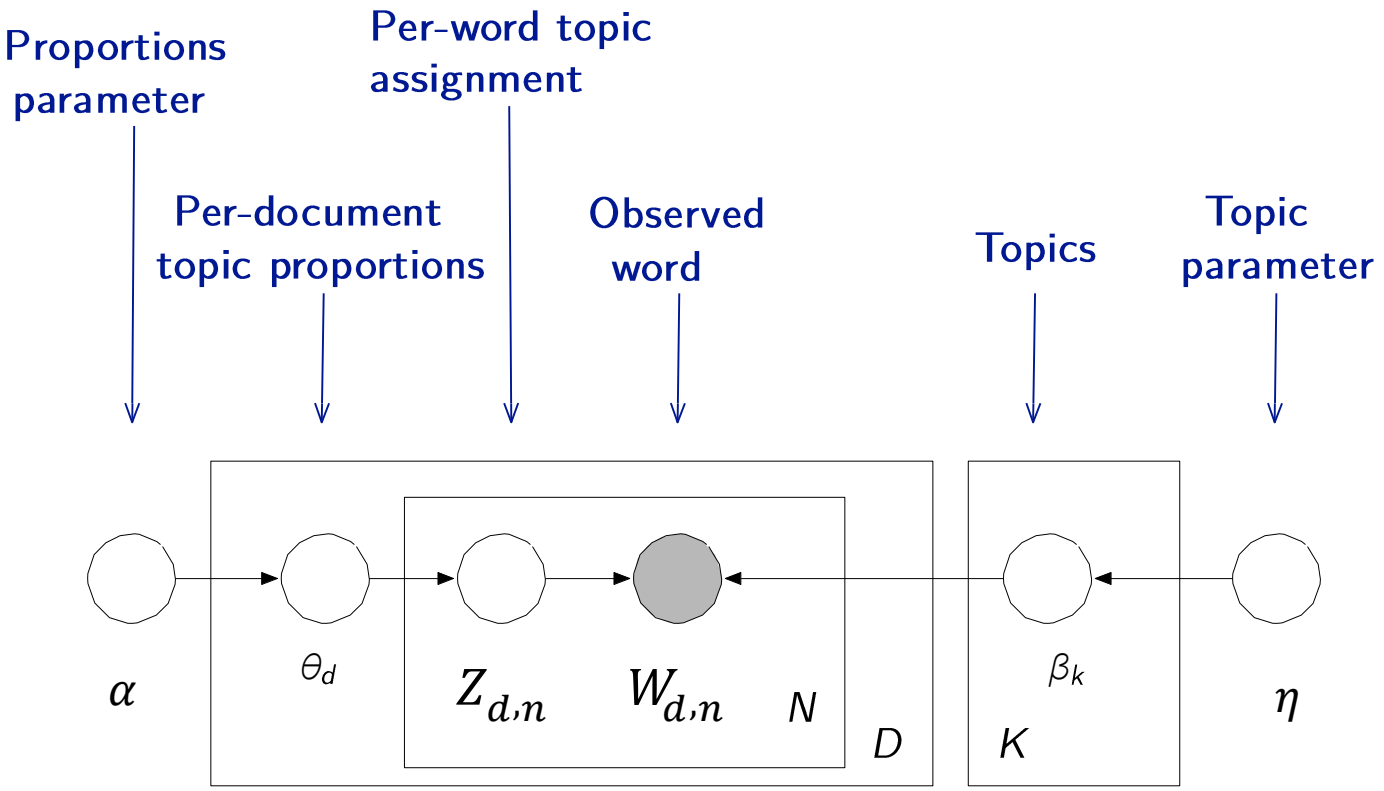
Credits: [David Blei, KDD12]

- Each **topic** is a distribution over words
- Each **document** is a mixture of corpus-wide topics
- Each **word** is drawn from one of those topics

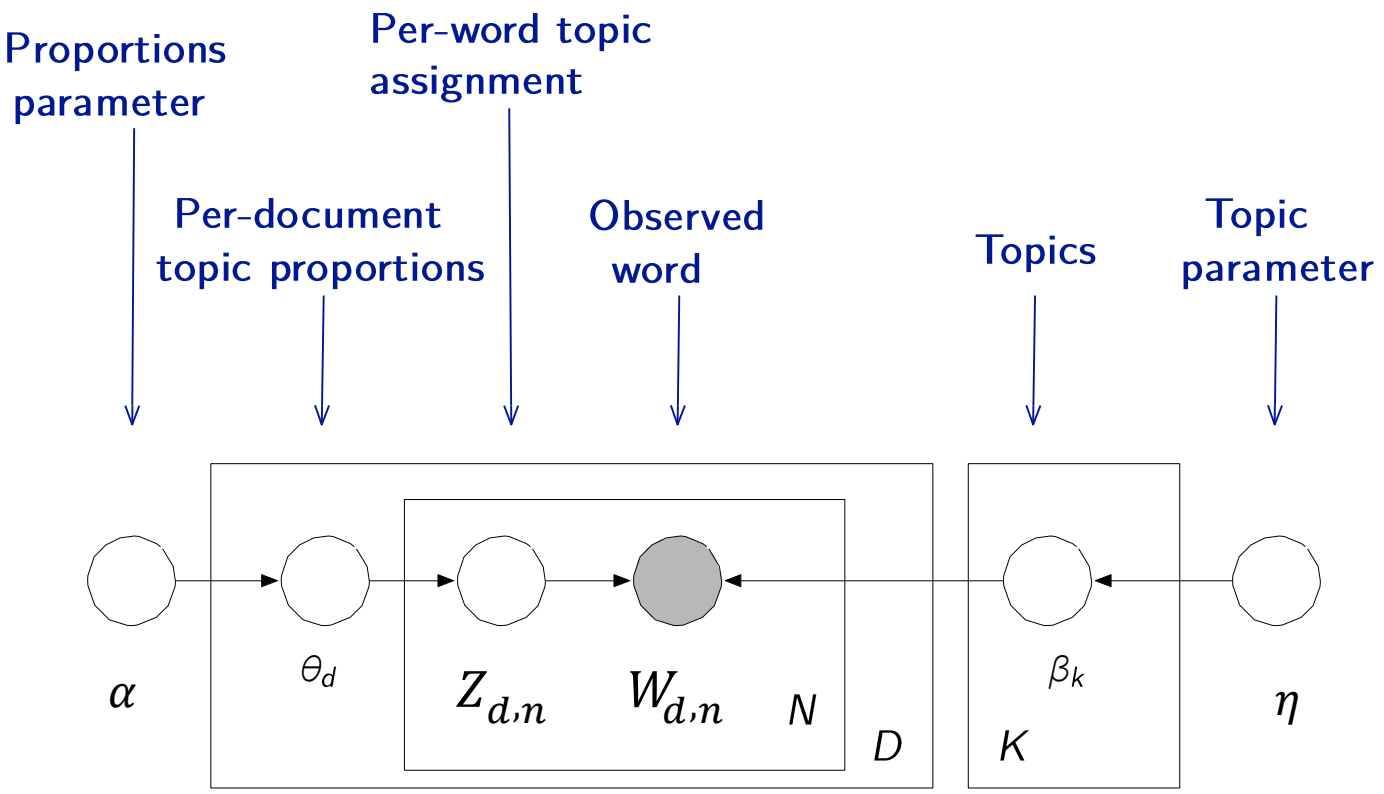
Posterior Distribution

- Only documents are observed
- Other variables are hidden
- Goal is to infer the hidden variables
  - P(topics, proportions, assignments | documents)

LDA as a PGM

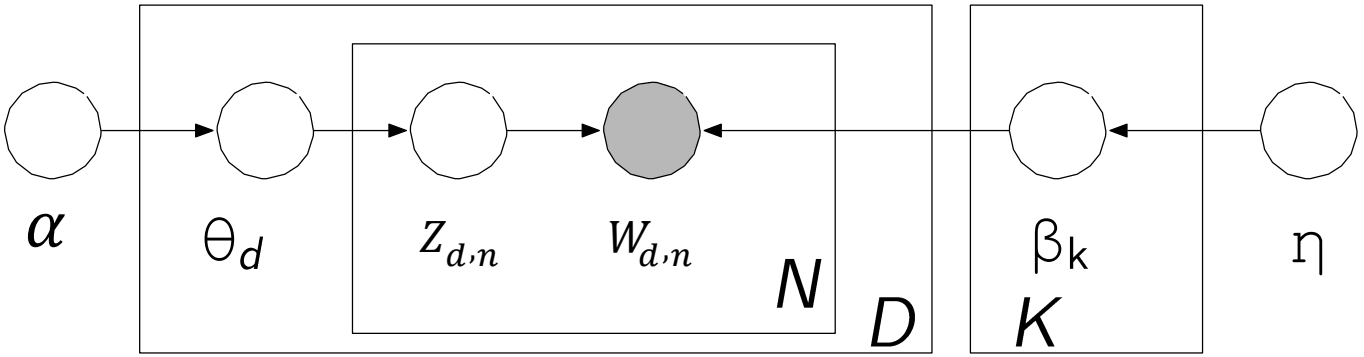

Proportions parameter
Per-word topic assignment
Per-document topic proportions
Observed word
Topics
Topic parameter
α
θ_d
$Z_{d,n}$
$W_{d,n}$
N
D
β_k
K
η

LDA as a PGM



$$\prod_{i=1}^{K} p(\beta_i \mid \eta) \prod_{d=1}^{D} p(\theta_d \mid \alpha) \left(\prod_{n=1}^{N} p(z_{d,n} \mid \theta_d) p(w_{d,n} \mid \beta_{1:K}, z_{d,n}) \right)$$

LDA



- This joint defines a posterior.
- From a collection of documents, infer
  - Per-word topic assignment $z_{d,n}$
  - Per-document topic proportions θ_d
  - Per-corpus topic distributions β_k
- Then use posterior expectations to perform the task at hand, e.g., information retrieval, document similarity, exploration

