Non-parametric Methods for Unsupervised Semantic Modelling

Wray Buntine

Monash University

Thanks Lan Du of Macquarie University and Swapnil Mishra and Kar Wai Lim of ANU for many great slides

Wednesday 14th January, 2015
Outline

1. Background
   - Motivation
     - Probability Vector Networks
     - Dirichlet distributions
     - Old School Probabilistic Reasoning

2. Non-parametric Bayesian Methods

3. High Performance Topic Models (with Swapnil Mishra)

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
We need new tools to help us: organize, search, summarise and understand information. The field of Information Access serves this purpose.
Information Warfare

Definition: "the use and management of information in pursuit of a competitive advantage over an opponent."

- Email spam, link spam, etc.
  - Whole websites are fabricated with fake content to trick search engines.
  - Spammers using social networks to personalise attacks (Nov. 2011).
- BBC reports trust in information on the web is being damaged “by the huge numbers of people paid by companies to post comments” (Dec. 2011).

It’s an information war out there on the internet between consumers (i.e., you), companies, not-for-profits, voters, parties, employees, bureaucrats, academics, ....
Outline

1. Background
   - Motivation
   - Probability Vector Networks
     - Dirichlet distributions
     - Old School Probabilistic Reasoning

2. Non-parametric Bayesian Methods

3. High Performance Topic Models (with Swapnil Mishra)

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
Probability Vectors

We often have probability vectors for:

- the next word given \((n-1)\) previous,
- an author/conference/corporation to be linked to/from a webpage/patent/citation,
- part-of-speech of a word in context,
- hashtag in a tweet given the author.

We need to work with distributions over probability vectors for:

- inheritance and sharing of information;
- networks of probability vectors;
- inference and learning.
Sharing/Inheritance with a Probability Hierarchy

We might model a set of vocabularies/documents hierarchically:

\[
\vec{\theta}_1 \sim \text{Dirichlet}\left(\alpha_0 \vec{\phi}\right)
\]

\[
\vec{\theta}_{1,2} \sim \text{Dirichlet}\left(\alpha_1 \vec{\theta}_1\right)
\]

Statistical estimation with these generally difficult:

\[
\ldots \frac{1}{\text{Beta}\left(\alpha_1 \vec{\theta}_1\right)} \prod_k \theta_1^{\alpha_1 \theta_{1,k} - 1} \prod_k \theta_1^{\alpha_1 \phi_k - 1} \ldots
\]

(N.B. fixed point MAP solutions exist for hierarchies)
Overview: Latent Semantic Modelling

- Variety of component and network models can be made non-parametric with deep probability vector networks.
Overview: Latent Semantic Modelling

- Variety of component and network models can be made non-parametric with deep probability vector networks.
- Fast methods for training deep probability vector networks.
Overview: Latent Semantic Modelling

- Variety of component and network models can be made non-parametric with deep probability vector networks.
- Fast methods for training deep probability vector networks.
  - avoid Chinese Restaurant processes and stick-breaking for Pitman-Yor and Dirichlet Processes!
  - the minimum path assumption can be poor
  - concentration parameter often should be fit
Overview: Latent Semantic Modelling

- Variety of component and network models can be made non-parametric with deep probability vector networks.
- Fast methods for training deep probability vector networks.
  - avoid Chinese Restaurant processes and stick-breaking for Pitman-Yor and Dirichlet Processes!
  - the minimum path assumption can be poor
  - concentration parameter often should be fit
- Allows efficient modelling of latent semantics:
  - semantic resources to integrate (WordNet, sentiment dictionaries, etc.),
  - inheritance and shared learning across multiple instances,
  - hierarchical modelling,
  - deep latent semantics,
  - integrating semi-structured and networked content,

i.e. Same as deep neural networks!
Outline

1. Background
   - Motivation
   - Probability Vector Networks
   - Dirichlet distributions
   - Old School Probabilistic Reasoning

2. Non-parametric Bayesian Methods

3. High Performance Topic Models (with Swapnil Mishra)

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
The Dirichlet distribution is used to sample finite probability vectors.

\[ \tilde{p} \sim \text{Dirichlet}_K (\tilde{\alpha}) \]

where \( \tilde{\alpha} \) is a positive \( K \)-dimensional vector.
Background

Dirichlet distributions

4-D Dirichlet samples

\[ \vec{p}_0 \]

\[ \vec{p}_1 \sim \text{Dirichlet}_4(500\vec{p}_0) \]

\[ \vec{p}_2 \sim \text{Dirichlet}_4(5\vec{p}_0) \]

\[ \vec{p}_3 \sim \text{Dirichlet}_4(0.5\vec{p}_0) \]
Forms for 3-D Dirichlet

Consider $\vec{p} = (p_1, p_2, p_3)$ where $\sum_k p_k = 1$.

$\vec{p} \sim \text{Dirichlet}_3 (\vec{\alpha})$ means that $p(\vec{p} | \vec{\alpha})$ is

$$\frac{\Gamma(\alpha_1)\Gamma(\alpha_2)\Gamma(\alpha_3)}{\Gamma(\alpha_1 + \alpha_2 + \alpha_3)} p_1^{\alpha_1-1} p_2^{\alpha_2-1} p_3^{\alpha_3-1},$$

where $\Gamma(\cdot)$ is the Gamma function.

Also have derived parameters

$$\alpha_0 = \alpha_1 + \alpha_2 + \alpha_3$$

$$\vec{\mu} = (\mu_1, \mu_2, \mu_3) = \left( \frac{\alpha_1}{\alpha_0}, \frac{\alpha_2}{\alpha_0}, \frac{\alpha_3}{\alpha_0} \right),$$

where mean of $\vec{p}$ is $\vec{\mu}$, and $\alpha_0$ is a concentration parameter.
The Dirichlet Distribution
Dirichlet Details

- $\alpha_0 = \alpha_1 + \alpha_2 + \alpha_3$ is called the concentration parameter. Its significance is shown by the variance

$$\text{Var}_{\alpha}[\vec{p}_i] = \frac{1}{\alpha_0 + 1} E_{\alpha}[\vec{p}_i] (1 - E_{\alpha}[\vec{p}_i]).$$

- Let $\vec{n} \sim \text{multinomial}(N, \vec{p})$, then

$$p(\vec{n}|\vec{p}) = C_n^N p_1^{n_1} p_2^{n_2} p_3^{n_3}$$

So data sampled from a multinomial using $\vec{p}$ has the same functional form as the Dirichlet distribution on $\vec{p}$, i.e., conjugacy.