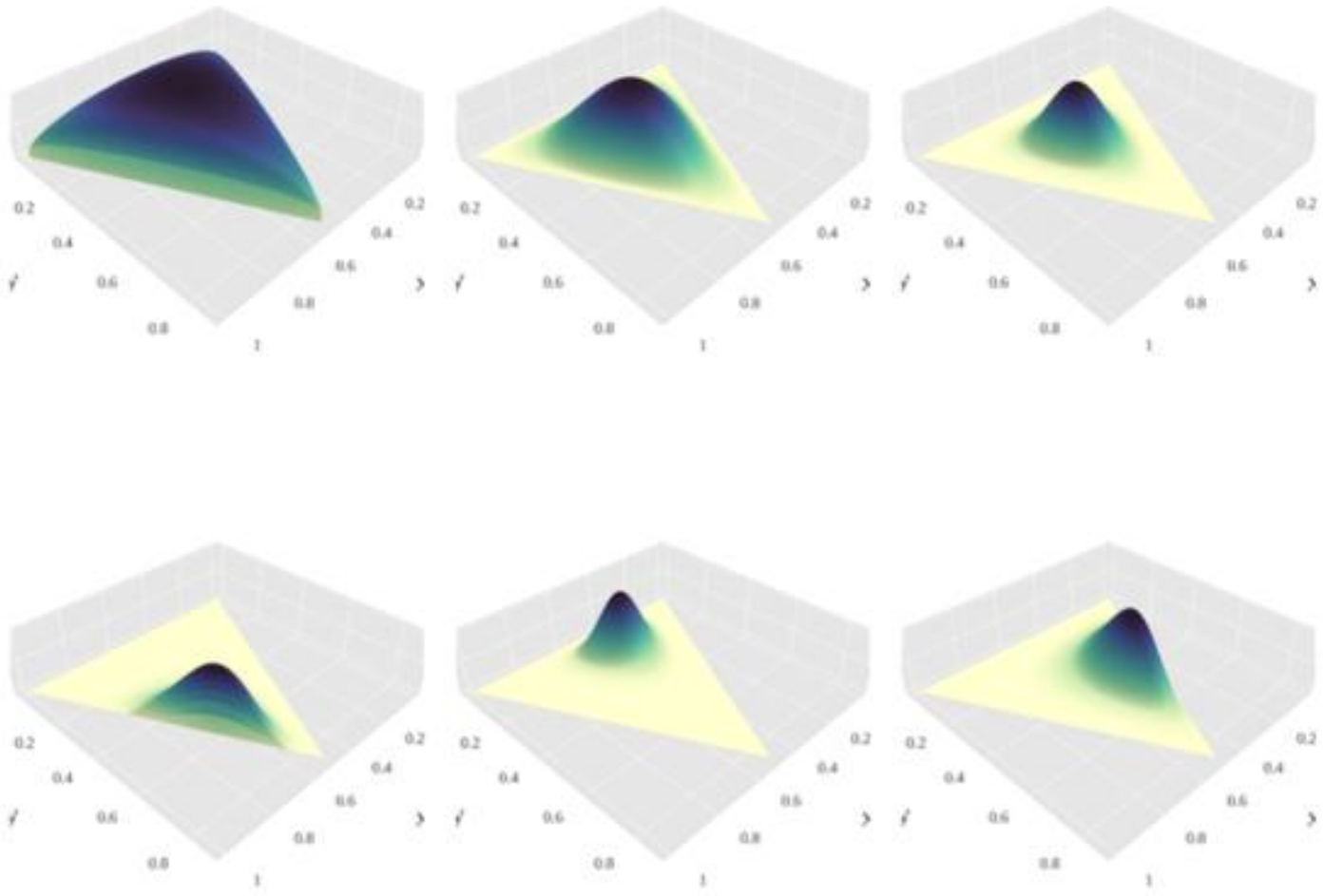
Dirichlet Distribution

- The Dirichlet distribution is an exponential family distribution over the simplex, i.e., positive vectors that sum to one
- It is **conjugate** to the multinomial. Given a multinomial observation, the posterior distribution of θ is a Dirichlet.

$$p(\theta \mid \vec{\alpha}) = \frac{\Gamma\left(\sum_i \alpha_i\right)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1}.$$

- The parameter α controls the mean shape and sparsity of θ
- The topic proportions are a K dimensional Dirichlet. The topics are a V dimensional Dirichlet

Dirichlet Distribution



[Credits: Wikipedia]

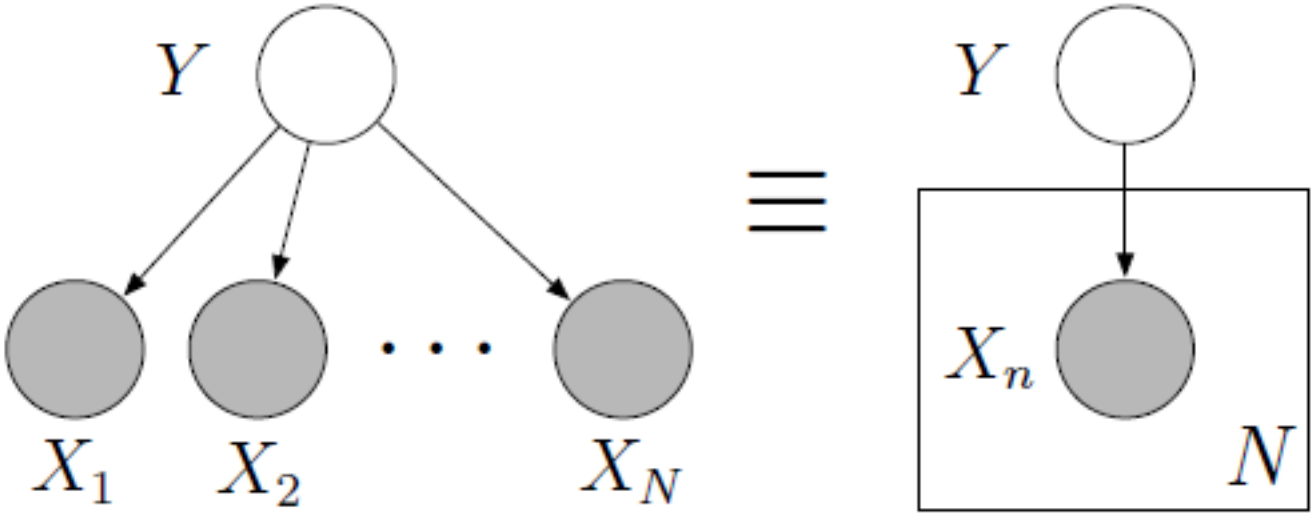
Why does LDA work?

Why does the LDA posterior put "topical" words together?

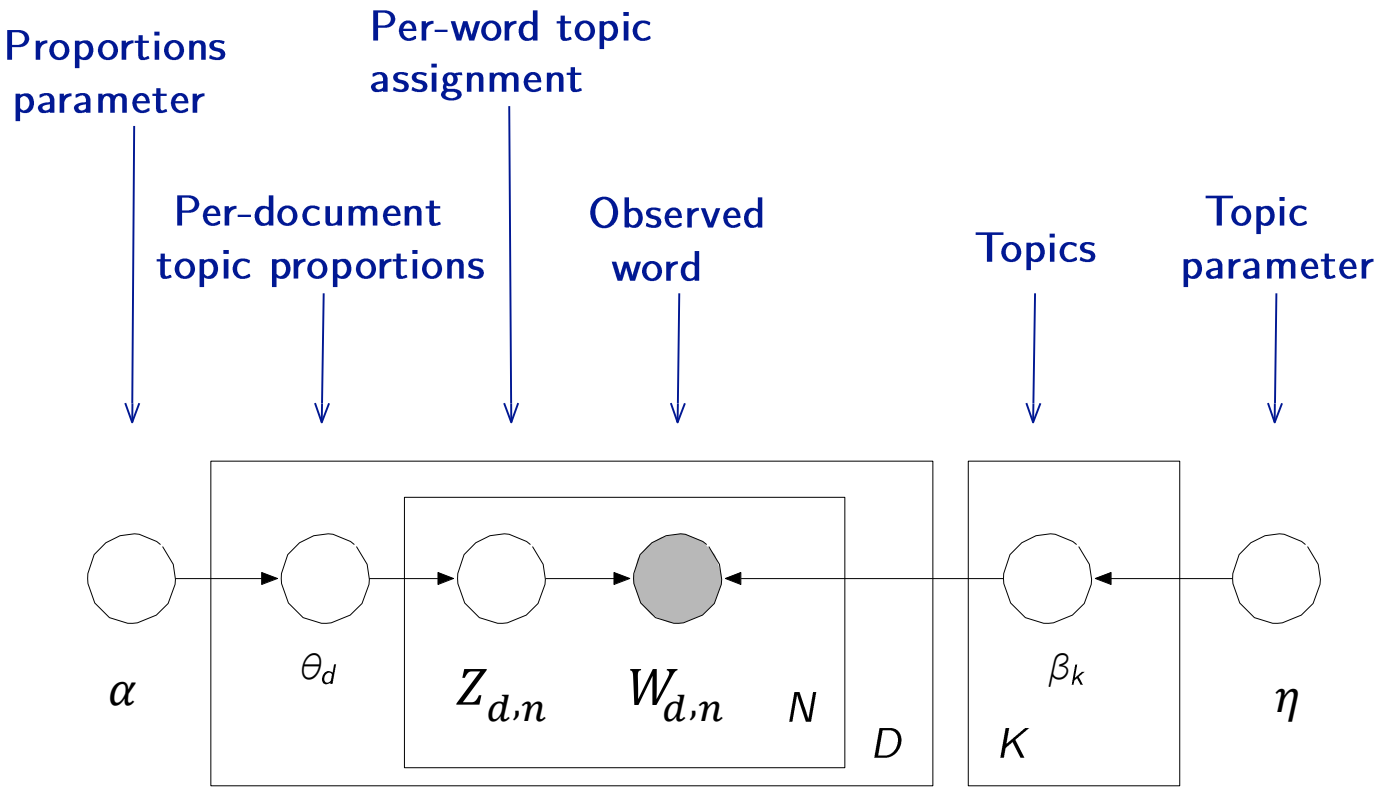
- Word probabilities are maximized by dividing the words among the topics. (More terms means more mass to be spread around.)
- In a mixture, this is enough to find clusters of co-occurring words.
- In LDA, the Dirichlet on the topic proportions can encourage sparsity, i.e., a document is penalized for using many topics.
- Loosely, this can be thought of as softening the strict definition of "co-occurrence" in a mixture model.
- This flexibility leads to sets of terms that more tightly co-occur.