- Normalising constant for Dirichlet is called a Beta function:

$$\text{Beta}_3(\alpha_1, \alpha_2, \alpha_3) = \frac{\Gamma(\alpha_1)\Gamma(\alpha_2)\Gamma(\alpha_3)}{\Gamma(\alpha_1 + \alpha_2 + \alpha_3)}$$
Outline

1. Background
   - Motivation
   - Probability Vector Networks
   - Dirichlet distributions
   - Old School Probabilistic Reasoning

2. Non-parametric Bayesian Methods

3. High Performance Topic Models (with Swapnil Mishra)

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
Latent Dirichlet Allocation

LDA Model

∀_d_ \_?_d ~ Dirichlet_K(\vec{\alpha})
∀_k_ \_?_k ~ Dirichlet_W(\vec{\beta})
∀_d_,_n_ z_{d,n} ~ Categorical(\vec{\theta}_d)
∀_d_,_n_ w_{d,n} ~ Categorical(\vec{\phi}_{z_{d,n}})

Collapsed Posterior for Gibbs Sampling

\[ \prod_d \frac{\text{Beta}_K(\vec{n}_{\vec{\theta}_d} + \vec{\alpha})}{\text{Beta}_K(\vec{\alpha})} \prod_k \frac{\text{Beta}_W(\vec{n}_{\vec{\phi}_k} + \vec{\beta})}{\text{Beta}_W(\vec{\beta})} \]
Learning Algorithms with Dirichlets

Common text-book algorithms/methods in modern machine-learning/statistics rely on Dirichlet distributions combined with:

- trees, tables;
- graphs, networks;
- context free grammars;

Algorithms on these combine Dirichlet normalizers with:

- model search;
- model averaging;
- EM algorithm;
- etc.

Arguably, many of these are of poor/mixed quality.

*e.g.*, probabilistic context free grammars, decision trees
**Context Free Grammar**

\[ \mathcal{R} = \left\{ \begin{array}{l}
S \rightarrow NP \ VP \\
D \rightarrow \text{the} \\
NP \rightarrow D \ N \\
N \rightarrow \text{dog} \\
VP \rightarrow V \\
V \rightarrow \text{barks} \\
\end{array} \right\} \]

In a **probabilistic context free grammar**, probabilities are associated with each rule, and rules apply independently of context.
Probabilistic Context Free Grammar, cont.

- Doesn’t perform well in practice because statistically, context matters.
  
  Google bought Youtube.

- Previous words, or higher parts-of-speech do affect probabilities.
- State-of-the-art systems “hack” context by making probabilities dependent on:
  - previous few words in the input stream;
  - previous few parts-of-speech higher in the parse tree;
  - head (“main”) words for nodes higher in the parse tree.

- Alternatively, they introduce specialised parts-of-special to introduce context.

- This requires algorithmic sophistication!
Bayesian Networks

Cancer metastasis
\[ p(X_1 = 1) = 0.80 \]
\[ p(X_2 = 1 | X_1 = 1) = 0.80 \]
\[ p(X_2 = 1 | X_1 = 0) = 0.20 \]

Increase in calcium content in serum
\[ p(X_4 = 1 | X_2 = 1, X_3 = 1) = 0.80 \]
\[ p(X_4 = 1 | X_2 = 0, X_3 = 1) = 0.60 \]
\[ p(X_4 = 1 | X_2 = 1, X_3 = 0) = 0.60 \]
\[ p(X_4 = 1 | X_2 = 0, X_3 = 0) = 0.20 \]

Brain tumor
\[ p(X_3 = 1 | X_1 = 1) = 0.80 \]
\[ p(X_3 = 1 | X_1 = 0) = 0.20 \]

Deep coma

Severe headache
\[ p(X_5 = 1 | X_3 = 1) = 0.80 \]
\[ p(X_5 = 1 | X_3 = 0) = 0.20 \]
Bayesian Model Averaging (BMA) for Bayesian Networks

Summaries of the posterior distributions of $N$ for the spina bifida data for all models with posterior probability greater than 0.01. $\hat{N}$ is a Bayes estimate, minimizing a relative squared error loss function.

<table>
<thead>
<tr>
<th>Model</th>
<th>Posterior</th>
<th>$\hat{N}$</th>
<th>2.5%, 97.5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>$D \rightarrow R$ $\rightarrow B$</td>
<td>0.373</td>
<td>731</td>
<td>(701, 767)</td>
</tr>
<tr>
<td>$B \rightarrow D \rightarrow R$</td>
<td>0.301</td>
<td>756</td>
<td>(714,811)</td>
</tr>
<tr>
<td>$B \rightarrow R \rightarrow D$</td>
<td>0.281</td>
<td>712</td>
<td>(681,751)</td>
</tr>
<tr>
<td>$B \rightarrow R \rightarrow D$</td>
<td>0.036</td>
<td>697</td>
<td>(628,934)</td>
</tr>
<tr>
<td>Model Averaging</td>
<td>—</td>
<td>731</td>
<td>(682,797)</td>
</tr>
</tbody>
</table>

From Madigan and York, 1995
Bayesian Model Averaging (BMA) for Bayesian Networks, cont

Build a pool of “good” models (i.e., the graphical structures) based on the training data: Good-Models(\mathcal{X}).