Review of LDA

LDA Review



Generative model for LDA



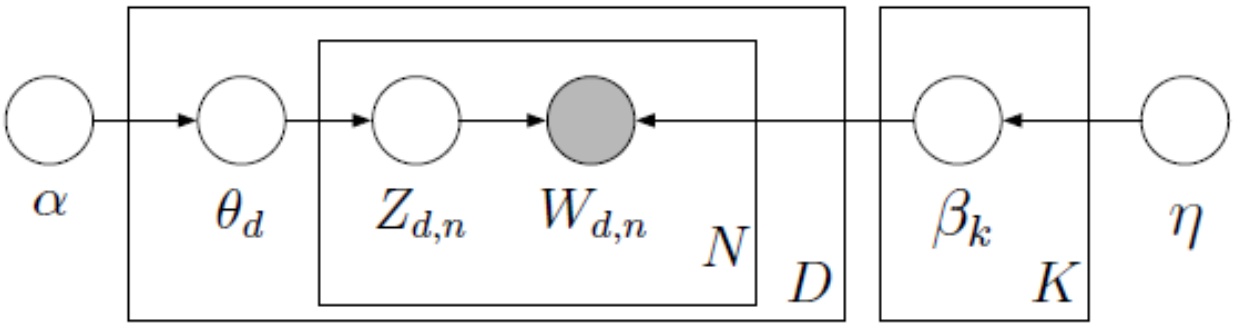
Generative Model

Topic 1: PGM (β_1)
Bayesian: 0.1
Markov: 0.09
Network: 0.07
Inference: 0.07
…

Topic 2: ML (β_2)
Inference: 0.2
Posterior: 0.15
Regression: 0.1
Gradient: 0.09
…

Topic 3: AI (β_3)
Markov: 0.09
Reinforcement: 0.08
Planning: 0.08
…

Topic 4: Deep Learning (β_4)
Backpropagation: 0.15
Convolution: 0.1
LSTM: 0.0.9
Dropout: 0.07
…



$\boldsymbol{\theta_d}$
Topic 1: 0.7
Topic 2: 0.1
Topic 3: 0.15
Topic 4: 0.05

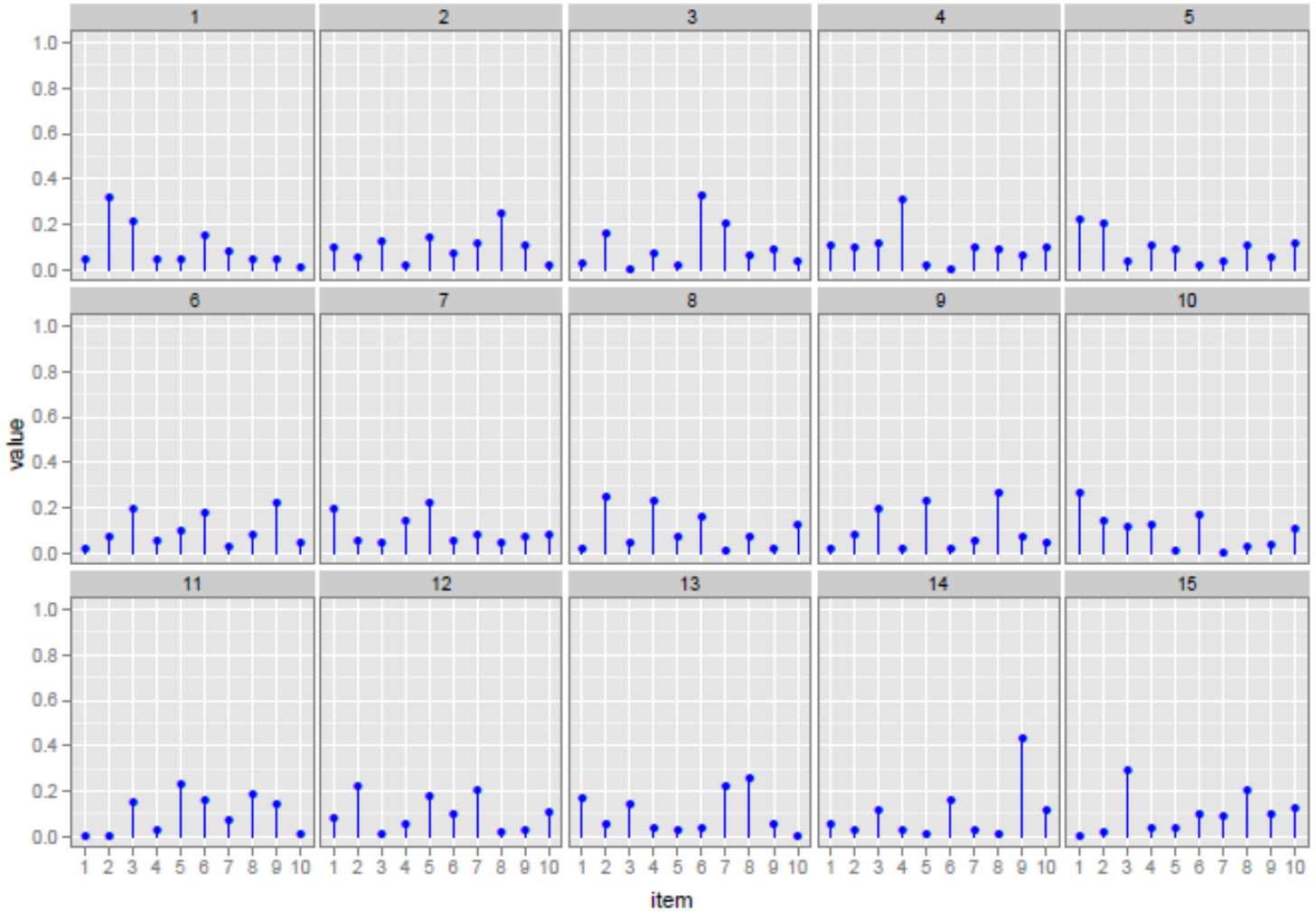
$\boldsymbol{Z_{d,n}}$
Topic 1

$W_{d,n}$
Markov

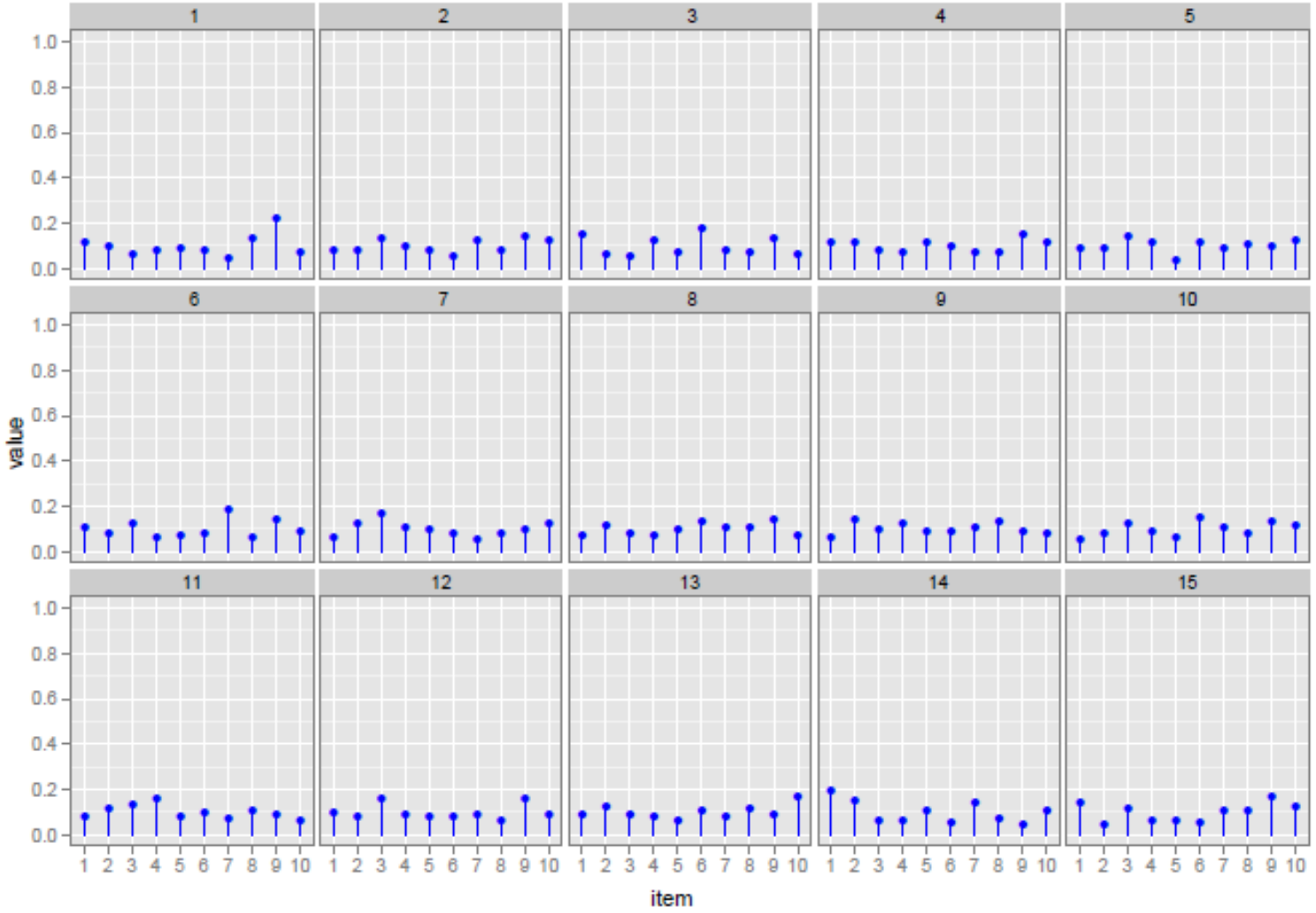
Joint Distribution

$$\prod_{i=1}^{K} p(\beta_i \mid \eta) \prod_{d=1}^{D} p(\theta_d \mid \alpha) \left(\prod_{n=1}^{N} p(z_{d,n} \mid \theta_d) p(w_{d,n} \mid \beta_{1:K}, z_{d,n}) \right)$$