BMA estimate of probability for new data \vec{x} given training sample \mathcal{X} is

\[
p(\vec{x}|\mathcal{X}) \approx \sum_{M \in \text{Good-Models}(\mathcal{X})} \frac{p(M|\mathcal{X})}{\sum_{M \in \text{Good-Models}(\mathcal{X})} p(M|\mathcal{X})} p(\vec{x}|M, \mathcal{X})
\]

Question: how do we build “good” models given there are a combinatoric number?
N-grams

To model a sequence of words $p(w_1, w_2, ..., w_N)$ we can use:

- **Unigram (1-gram) model:** $p(w_n | \vec{\theta}_1)$
- **Bigram (2-gram) model:** $p(w_n | w_{n-1}, \vec{\theta}_2)$
- **Trigram (3-gram) model:** $p(w_n | w_{n-1}, w_{n-2}, \vec{\theta}_3)$

By BMA with data $\mathcal{W}$, we use the sequence of $k$-gram models $M_k$:

$$\sum_{k=1}^{K} p(w_n | w_{n-1}, ..., w_{n-k+1}, \mathcal{W}, M_k) p(M_k | \mathcal{W})$$

where the first probability is the $k$-gram estimate from the data.

For $n$-grams, the good models are easy.
Decision Trees

<table>
<thead>
<tr>
<th>Tid</th>
<th>Refund</th>
<th>Marital Status</th>
<th>Taxable Income</th>
<th>Cheat</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Yes</td>
<td>Single</td>
<td>125K</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>No</td>
<td>Married</td>
<td>100K</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>No</td>
<td>Single</td>
<td>70K</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>Yes</td>
<td>Married</td>
<td>120K</td>
<td>No</td>
</tr>
<tr>
<td>5</td>
<td>No</td>
<td>Divorced</td>
<td>95K</td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>No</td>
<td>Married</td>
<td>60K</td>
<td>No</td>
</tr>
<tr>
<td>7</td>
<td>Yes</td>
<td>Divorced</td>
<td>220K</td>
<td>No</td>
</tr>
<tr>
<td>8</td>
<td>No</td>
<td>Single</td>
<td>85K</td>
<td>Yes</td>
</tr>
<tr>
<td>9</td>
<td>No</td>
<td>Married</td>
<td>75K</td>
<td>No</td>
</tr>
<tr>
<td>10</td>
<td>No</td>
<td>Single</td>
<td>90K</td>
<td>Yes</td>
</tr>
</tbody>
</table>

There could be more than one tree that fits the same data!

Classification on new data based on tree. Early classical non-parametric method made famous with CART (Breiman, Friedman and Stone) and ID3/C4.5 (Quinlan).
Breiman’s Random Forests or Bagging of Decision Trees

The ensemble model

Forest output probability \( p(c|v) = \frac{1}{T} \sum_{t} p_t(c|v) \)

(source unknown!)

1990: York and Madigan develop **BMA** for Bayesian networks.

1994: Breiman developed **bagging** (or random forests) for trees as a Frequentist response:

→ still one of the top performing classification algorithms

1995: Willems, Shtarkov, Tjalkens adapt **BMA** for n-grams, **context tree weighting (CTW)** for lossless compression.

Bayesian model averaging and Frequentist bagging became dominant paradigms.
Bayesian Model Averaging and Non-parametrics Storyline, cont.

2006: Y.W. Teh develops hierarchical Pitman-Yor model for n-grams.

2009: Gasthaus, Wood, Archambeau, Teh and James develop Sequence Memoizer for n-grams for lossless compression. Beats CTW.

2009: Wood and Teh develop Statistical Language Model Domain Adaptation. Further improves n-gram modelling by allowing adaptation.

• but the algorithm is impractical

Non-parametric Bayesian methods give new life to BMA because they are substantially better priors.
Motivation

- While somewhat successful, the BMA paradigm based on standard (simple) conjugate priors has reached a limit for some models consisting of Dirichlets.
- Many problems in machine learning are ripe for improvement with better modelling of context.
- But this requires efficient non-parametric modelling.
Outline

1. Background

2. Non-parametric Bayesian Methods
   - Hierarchical Dirichlet and Pitman-Yor Processes
   - PYPs on Discrete Data
   - Non-parametric Feature Models

3. High Performance Topic Models (with Swapnil Mishra)

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
Dirichlet Process

The **Dirichlet Process (DP)** has two arguments $\text{DP}(\alpha, H(\cdot))$:

- $H(\cdot)$ is another distribution that is the mean of the DP
- when $H(\cdot) = \vec{\mu}$, applied to a finite probability vector $\vec{\mu}$ of dimension $K$, the DP and the Dirichlet are identical:

$$\text{Dirichlet}_K(\alpha, \vec{\mu}) = \text{DP}(\alpha, \vec{\mu}).$$

- often the use of a DP is equivalent to the use of a Dirichlet.
- **why use the DP then?**
  - nested/hierarchical DPs have fairly fast samplers.
The **Pitman-Yor Process (PYP)** $\text{PYP}(d, \alpha, H(\cdot))$:

- extends the DP with an extra parameter, and is more suited to Zipfian data,
- discount $d$ makes the expected probabilities converge faster,
- the PYP is not Dirichlet, unlike the DP
- same samplers as the DP
Bayesian Idea: Similar Context Means Similar Word

- Words in **a ??** should be like words in **?**
  - though no plural nouns

- Words in **caught a ??** should be like words in **a ??**
  - though a suitable object for “caught”

- Words in **he caught a ??** be very like words in **caught a ??**
  - “he” shouldn’t change things much
Bayesian N-grams

Build all three together, making the 1-gram a prior for the 2-grams, and the 2-grams a prior for the 3-grams, etc.

For this, we need to say each probability vector in the hierarchy is a variant of its parent.
Bayesian N-grams, cont.

\[ S = \text{symbol set, fixed or possibly countably infinite} \]
\[ \tilde{\mathbf{p}}. \sim \text{prior on prob. vectors (initial vocabulary)} \]
\[ \tilde{\mathbf{p}}. | x_1 \sim \text{dist. on prob. vectors with mean } \tilde{\mathbf{p}}. \quad \forall x_1 \in S \]
\[ \tilde{\mathbf{p}}. | x_1, x_2 \sim \text{dist. on prob. vectors with mean } \tilde{\mathbf{p}}. | x_1 \quad \forall x_1, x_2 \in S \]
Historical Context

1990s: Pitman and colleagues in mathematical statistics develop statistical theory of partitions, Pitman-Yor process, etc.


2006: Teh develops hierarchical n-gram models using HPYs.

2006: Teh, Jordan, Beal and Blei develop hierarchical Dirichlet processes (HDP), e.g. applied to LDA.