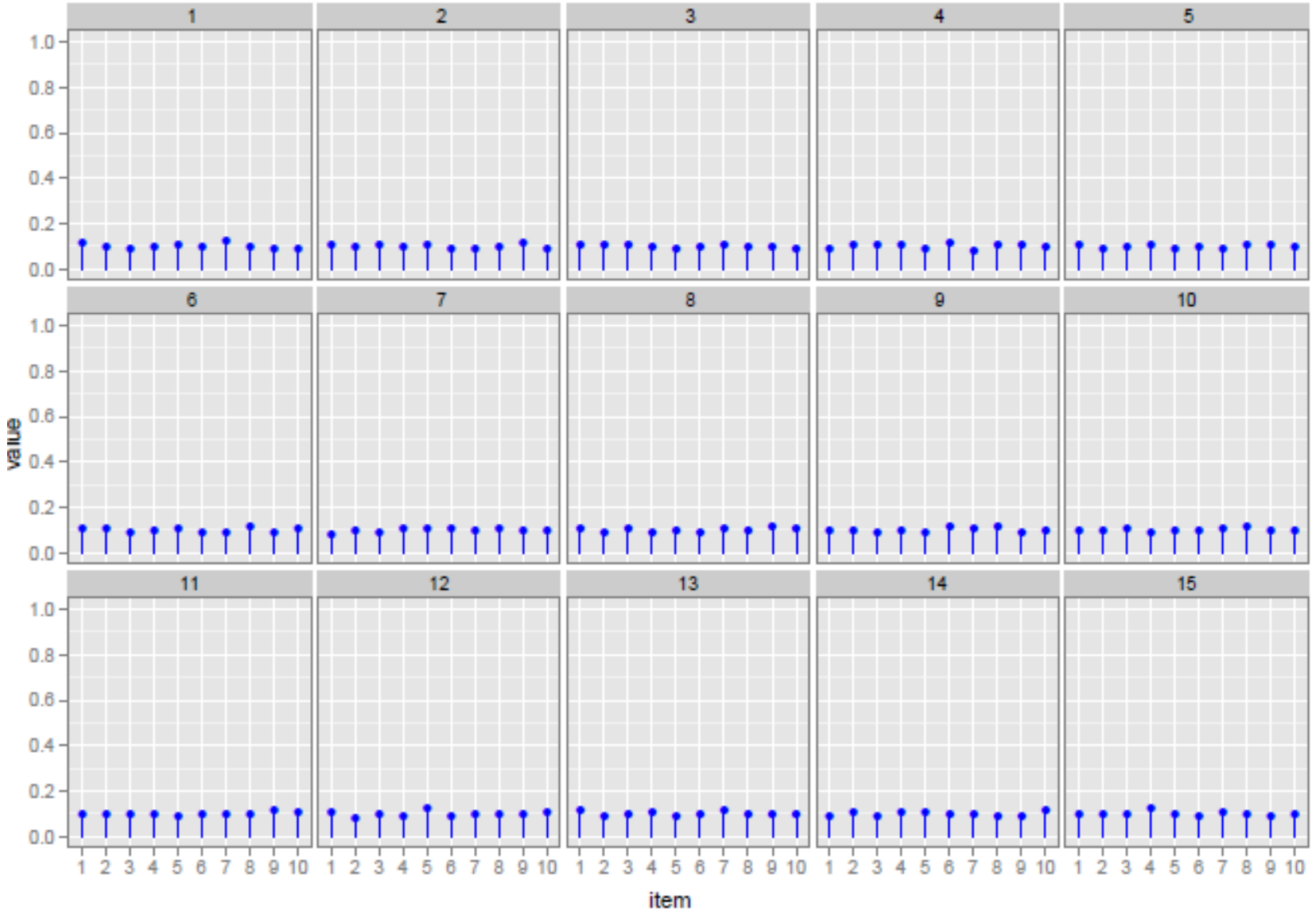
$\alpha = 1$


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
value
item
0.0
0.2
0.4
0.6
0.8
1.0
1 2 3 4 5 6 7 8 9 10