2006-2011: Chinese restaurant processes (CRPs) go wild!

- require dynamic memory in implementation,
- Chinese restaurant franchise,
- multi-floor Chinese restaurant process,
- etc.
Historical Context

1990s: Pitman and colleagues in mathematical statistics develop statistical theory of partitions, Pitman-Yor process, etc.


2006: Teh develops hierarchical n-gram models using HPYs.

2006: Teh, Jordan, Beal and Blei develop hierarchical Dirichlet processes (HDP), e.g. applied to LDA.

2006-2011: Chinese restaurant processes (CRPs) go wild!
- require dynamic memory in implementation,
- Chinese restaurant franchise,
- multi-floor Chinese restaurant process,
- etc.

2011: Chen, Du, Buntine show Chinese restaurants and stick-breaking not needed by introducing block table indicator samplers.
Outline

1. Background

2. Non-parametric Bayesian Methods
   - Hierarchical Dirichlet and Pitman-Yor Processes
   - PYPs on Discrete Data
   - Non-parametric Feature Models

3. High Performance Topic Models (with Swapnil Mishra)

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
The Ideal Hierarchical Component?

We want a magic distribution that looks like a multinominal likelihood in $\vec{\theta}$.

\[ \vec{p} \sim \text{Magic}(\alpha, \vec{\theta}) \]

\[ x_n \sim \text{Discrete}(\vec{p}) \quad \forall n \]

\[
p \left( \vec{n} \middle| \alpha, \vec{\theta}, N \right)
= F_\alpha(\vec{n}) \prod_k \theta_{tk}^{n_k}
\]

where \[ \sum_k n_k = N \]
The PYP/DP is the Magic

- The PYP/DP plays the role of the magic distribution.
- However, the exponent $t_k$ for the $\theta$ now becomes a latent variable, so needs to be sampled as well.
- The $t_k$ are constrained:
  - $t_k \leq n_k$
  - $t_k > 0$ iff $n_k > 0$
- The $\vec{t}$ act like data for the next level up involving $\vec{\theta}$.

\[
p \left( \vec{n}, \vec{t} \mid d, \alpha, \vec{\theta}, N \right) = F_{d,\alpha}(\vec{n}, \vec{t}) \prod_k \theta_{k}^{t_k}
\]

where $\sum_k n_k = N$
Interpreting the Auxiliary Counts

**Interpretation:** $t_k$ is how much of the count $n_k$ that affects the parent probability (i.e. $\bar{\theta}$).

- If $\bar{t} = \bar{n}$ then the sample $\bar{n}$ affects $\bar{\theta}$ 100%.
- When $n_k = 0$ then $t_k = 0$, no effect.
- If $t_k = 1$, then the sample of $n_k$ affects $\bar{\theta}$ minimally.

\[
p\left(\bar{n}, \bar{t} \mid d, \alpha, \bar{\theta}, N\right) = F_{d,\alpha}(\bar{n}, \bar{t}) \prod_k \theta_{tk}
\]

where \(\sum_k n_k = N\)
Why We Prefer DPs and PYPs over Dirichlets!

For the PYP, the $\theta_k$ just look like multinomial data, but you have to introduce a discrete latent variable $\vec{t}$.

For the Dirichlet, the $\theta_k$ are in a complex gamma function.
Typed Data with Species

Within types there are separate species, pink and orange for type $k$, blue and green for type $l$. Chinese restaurant samplers work in this space.
Within types there are separate *species*, but we only know which data is the first of a new species. For other data, species is unknown. **Block table indicator samplers work in this space.**
Better Sampling Methods for HDP and HPYP

Sampling for hierarchical Dirichlet Processes and Pitman-Yor Processes:

The Old: hierarchical Chinese Restaurant Processes (CRP) from Teh et al. 2006.


- requires no dynamic memory
- more rapid mixing so leads to better models
- more easily applied to more complex models
- demonstrated extensively on different problems!

See http://topicmodels.org, “A tutorial on non-parametric methods”
Outline

1. Background

2. Non-parametric Bayesian Methods
   - Hierarchical Dirichlet and Pitman-Yor Processes
   - PYPs on Discrete Data
   - Non-parametric Feature Models

3. High Performance Topic Models (with Swapnil Mishra)

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
Conjugate Discrete Families

Well known families used in discrete statistics:

- **multinomial-Dirichlet** (*e.g.*, in LDA)
- **Poisson-Gamma** (*e.g.*, in some versions of NMF)
- **Bernoulli-Beta** (*e.g.*, basis of IBP)
- **negative-Binomial-Gamma**
  - not quite conjugate, the negative-Binomial is version of a Poisson-Gamma and is used as a “robust” Poisson

Each has a corresponding non-parametric version:

- effectively uses the improper version of the prior (Gamma, Beta, Dirichlet),
- want to generate a countably infinite number of them but have almost all infinitesimally small,
- theory done with Poisson processes; see 2014 ArXiv paper by Lancelot James
Conjugate non-Parametric Discrete Families

- Let $\omega_k$ for $k = 1, \ldots, \infty$ be index points for an infinite vector.
- Have (unknown) infinite parameter vector $\bar{\lambda} = \sum_{k=0}^{\infty} \lambda_k \delta_{\omega_k}$.
- Generate $I$ finite discrete feature vectors $\bar{x}_i = \sum_{k=0}^{\infty} x_{i,k} \delta_{\omega_k}$.
- Intent is that only finite number of $x_{i,k} \neq 0$ for given $i$.

Non-parametric versions of models for discrete feature vectors:

| Process Name                  | $p(x|\lambda)$                          | $p(\lambda)$                |
|-------------------------------|-----------------------------------------|------------------------------|
| Poisson-Gamma                 | $\frac{1}{x!} \lambda^x e^{-\lambda}$  | $\theta \lambda^{-\alpha-1} e^{-\beta \lambda}$ |
| Bernoulli-Beta                | $\lambda^x (1 - \lambda)^{1-x}$        | $\theta \lambda^{-\alpha-1} (1 - \lambda)^{\alpha+\beta-1}$ |
| negative-Binomial-Gamma       | $\frac{1}{x!} (\lambda)^x \rho^x (1 - \rho)^\lambda$ | $\theta \lambda^{-\alpha-1} e^{-\beta \lambda}$ |

$(\lambda)_x$ is rising factorial $\lambda(\lambda + 1)\ldots(\lambda + x - 1)$

(negative-Binomial not quite conjugate, an approximation is needed for key integral)
Conjugate non-Parametric Discrete Families, cont.

- **Posterior marginal** given by

\[
p(\vec{x}_1, \ldots, \vec{x}_l | \vec{\omega}) = e^{-\Psi_I} \prod_{k=1}^{K} \left( \int_0^{\infty} \prod_{i=1}^{l} p(x_{i,k} | \lambda) \rho(\lambda | \omega) d\lambda \right) G_0(d\omega_k)
\]

where \( \omega_1, \ldots, \omega_K \) is entries in \( \vec{\omega} \) that have some non-zero data \( x_{i,k} \).

- Intuitively, let \( \Psi_I \) be **Poisson non-zero rate** for a dimension giving \( l \) samples with at least one non-zero entry.

  - \( (1 - p(x_{i,k} \neq 0 | \lambda)^l) = \text{“not all } l \text{ samples } x_{1,k}, \ldots, x_{l,k} \text{ are zero”} \)
  - then

\[
\Psi_I = \int_\Omega \int_0^{\infty} (1 - p(x_{i,k} = 0 | \lambda)^l) \rho(\lambda | \omega) d\lambda G_0(d\omega_k)
\]
Bernoulli-Beta Process (Indian Buffet Process)

- Infinite Boolean vectors \( \bar{x}_i \) with a finite number of 1’s;
- Each parameter is an independent probability \( \lambda_k \),

\[
p(x_{i,k} | \lambda_k) = \lambda_k^{x_{i,k}} (1 - \lambda_k)^{1 - x_{i,k}}
\]

- For finite 1’s, \( \sum_k \lambda_k < \infty \)
- Poisson rate is the 3-parameter Beta process

\[
p(\lambda | \alpha, \beta, \theta) = \theta \lambda^{-\alpha - 1} (1 - \lambda)^{\alpha + \beta - 1}
\]

(some versions add additional constants with \( \theta \))
- Is in improper Beta because seeing “1” makes it proper:

\[
\int_{\lambda=0}^{1} p(x = 1 | \lambda) p(\lambda) d\lambda = \theta \text{Beta}(1 - \alpha, \alpha + \beta)
\]
Bernoulli-Beta Process (Indian Buffet Process)

- The Poisson non-zero rate trick: evaluate using $1 - y^l = (1 - y) \sum_{i=0}^{l} y^i$

$$
\Psi_L = \theta \Gamma(1 - \alpha) \sum_{i=0}^{l} \frac{\Gamma(\beta + \alpha + i)}{\Gamma(\beta + 1 + i)} .
$$

- The marginal for the $k$-th dimension

$$
\int_{0}^{\infty} \prod_{i=1}^{l} p(x_{i,k} | \lambda) \rho(\lambda | \omega) d\lambda = \theta \text{Beta}(c_k - \alpha, l - c_k + \alpha + \beta)
$$

where $c_k$ is times dimension is “on,” so $c_k = \sum_{i=1}^{l} x_{i,k}$.

- Gibbs sampling $x_{i,k}$ is thus simple.

- Sampling parameters: posterior of $\theta$ is Poisson; $\beta$ is posterior log-concave so sampling “easier”. challenging
normalised heterogenous generalised Gamma process.

- Construct $p(\lambda | \omega_k)$ as follows:

$$
\rho(\psi_k | \omega_k) \sim \text{Generalised-Gamma-Process}(\alpha, 1) \\
p(\beta_k | \omega_k) \sim \text{Gamma}(1.1, 1.1) \\
p(\lambda_k | \psi_k, \omega_k) \sim \text{Gamma}(\psi, \beta_k)
$$

- Now have multinomial with probabilities $\lambda_k / \sum_k \lambda_k$. 
Outline

1. Background

2. Non-parametric Bayesian Methods

3. High Performance Topic Models (with Swapnil Mishra)
   - Topic Models
   - Background
   - Evolution of Models
   - Our Non-parametric Topic Model

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
Component Models, Generally

Approximate faces/bag-of-words (RHS) with a linear combination of components (LHS).
Matrix Approximation View

\[ W \simeq L \times \Theta^T \]

- Different variants:
  - **Data** $W$:
    - real valued
    - non-negative
    - non-neg integer
    - real valued
  - **Components** $L$:
    - unconstrained
    - non-negative
    - non-negative independent
  - **Error**:
    - least squares
    - least squares
    - small
  - **Models**:
    - PCA and LSA
    - learning codebooks, NMF
    - topic modelling, NMF
    - ICA
Why Topic Models?

- *Topic Models* discover hidden themes in text data to aid understanding.
- Recent research develops higher performance topic models.
Why Topic Models?

- **Topic Models** discover hidden themes in text data to aid understanding.
- Recent research develops higher performance topic models.
- But why should you care?
- Moreover, why should I care?
Topic Models: Potential for Semantics

Following sets of topic words created from the New York Times 1985-2005 news collection using hca (see Buntine and Mishra, KDD 2014):

- career, born, grew, degree, earned, graduated, became, studied, graduate
- mother, daughter, son, husband, family, father, parents, married, sister
- artillery, shells, tanks, mortars, gunships, rockets, firing, tank
- clues, investigation, forensic, inquiry, leads, motive, investigator, mystery
- freedom, tyranny, courage, america, deserve, prevail, evil, bless, enemies
- viewers, cbs, abc, cable, broadcasting, channel, nbc, broadcast, fox, cnn
- anthrax, spores, mail, postal, envelope, powder, letters, daschle, mailed

Topic models yield high-fidelity semantic associations!
Topic Models: Just an Intermediate Goal

Topic models are the leading edge of a new wave of deep latent semantic models applied to real NLP tasks:

e.g., document segmentation, word sense disambiguation, facet discovery for sentiment analysis, unsupervised POS discovery, social networks, ...

in the middle of this segmentation model is a topic model

i.e., we don’t care about topic models per se!
ASIDE: Aspects, Ratings and Sentiments


State of the art sentiment model.