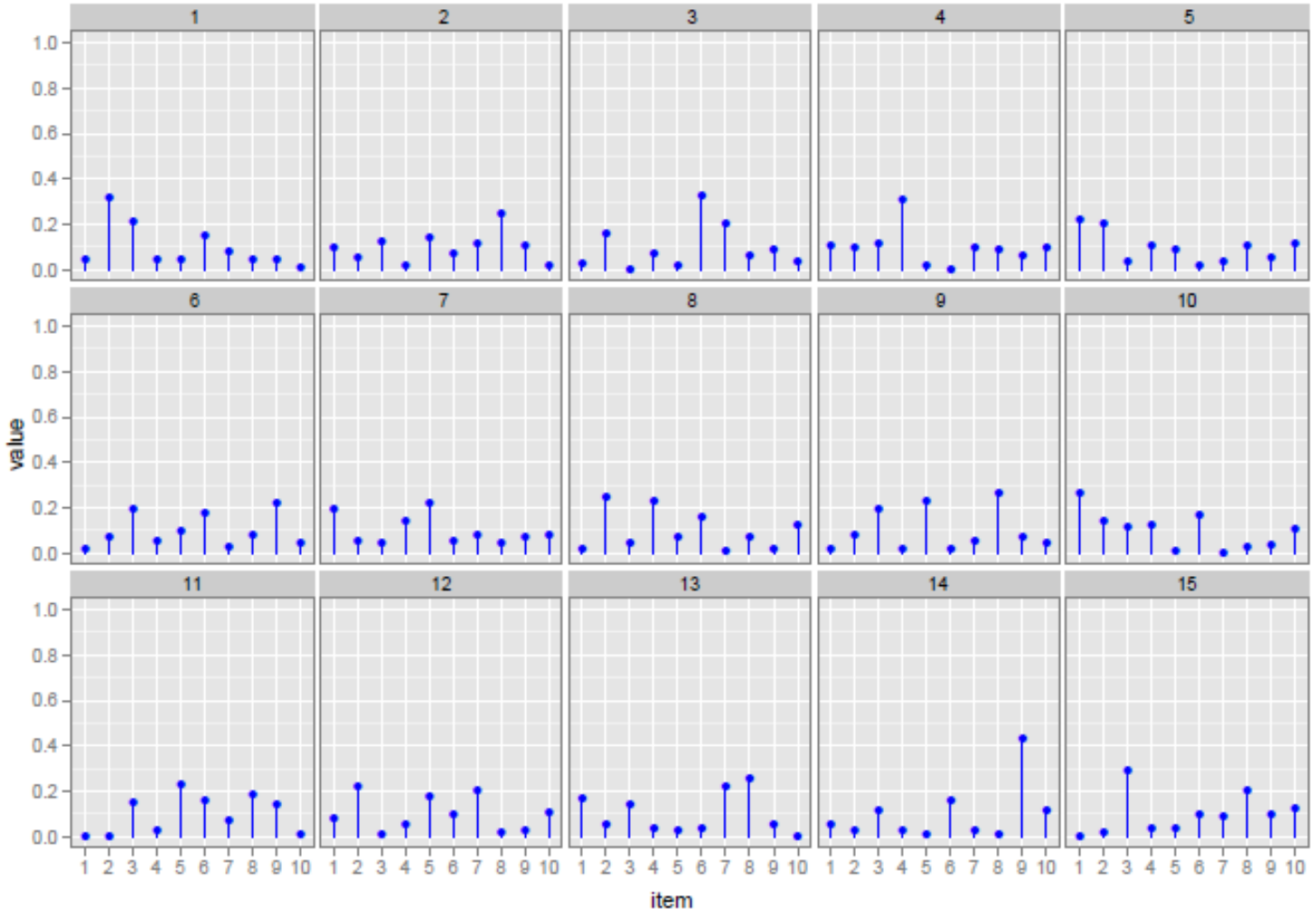
$\alpha = 10$



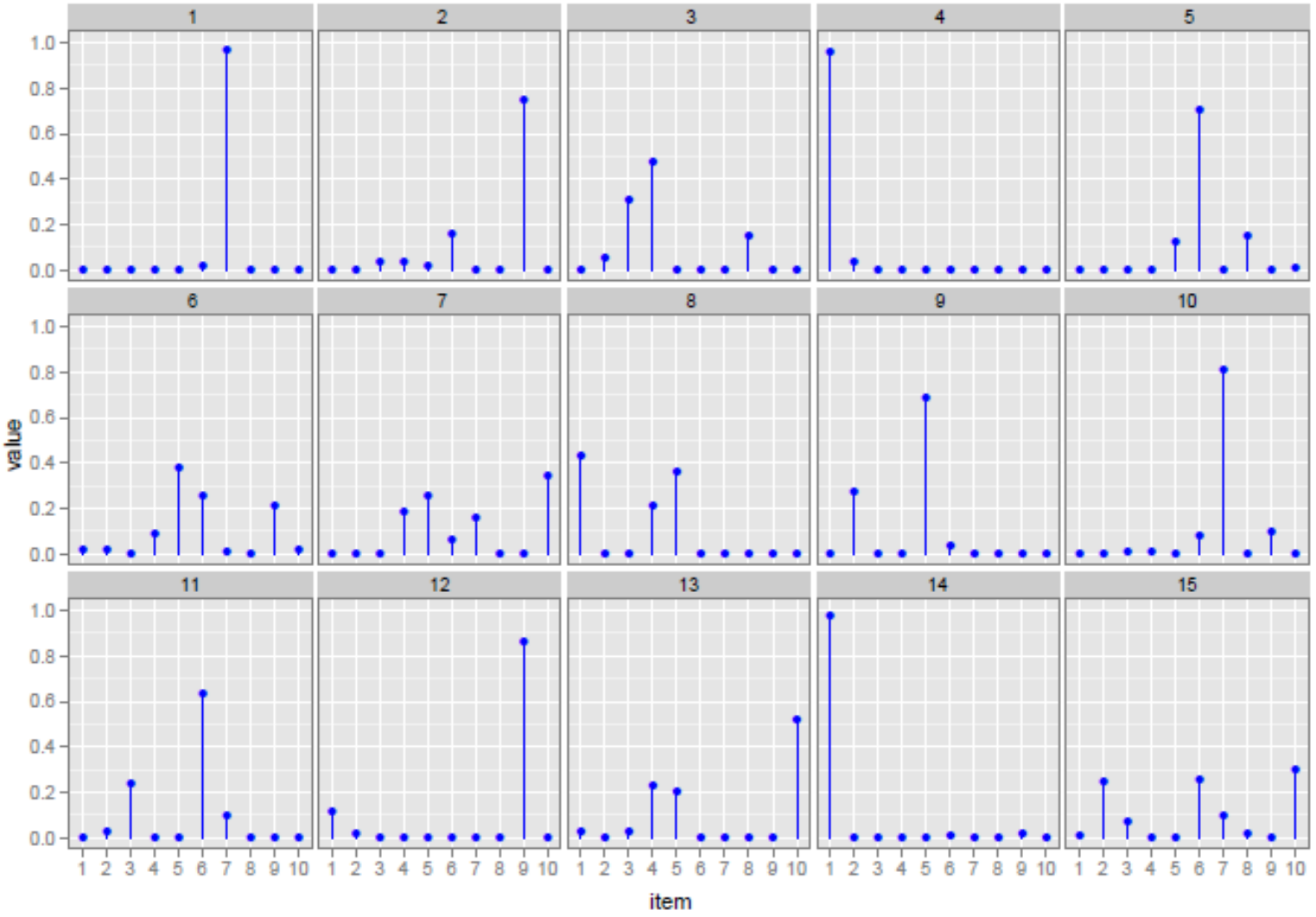
$\alpha = 100$


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
value
item
0.0
0.2
0.4
0.6
0.8
1.0
1 2 3 4 5 6 7 8 9 10

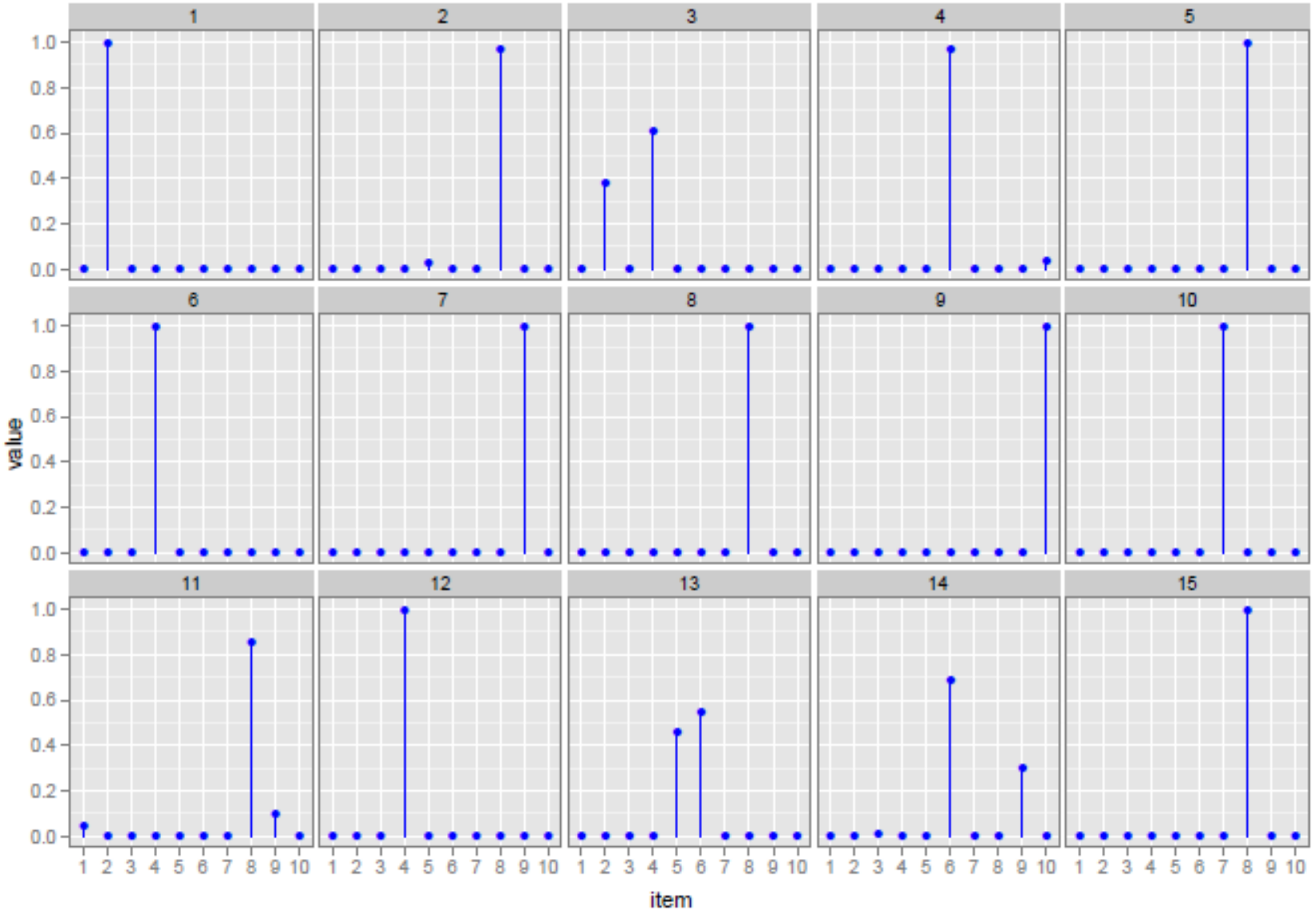
$\alpha = 1$


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
value
item
0.0
0.2
0.4
0.6
0.8
1.0
1 2 3 4 5 6 7 8 9 10

$\alpha = 0.1$



$\alpha = 0.01$


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
value
item
0.0
0.2
0.4
0.6
0.8
1.0
1 2 3 4 5 6 7 8 9 10

Posterior Inference

Joint Distribution:

$$\prod_{i=1}^{K} p(\beta_i \mid \eta) \prod_{d=1}^{D} p(\theta_d \mid \alpha) \left(\prod_{n=1}^{N} p(z_{d,n} \mid \theta_d) p(w_{d,n} \mid \beta_{1:K}, z_{d,n}) \right)$$

Posterior of Latent Variables:

$$p(\beta, \theta, z \mid w)$$

This is equal to:

$$\frac{p(\beta, \theta, z, w)}{\iint_{\beta,\theta} \sum_z p(\beta, \theta, z, w)}$$

The denominator can't be computed easily

Variational Inference

- Alternative to MCMC
- Converts posterior inference into optimization
- Let $x_{1:N}$ be observations and $z_{1:M}$ be latent variables
- Goal: Compute

$$p(z_{1:M} \mid x_{1:N}) = \frac{p(z_{1:M}, x_{1:N})}{\int p(z_{1:M}, x_{1:N}) dz_{1:M}}$$

Variational Inference

- Use Jensen's inequality to bound the log prob of observations (Evidence Lower Bound (ELBO))

$$
\begin{aligned}
\log p(x) &= \log \int p(\beta, Z, x)\, dZ d\beta \\
&= \log \int p(\beta, Z, x) \frac{q(\beta, Z)}{q(\beta, Z)}\, dZ d\beta \\
&\geq \int q(\beta, Z) \log \frac{p(\beta, Z, x)}{q(\beta, Z)}\, dZ d\beta \\
&= \mathrm{E}_q[\log p(\beta, Z, x)] - \mathrm{E}_q[\log q(\beta, Z)].
\end{aligned}
$$

- Introduced a distribution with free variational parameter ν.
- Optimize those parameters to tighten this bound

Mean Field Variational Inference

- Assume $q(\beta, z)$ to be fully factorized:

$$q(\beta, z) = q(\beta \mid \lambda) \prod_{i=1}^{n} q(z_i \mid \phi_i).$$

- Assume each complete conditional is exponential family:

$$\begin{aligned} p(z_i \mid \beta, x_i) &= h(z_i) \exp\{\eta_\ell(\beta, x_i)^\top z_i - a(\eta_\ell(\beta, x_i))\} \\ p(\beta \mid z, x) &= h(\beta) \exp\{\eta_g(z, x)^\top \beta - a(\eta_g(z, x))\}. \end{aligned}$$

- Each component is in the same family as model conditional:

$$q(\beta \mid \lambda) = h(\beta) \exp\{\lambda^\top \beta - a(\lambda)\}$$

- Optimize ELBO w.r.t parameters:

$$\mathcal{L}(\lambda, \phi_{1:n}) = \mathrm{E}_q[\log p(\beta, Z, x)] - \mathrm{E}_q[\log q(\beta, Z)].$$

- Coordinate Ascent: Iteratively update each parameter holding others fixed
- Gradient is

$$\nabla_\lambda \mathcal{L} = a''(\lambda)(\mathrm{E}_\phi[\eta_g(Z, x)] - \lambda).$$