Typical methods currently lack probability vector hierarchies.

Figure 1: Factorized rating and review model.
Evaluation

David Lewis (Aug 2014) “topic models are like a Rorschach inkblot test” (not his exact words .... but the same idea)
Evaluation

David Lewis (Aug 2014) “topic models are like a Rorschach inkblot test” (not his exact words .... but the same idea)

Perplexity:
- measure of test set likelihood;
- equal to effective size of vocabulary;
- we use “document completion,” see Wallach, Murray, Salakhutdinov, and Mimno, 2009;
- however it is not a bonafide evaluation task
Evaluation

David Lewis (Aug 2014) “topic models are like a Rorschach inkblot test” (not his exact words .... but the same idea)

Perplexity:
- measure of test set likelihood;
- equal to effective size of vocabulary;
- we use “document completion,” see Wallach, Murray, Salakhutdinov, and Mimno, 2009;
- however it is not a bonafide evaluation task

PMI:
- measure of topic coherence: “average pointwise mutual information between all pairs of top 10 words in the topic”
- see Newman, Lau, Grieser, and Baldwin, 2010; Lau, Newman and Baldwin, 2014
- but at least it corresponds to a semi-realistic evaluation task
Outline

1. Background

2. Non-parametric Bayesian Methods

3. High Performance Topic Models (with Swapnil Mishra)
   - Topic Models
   - Background
   - Evolution of Models
   - Our Non-parametric Topic Model

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
Previous Work

- “Topic models with power-law using Pitman-Yor process,” Sato and Nakagawa 2010
Text and Burstiness

Original news article:
Women may only account for 11% of all Lok-Sabha MPs but they fared better when it came to representation in the Cabinet. Six women were sworn in as senior ministers on Monday, accounting for 25% of the Cabinet. They include Swaraj, Gandhi, Najma, Badal, Uma and Smriti.

Bag of words:
11% 25% Badal Cabinet(2) Gandhi Lok-Sabha MPs Monday Najma Six Smriti Swaraj They Uma Women account accounting all and as better but came fared for(2) in(2) include it may ministers of on only representation senior sworn the(2) they to were when women

NB. “Cabinet” appears twice! It is bursty (see Doyle and Elkan, 2009)
Aside: Burstiness and Information Retrieval

- burstiness and eliteness are concepts in information retrieval used to develop BM25 (i.e. dominant TF-IDF version)
- the two-Poisson model and the Pitman-Yor model can be used to justify theory (Sunehag, 2007; Puurula, 2013)
- relationships not yet fully developed
Outline

1. Background
2. Non-parametric Bayesian Methods
3. High Performance Topic Models (with Swapnil Mishra)
   - Topic Models
   - Background
   - Evolution of Models
   - Our Non-parametric Topic Model
4. Twitter Opinion Topic Model (with Kar Wai Lim)
5. Segmentation with a Structured Topic Model
Evolution of Models

LDA - Scalar
original LDA
LDA- Vector
adds asymmetric Dirichlet prior like Wallach et al.;
is also truncated HDP-LDA;
implemented by Mallet since 2008 as assymetric-symmetric LDA
no one knew!
Evolution of Models

HDP-LDA
adds proper modelling of topic prior like Teh et al.
Evolution of Models

NP-LDA
adds power law on word distributions
like Sato et al. and estimation of background word distribution
High Performance Topic Models (with Swapnil Mishra)

Evolution of Models

NP-LDA with Burstiness

add’s burstiness like Doyle and Elkan
Outline

1. Background
2. Non-parametric Bayesian Methods
3. High Performance Topic Models (with Swapnil Mishra)
   - Topic Models
   - Background
   - Evolution of Models
   - Our Non-parametric Topic Model
4. Twitter Opinion Topic Model (with Kar Wai Lim)
5. Segmentation with a Structured Topic Model
Our Non-parametric Topic Model

\[ \{\vec{\theta}_d\} = \text{document} \otimes \text{topic matrix} \]
\[ \{\vec{\phi}_k\} = \text{topic} \otimes \text{word matrix} \]
\[ \vec{\alpha} = \text{prior for document} \otimes \text{topic matrix} \]
\[ \vec{\beta} = \text{prior for topic} \otimes \text{word matrix} \]

- Full fitting of priors, and their hyperparameters.
- Topic \otimes \text{word vectors} \vec{\phi}_k specialised to the document to yield \vec{\psi}_k.
Our Non-parametric Topic Model, cont.

The blue nodes+arcs are Pitman-Yor process hierarchies.

Note in \( \{\psi_k\} \) there are hundreds times more parameters than data points!
Our Non-parametric Topic Model, cont.

The red nodes are hyper-parameters fit with Adaptive-Rejection sampling or slice sampling.

Use DP on document side ($a_\alpha = 0, a_\theta = 0$) as fitting usually wants this anyway.
Our Non-parametric Topic Model, cont.

Auxiliary latent variables (the $\tilde{t}$) propagate part of the counts (their $\tilde{n}$) up to the parent.

We keep/recompute sufficient statistics for matrices.

e.g. the $\tilde{\psi}$ statistics $\tilde{n}_{\psi,d,k}, \tilde{t}_{\psi,d,k}$ are not stored but recomputed from booleans as needed.

Double the memory of regular LDA, and only static memory.
Hierarchical priors: whenever parts of the system seem similar, we give them a common prior and learn the similarity.
Hierarchical priors: whenever parts of the system seem similar, we give them a common prior and learn the similarity.

Estimating parameters: whenever parameters cannot be reasonable set, we learn them instead.
Hierarchical priors: whenever parts of the system seem similar, we give them a common prior and learn the similarity.

Estimating parameters: whenever parameters cannot be reasonable set, we learn them instead.

Burstiness: we developed a Gibbs sampler that acts as a front end to any LDA-style model with Gibbs:

- implemented as a C function that calls the Gibbs sampler
- adds smallish memory (20%) and time (20%) overhead
- in all, NP-LDA with burstiness is double memory and time to regular LDA Gibbs sampling
- multi-core implementation good for upto 8 core
Performance on Reuters-21578 ModLewis Split

Training on 11314 news articles with vocabulary of 16994.
Perplexity performance on MLT Data for different Topics

2691 abstracts from the JMLR including 306 test documents with a vocabulary of 4662 words
Comparison to PCVB0 and Mallet

Protocol is train on 80% of all documents then using trained topic probs get predictive probabilities on remaining 20%, and replicate 5 times.