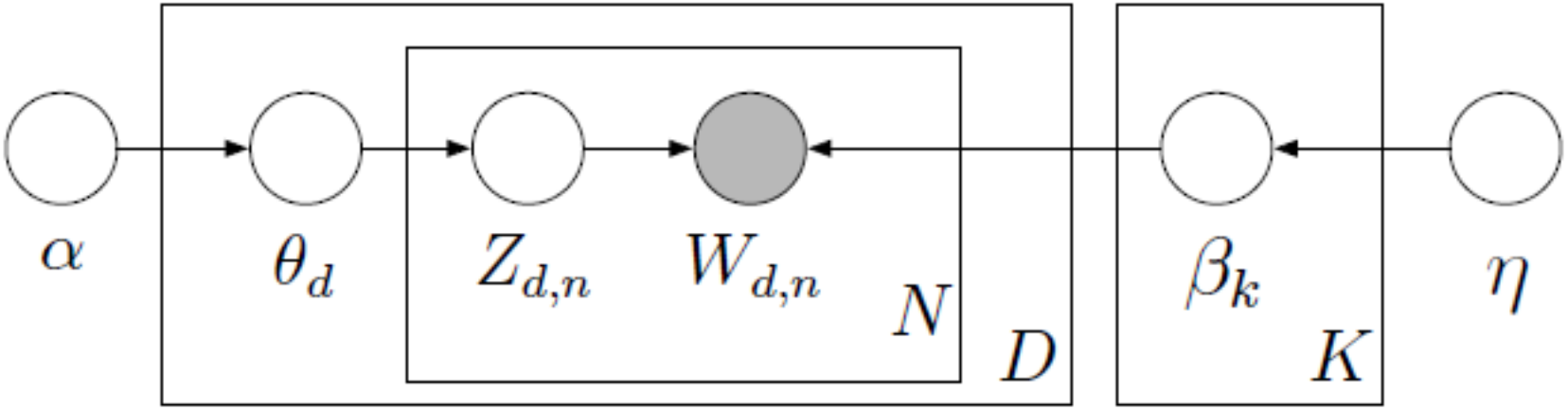
- Coordinate update:

$$\lambda^* = \mathrm{E}_\phi\left[\eta_g(Z, x)\right].$$

Gibbs Sampling

- Define a Markov chain whose stationary distribution is the posterior of interest
- Collect independent samples from that stationary distribution; approximate the posterior with them
- In Gibbs sampling, the space of the MC is the space of possible configurations of the hidden variables.
- The chain is run by iteratively sampling from the conditional distribution of each hidden variable given observations and the current state of the other hidden variables

Gibbs sampling



- Define $n(z_{1:N})$ to be the counts vector
- A simple Gibbs sampler is:

Where

$$\theta \mid w_{1:N}, z_{1:N} \sim \mathrm{Dir}(\alpha + n(z_{1:N}))$$
$$z_i \mid z_{-i}, w_{1:N} \sim \mathrm{Mult}(\pi(z_{-i}, w_i))$$
$$\pi(z_{-i}, w_i) \propto (\alpha + n(z_{1:N}))p(w_i \mid \beta_{1:K})$$

Collapsed Gibbs Sampling

- Topic proportions θ can be integrated out
- A collapsed Gibbs sampler draws from:

$$p(z_i \mid z_{-i}, w_{1:N}) \propto p(w_i \mid \beta_{1:K}) \prod_{k=1}^{K} \Gamma(n_k(z_{-i})),$$

where $n_k(z_{-i})$ is the number of times we've seen topic k in the collection of topic assignments z_{-i}

Integrating out leads to faster mixing in chain

Open Domain Event Extraction from Twitter

Based on Alan Ritter's Slides

Goal: Realtime Stream of Structured Information About Events

Q: which set of events do we want to know about?

TYPE:
Launch
DATE:
Mar 7

Steve Jobs
TYPE:
Death
DATE:
Oct 6

Yelp
TYPE:
IPO
DATE:
March 2

Q: How soon can we know about an event?

TIME

Data for events?

- **Structured / Semi-Structured data sources?**
  - Facebook / Eventbrite
  - HTML / Wrapper induction
- **Natural Language / Text?**
  - News articles
  - Status Messages / Twitter

Claim: This is worth investigating

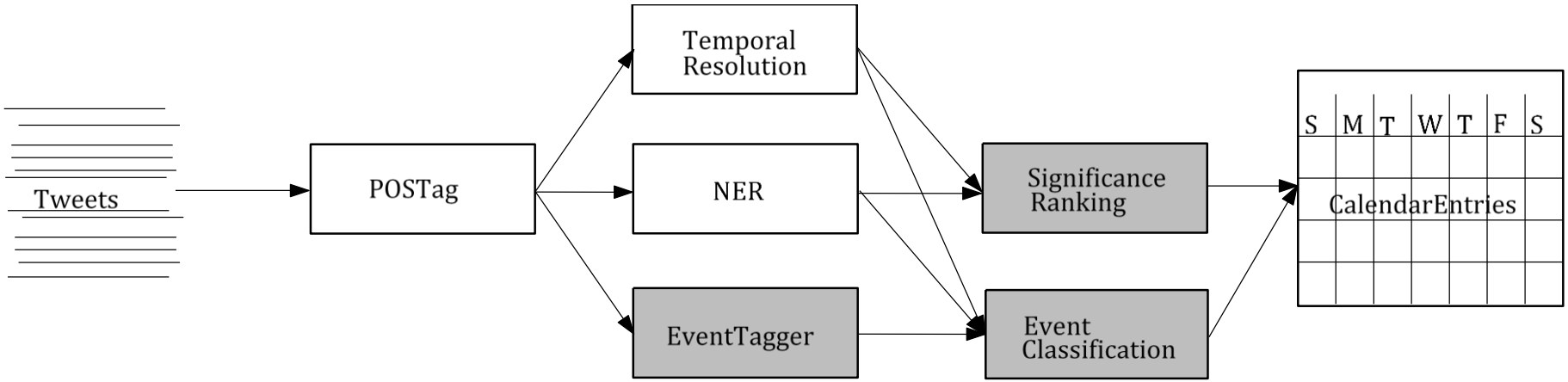
Calendar Demo

- Extract Named Entities
  - **1%** sample of global Twitter stream
  - 2.5 Million / Day
  - Using NER trained on Labeled Tweets
    - [Ritter et. al. EMNLP 2011]
- Extract and Resolve Temporal Expressions
  - For example "Next Tuesday" = **21-02-17**
- Count Entity/Day co-occurrences
  - G^2 Log Likelihood Ratio
- Plot Top K Entities for Each Day

[Prachi] Events shown as url

http://statuscalendar.com

Pipeline



[Nupur] Model Architecture
[Happy] Normalization?
[Shantanu, Surag] Error Accumulation
[Himanshu, Prachi] Reliance on POS tagger

Background: Event Extraction from News

- Historically, the most important source of info on current events
  - Since spread of printing press
- Lots of previous work on Newswire
  - Timebank
  - MUC & ACE competitions
    - Limited to narrow domains
    - Performance is still not great

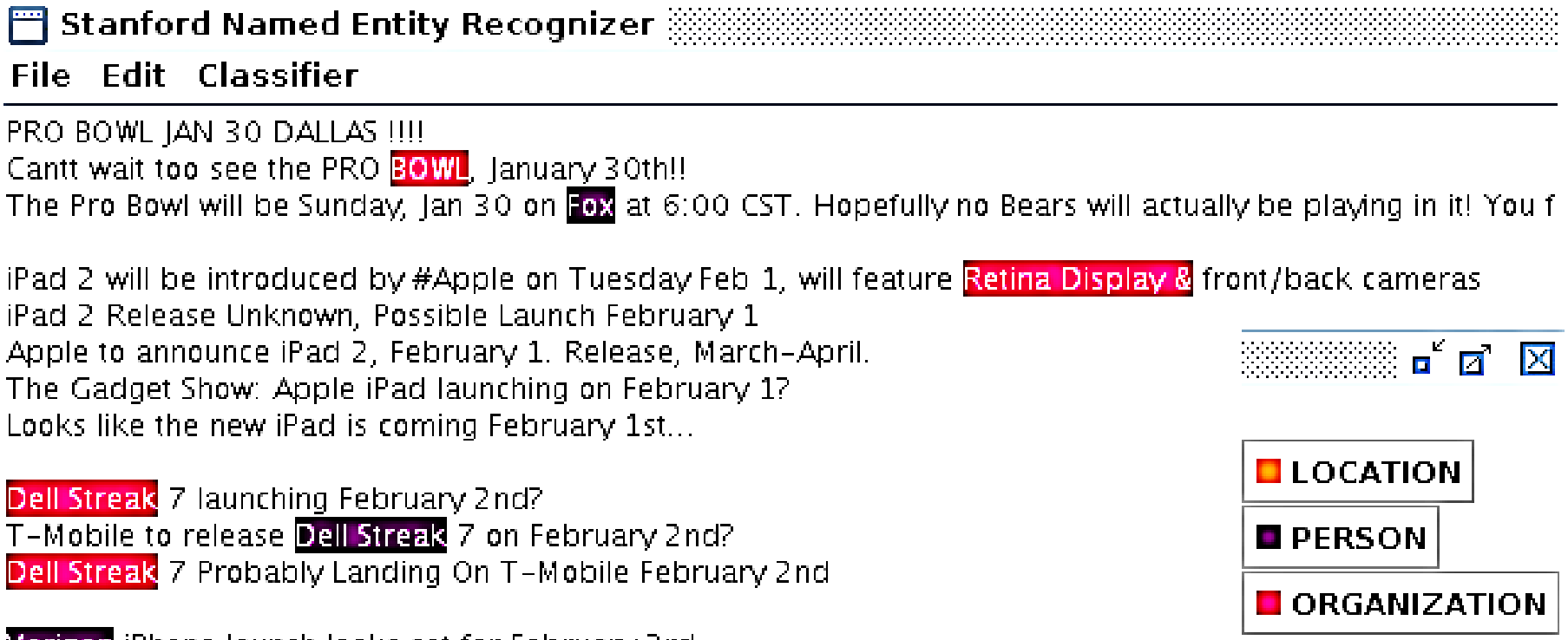
Social Media

- Competing source of info on current events
- Status Messages
  - Short
  - Easy to write (even on mobile devices)
  - Instantly and widely disseminated
- Double Edged Sword
  - Many irrelevant messages
  - Many redundant messages

Noisy Text: Challenges

- Lexical Variation (misspellings, abbreviations)
  - `2m', `2ma', `2mar', `2mara', `2maro', `2marrow', `2mor', `2mora', `2moro', `2morow', `2morr', `2morro', `2morrow', `2moz', `2mr', `2mro', `2mrrw', `2mrw', `2mw', `tmmrw', `tmo', `tmoro', `tmorrow', `tmoz', `tmr', `tmro', `tmrow', `tmrrow', `tmrrw', `tmrw', `tmrww', `tmw', `tomaro', `tomarow', `tomarro', `tomarrow', `tomm', `tommarow', `tommarrow', `tommoro', `tommorow', `tommorrow', `tommorw', `tommrow', `tomo', `tomolo', `tomoro', `tomorow', `tomorro', `tomorrw', `tomoz', `tomrw', `tomz‘
- Unreliable Capitalization
  - “The Hobbit has FINALLY started filming! I cannot wait!”
- Unique Grammar
  - “watchng american dad.”

Off The Shelf NLP Tools Fail



Annotating Named Entities

RT @daviddesrosiers : Happy birthday @chuckcomeau ! have fun in [vancouver]$_{loc}$ tonight !

Spotted : [Kanye West]$_{per}$ Celebrates [LAMB]$_{company}$ With [Gwen Stefani]$_{per}$: [New York Fashion Week]$_{event}$ is coming to a close , but not before ... http://bit.ly/cSyZUi

@zeeDOTi i might join in if I make it home in time . :)

@SinfulSalvatore { drinks slowly from my cup observing your actions before clearing my throat } is -this- your plans for today ?

If you are in [Belfast]$_{loc}$ this weekend go to [Botanic Gdns]$_{loc}$, for Taste N I , free food drink face painting etc .

- Annotated 2400 tweets (about 34K tokens)
- Train on in-domain data

My teacher turned on the radio in class and all I heard was @justinbieber U smile I smile that made my day :D

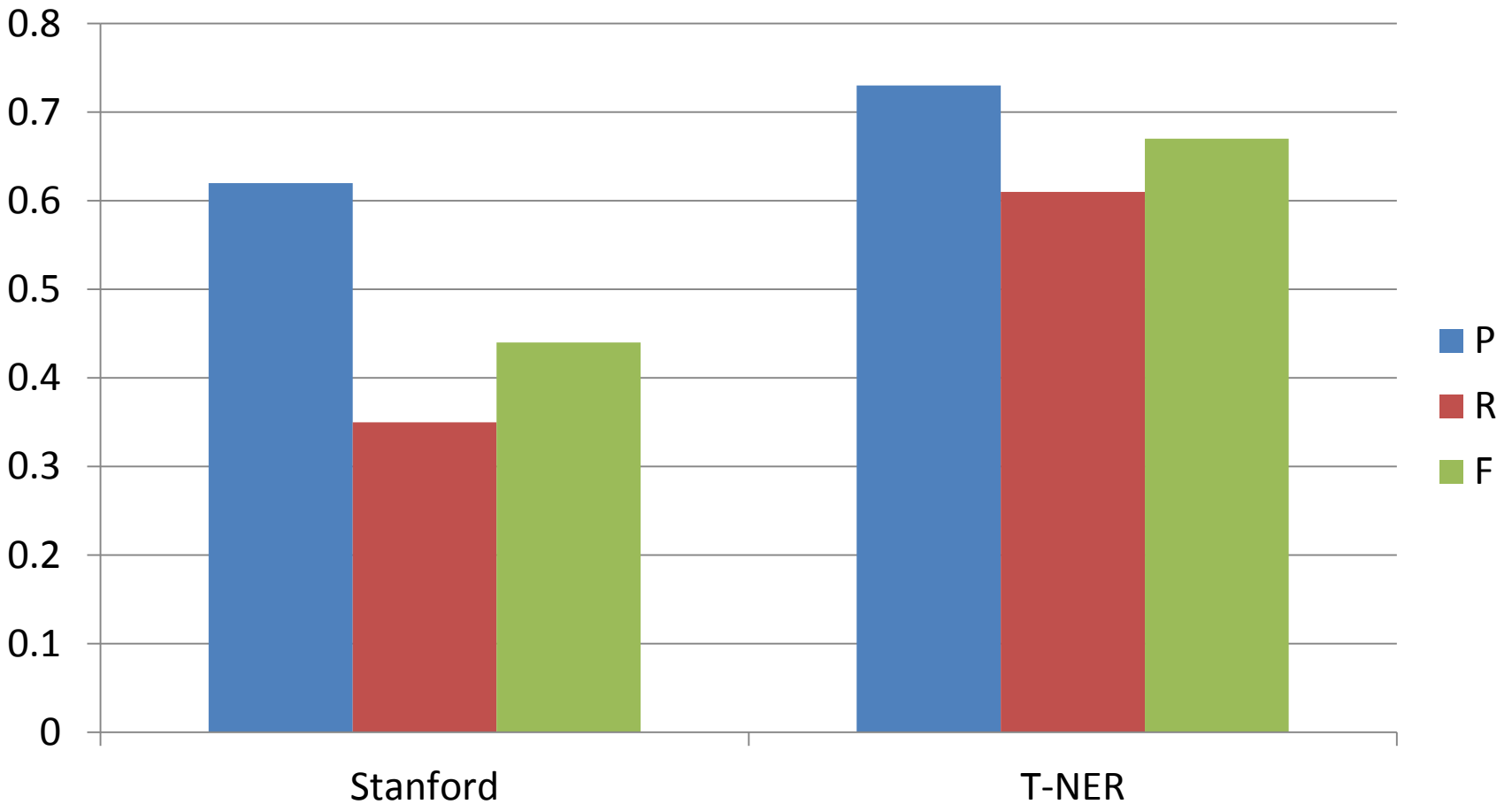
RT @CantBeliebIt : When [Justin]$_{per}$ gets a girlfriend That Should Be Me is gonna be #1 on [iTunes]$_{internetwebsite}$ '

When the last time you ran into [Rick Ross]$_{per}$ and [Drake]$_{per}$ twice in the same day at 2 diff video shoot locations . Today I did !

@atlgc17 Im 16 and a freshmen again . I fluncked once in elementry school and I fluncked last year because I never did my work .

@Cuddle_Factory will send again tomorrow honey too

Performance (NE Segmentation)



Event-Referring Phrases

- Useful to display in connection with events
  - E.g. "**Steve Jobs**" + + "**October 6**"
- Helpful in categorizing Events into Types

Examples:

Apple to **Announce** iPhone 5 on October 4th! YES!

iPhone 5 **announcement** coming Oct 4th

WOOOHOO **NEW** IPHONE TODAY! CAN'T WAIT!

Event Phrases: Annotation/Learning

- Annotated 1,000 tweets (19,484 tokens)
- Similar to EVENT tags in TimeBank
- Sequence-labeling problem
  - IOB Encoding
  - Conditional Random Fields

Event Phrases: Features

- Contextual features
  - POS tags
  - Adjacent words
- Dictionary Features
  - Event words gathered from WordNet
  - Brown Clusters
- Orthographic Features
  - Prefixes, suffixes

Event Segmentation Results

	precision	recall	F1
T-EVENT	0.56	0.74	0.64
No POS	0.48	0.70	0.57
TimeBank	0.24	0.11	0.15

Event Representation

Entity	Event Phrase	Date
Steve Jobs	died	10/6/11
iPhone	announcement	10/4/11
GOP	debate	9/7/11
...	...	...

Categorizing Event Types

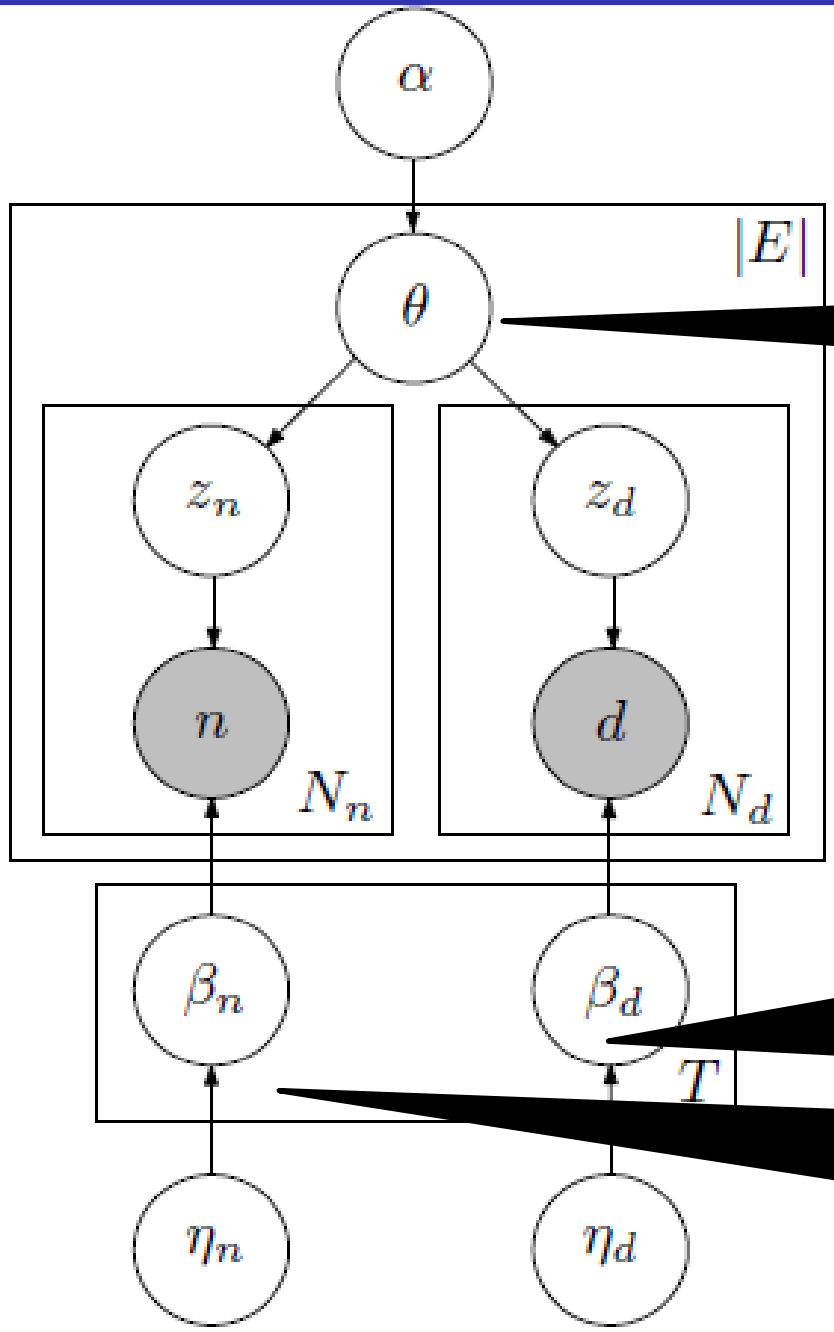
- Would like to categorize events into types, for example:
  - Sports
  - Politics
  - Product releases
  - ...
- Benefits:
  - Allow more customized calendars
  - Could be useful in upstream tasks