- Data contributed by Sato. Protocol by Sato et al.
- PCVB0 is by Sato, Kurihara, Nakagawa KDD 2012.
- Mallet (asymmetric-symmetric) is a truncated HDP implementation.
Comparison to Bryant+Sudderth (2012) on NIPS data
Comparison to FTM and LIDA

FTM and LIDA use IBP models to select words/topics within LDA. Archambeau, Lakshminarayanan, and Bouchard; Trans IEEE PAMI 2014.

<table>
<thead>
<tr>
<th>Data</th>
<th>KOS</th>
<th>NIPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>FTM (1-par)</td>
<td>7.262±0.007</td>
<td>6.901±0.005</td>
</tr>
<tr>
<td>FTM (3-par)</td>
<td>7.266±0.009</td>
<td>6.883±0.008</td>
</tr>
<tr>
<td>LIDA</td>
<td>7.257±0.010</td>
<td>6.795±0.007</td>
</tr>
<tr>
<td>HPD-LDA</td>
<td>7.253±0.003</td>
<td>6.792±0.002</td>
</tr>
<tr>
<td>time</td>
<td>3 min</td>
<td>22 min</td>
</tr>
<tr>
<td>NP-LDA</td>
<td>7.156±0.003</td>
<td>6.722±0.003</td>
</tr>
</tbody>
</table>

- KOS data contributed by Sato (D=3430, V=6906).
- NIPS data from UCI (D=1500, V=12419).
- Protocol same as with PCVB0 but a 50-50 split.
- Figures are log perplexity. Using 300 cycles.

- Better implementation of HDP-LDA now similar to LIDA.
- But LIDA still substantially better than LDA so we need to consider combining the technique with NP-LDA.
Conclusion on Topic Models

- NP-LDA model is about 50% slower than HDP-LDA but usually performs substantially better
  - most previous work failed to show this
Conclusion on Topic Models

- NP-LDA model is about 50% slower than HDP-LDA but usually performs substantially better
  - most previous work failed to show this
- burstiness has substantially improved topic comprehensibility and dramatically improved perplexity
  - now available, efficiently, for a broad variety of models
Conclusion on Topic Models

- NP-LDA model is about 50% slower than HDP-LDA but usually performs substantially better
  - most previous work failed to show this
- burstiness has substantially improved topic comprehensibility and dramatically improved perplexity
  - now available, efficiently, for a broad variety of models
- NP-LDA using block table indicator Gibbs sampling methods from Chen et al. (2011) are superior to (several) state-of-the-art algorithms.
  - simple (4-8 cpu) multicore version available

See KDD 2014 paper by Mishra and Buntine.
Conclusion on Topic Models

- NP-LDA model is about 50% slower than HDP-LDA but usually performs substantially better
  - most previous work failed to show this
- burstiness has substantially improved topic comprehensibility and dramatically improved perplexity
  - now available, efficiently, for a broad variety of models
- NP-LDA using block table indicator Gibbs sampling methods from Chen et al. (2011) are superior to (several) state-of-the-art algorithms.
  - simple (4-8 cpu) multicore version available
- Still need to explore IBP (as in LIDA) and split-merge techniques.
Conclusion on Topic Models

- NP-LDA model is about 50% slower than HDP-LDA but usually performs substantially better
  - most previous work failed to show this
- burstiness has substantially improved topic comprehensibility and dramatically improved perplexity
  - now available, efficiently, for a broad variety of models
- NP-LDA using block table indicator Gibbs sampling methods from Chen et al. (2011) are superior to (several) state-of-the-art algorithms.
  - simple (4-8 cpu) multicore version available
- Still need to explore IBP (as in LIDA) and split-merge techniques.
- Grab our topic modelling code from
  - https://github.com/wbuntine/topic-models
  - http://mloss.org/software/view/527/

See KDD 2014 paper by Mishra and Buntine.
Outline

1. Background
2. Non-parametric Bayesian Methods
3. High Performance Topic Models (withSwap-nil Mishra)
4. Twitter Opinion Topic Model (with Kar Wai Lim)
5. Segmentation with a Structured Topic Model
6. Conclusion
Aspect-based Opinion Aggregation

- Opinion Aggregation for reviews.
  - A process to collect reviews of products and services to analyze in aggregate.
- Aspect-based.
  - Groups reviews based on “aspects”.
  - Example:
    - Product types
      - Game consoles
      - Mobile phones
    - Product specs
      - Computer specs
      - Flight quality

<table>
<thead>
<tr>
<th>Aspect</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Game console</td>
<td>PS4, Xbox One, Wii U...</td>
</tr>
<tr>
<td>Mobile phone</td>
<td>iPhone, Samsung Note...</td>
</tr>
<tr>
<td>Computer spec</td>
<td>CPU, RAM, GPU...</td>
</tr>
<tr>
<td>Flight quality</td>
<td>Food, customer service...</td>
</tr>
</tbody>
</table>
Model target-opinion interaction directly. This improves opinion prediction significantly.
Explaining the Model, cont.

Emotion indicator – determined by seen emoticons or strong sentiment words (such as ‘happy’, ‘sad’).

Positive opinions tend to associate with positive emotions.
Incorporate sentiment lexicon as prior.
## Explaining the Model, cont.