Classifying Events: Challenges

- Many Different Types
- Not sure what is the right set of types
- Set of types might change
  - Might start talking about different things
  - Might want to focus on different groups of users

Unsupervised Event Type Induction

- Latent Variable Models
  - Generative Probabilistic Models
- Advantages:
  - Discovers types which **match the data**
  - No need to annotate individual events
  - Don’t need to commit to a specific set of types
  - Modular, can integrate into various applications



Each **Event Phrase** is modeled as a mixture of types

[Happy, Arindam, Akshay, Surag, Dinesh R] Liked
[Akshay] New entities?
[Anshul] Sensitive to parameters

P(**SPORTS**|**cheered**)= 0.6
P(**POLITICS**|**cheered**)= 0.4

Each **Event Type** is Associated with a Distribution over Entities and Dates

Details...

- Gathered about 65M (entity, event, date) tuples
- Collapsed Gibbs Sampling
  - 1,000 iterations of burn in
  - Parallelized sampling (approximation) using MPI

    [Newman et. al. 2009]
- 100 Event Types

[Happy, Nupur] Disliked manual annotation
[Anshul] 'Legal', 'Food' not event categories

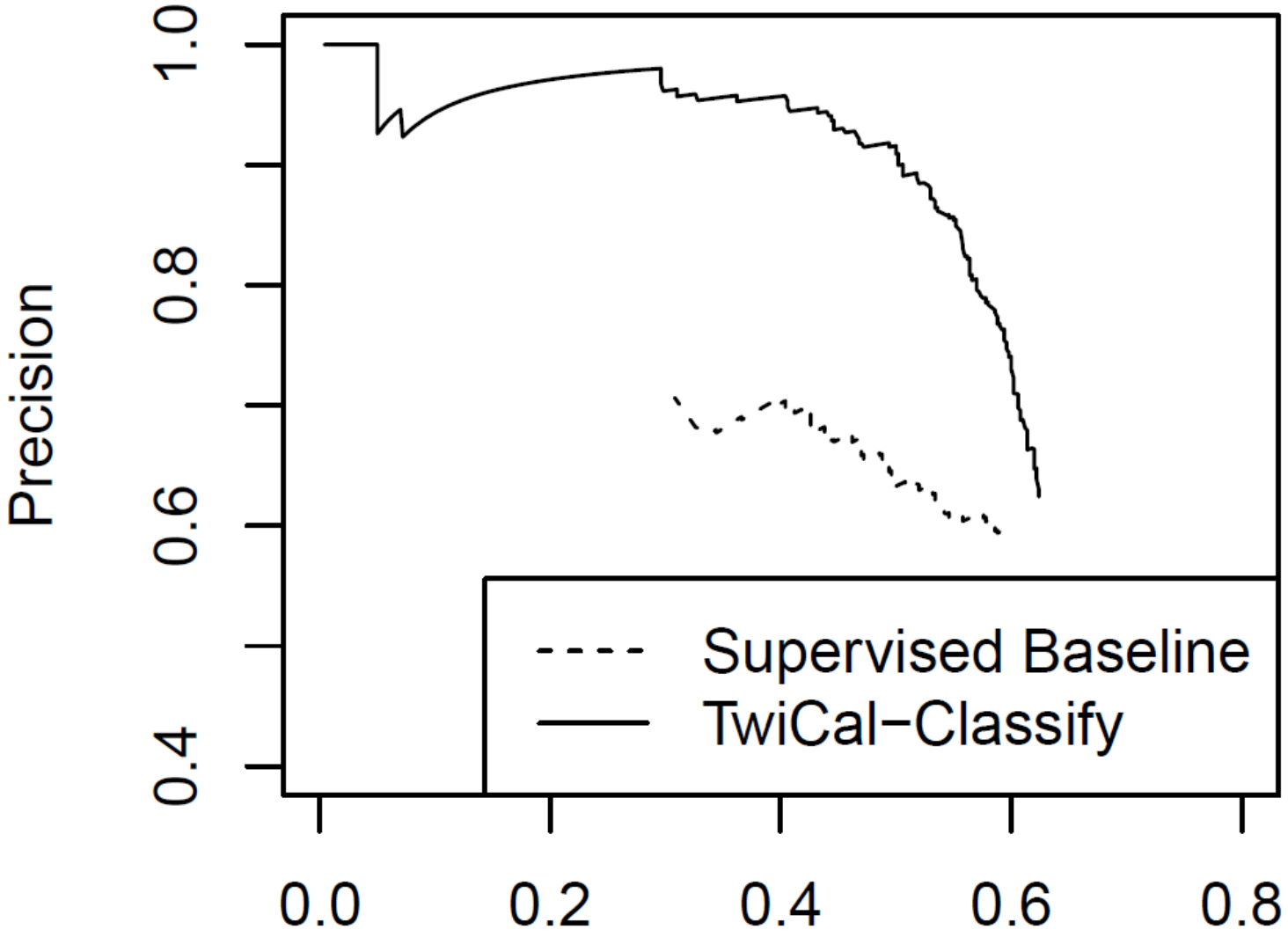
Label	Top 5 Event Phrases	Top 5 Entities
Sports	tailgate - scrimmage - tailgating - homecoming - regular season	espn - ncaa - tigers - eagles - varsity
Concert	concert - presale - performs - concerts - tickets	taylor swift - toronto - britney spears - rihanna - rock
Perform	matinee - musical - priscilla - seeing - wicked	shrek - les mis - lee evans - wicked - broadway
TV	new season - season finale - finished season - episodes - new episode	jersey shore - true blood - glee - dvr - hbo
Movie	watch love - dialogue theme - inception - hall pass - movie	netflix - black swan - insidious - tron - scott pilgrim
Sports	inning - innings - pitched - homered - homer	mlb - red sox - yankees - twins - dl
Politics	presidential debate - osama - presidential candidate - republican debate - debate performance	obama - president obama - gop - cnn - america

Experiment: Categorizing Events

- Randomly Sampled 500 (entity, date) pairs
- Annotated with event types
  - Using types discovered by the topic model
- Baseline:
  - Supervised classification using 10-fold cross validation
  - Treat event phrases like bag of words

[Nupur] Multiple entity events?
[Nupur, Anshul] Very simple baseline

Event Classification Performance



Ranking Events

- Twitter users often mention mundane events
  - What they ate for lunch
- Can't just list by frequency
  - Entities such as McDonalds would be frequent on most days
- Use Statistical Test between entities and dates
  - Only show if entities appear more than expected

G^2 Test for Ranking

- Given an entity e, and date d the statistic is computed as:

$$G^2 = \sum_{x \in \{e, \neg e\}, y \in \{d, \neg d\}} O_{x,y} \times ln\left(\frac{O_{x,y}}{E_{x,y}}\right)$$

$O_{e,d}$: Observed fraction of tweets containing e and d

$O_{e,\neg d}$: Tweets containing e and not d

$E_{e,d}$: Expected fraction of tweets containing e and d, assuming independence

[Happy, Akshay, Shantanu, Nupur, Anshul, Rishab, Dinesh R] Liked

[Barun, Shantanu] Same event on multiple days?

[Rishab] Why not χ^2?

End-to-end Evaluation

- Evaluate end-to-end calendar entries
- Collect tweets up to cutoff date
- Extract Named Entities, Event Phrases, Temporal Expressions
- Classify Event Type
- Rank Events
- Pick top K events occurring in a 2 week future window
- Evaluate Precision

[Akshay, Barun] Liked

End-to-end Evaluation

	precision				
# calendar entries	ngram baseline	entity + date	event phrase	event type	entity + date + event + type
100	0.50	0.90	0.86	0.72	0.70
500	0.46	0.66	0.56	0.54	0.42
1,000	0.44	0.52	0.42	0.40	0.32

- Ngram Baseline
  - No Named Entity Recognition
  - Rely on significance test to rank ngrams
  - A few extra heuristics (filter out temporal expressions etc…)

Extensions

- [Happy] Multiple entities: Eg: Ind vs Aus test series
- [Happy] Correlation b/w entity and location for ranking?
- [Arindam, Haroun, Prachi] Multi day event calendar, sparse bit vector based on MultiR, Regex for start/end date
- [Akshay, Dinesh R] Multiple entities for same event
- [Barun] Normalization for false negatives
- [Himanshu] Confidence score based on redundancy?
- [Anshul] Bootstrapping framework
- [Anshul] Bidirectional LSTM's with CRF's
- [Rishab] Joint inference with NER's
- [Prachi] Events as hierarchies
- [Surag] Sets of dates following a pattern
- [Surag] Weak association b/w entity and date – (entity, event type) and date