<table>
<thead>
<tr>
<th>Target (t)</th>
<th>+/−</th>
<th>Opinions (o)</th>
</tr>
</thead>
<tbody>
<tr>
<td>phone</td>
<td>−</td>
<td>dead  damn stupid bad crazy</td>
</tr>
<tr>
<td></td>
<td>+</td>
<td>mobile smart good great f***ing</td>
</tr>
<tr>
<td>battery life</td>
<td>−</td>
<td>terrible poor bad horrible non-existence</td>
</tr>
<tr>
<td></td>
<td>+</td>
<td>good long great 7hr ultralong</td>
</tr>
<tr>
<td>game</td>
<td>−</td>
<td>addictive stupid free full addicting</td>
</tr>
<tr>
<td></td>
<td>+</td>
<td>great good awesome favorite cat-and-mouse</td>
</tr>
<tr>
<td>sausage</td>
<td>−</td>
<td>silly argentinian cold huge stupid</td>
</tr>
<tr>
<td></td>
<td>+</td>
<td>hot grilled good sweet awesome</td>
</tr>
</tbody>
</table>

* Words in **bold** are more specific and can only describe certain targets.
1. Background

2. Non-parametric Bayesian Methods

3. High Performance Topic Models (with Swapnil Mishra)

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
   - Document Segmentation (with Lan Du and Mark Johnson)
   - With a Structured Topic Model
Task, Roughly

Unsegmented Text

The EPA has issued a rule requiring greater use of ethanol. Made from corn, ethanol helps gasoline burn cleaner and lowers some kinds of emissions. Well, if you drive a car, you’ll really be interested in our next report. New rules about gasoline mean cleaner air but higher prices at the pump. CNN’s Deborah Potter explains. The government is opening the way for more of this [corn] to wind up, not in the trough or on the table, but here, at the gas pump. To clear the air in the nation’s smoggiest cities, the EPA has ordered the use of cleaner-burning gasoline. Adding renewable fuels to gasoline promotes products that are grown on American farms by American farmers. It promotes environmentally friendly jobs. It reduces air pollution. It protects the public’s health.

Segmented Text

The EPA has issued a rule requiring greater use of ethanol. Made from corn, ethanol helps gasoline burn cleaner and lowers some kinds of emissions.

Well, if you drive a car, you’ll really be interested in our next report. New rules about gasoline mean cleaner air but higher prices at the pump. CNN’s Deborah Potter explains.

The government is opening the way for more of this [corn] to wind up, not in the trough or on the table, but here, at the gas pump. To clear the air in the nation’s smoggiest cities, the EPA has ordered the use of cleaner-burning gasoline.

Adding renewable fuels to gasoline promotes products that are grown on American farms by American farmers. It promotes environmentally friendly jobs. It reduces air pollution. It protects the public’s health.
Task, Roughly

**Passage:** contiguous text with no boundary, *e.g.*, a sentence

**Segment:** consecutive text passages that are semantically related.

**Document:** a sequence of topically coherent text segments.

**Document Segmentation Task:** (roughly) given a document as a monolithic block of text, where should we put the segment boundaries.
Motivation—Structured Topic Modelling

Structured topic models (STM) by Du et al., (2010): hierarchical topic models with non-parametric Bayesian methods.

Has a hierarchy of topic probability vectors corresponding to document structure.
Bayesian Segmentation

Bayesian word segmentation models (Goldwater et al., 2009)

- Learn to place boundaries after phonemes in an utterance.
- A pointwise boundary sampling algorithm: compute the probability of placing a word boundary after each phoneme.
Bayesian Segmentation

Bayesian word segmentation models (Goldwater et al., 2009)

- Learn to place boundaries after phonemes in an utterance.
- A pointwise boundary sampling algorithm: compute the probability of placing a word boundary after each phoneme.

Motivation:
- treat text passages like phonemes in the model,
- i.e., estimate text passage boundaries as per phoneme boundaries.
Outline

1. Background

2. Non-parametric Bayesian Methods

3. High Performance Topic Models (with Swapnil Mishra)

4. Twitter Opinion Topic Model (with Kar Wai Lim)

5. Segmentation with a Structured Topic Model
   - Document Segmentation (with Lan Du and Mark Johnson)
   - With a Structured Topic Model
Hypothesis: simultaneously learning topic segmentation and topic identification should allow better detection of topic boundaries.
Segmentation Model–Generative process

Generative process

\[ \vec{\phi} \sim \text{Dirichlet}(\vec{\gamma}) \]
\[ \vec{\mu} \sim \text{Dirichlet}(\vec{\alpha}) \]
\[ \pi \sim \text{Beta}(\vec{\lambda}) \]
\[ \vec{\nu} \sim \text{PYP}(a, b, \vec{\mu}) \]
\[ \rho \sim \text{Bernoulli}(\pi) \]
\[ z \sim \text{Discrete}(\vec{\nu}_s) \]
\[ w \sim \text{Discrete}(\vec{\phi}_z) \]

- \( z \): topic assignment of word \( w \);
- \( N \): the number of words in a passage.
We need to sample the topic assignments $z$ and segment boundaries $\rho$. 

**Computed**

**Marginalised**

**Hyperparameters**

**Data**
Experiments on two meeting transcripts

Figure: Probability of a topic boundary, compared with gold-standard segmentation on one ICSI transcript.

<table>
<thead>
<tr>
<th>Gold Standard</th>
<th>PLDA</th>
<th>TSM</th>
</tr>
</thead>
<tbody>
<tr>
<td>{77, 95, 189, 365, 508, 609, 860}</td>
<td>{96, 136, 203, 226, 361, 508, 860}</td>
<td>{85, 96, 188, 363, 499, 508, 860}</td>
</tr>
</tbody>
</table>
Conclusion of Segmentation with a Structured Topic Model

Paper given at NAACL 2013, main author Lan Du, also Mark Johnson

- A new hierarchical Bayesian model for unsupervised topic segmentation, using Bayesian segmentation + structured topic modelling.
- A novel sampling algorithm for splitting/merging restaurant(s) in CRP.
Conclusion of Segmentation with a Structured Topic Model

Paper given at NAACL 2013, main author Lan Du, also Mark Johnson

- A new hierarchical Bayesian model for unsupervised topic segmentation, using Bayesian segmentation + structured topic modelling.
- A novel sampling algorithm for splitting/merging restaurant(s) in CRP.
- Code is available at Lan Du’s website.
- Now running multi-core.
Fun with Bibliographies (Lim et al/ ACML 2014)
Use a sentiment dictionary as a prior to learn aspect specific sentiment words from tweets.
Conclusion

- Latent Semantic Modelling with non-parametric Bayesians methods!
- Hierarchical stick-breaking and Chinese restaurant process methods seem inferior to block table indicator sampling.
- See the individual papers.
- Read my blog and tutorials
  
  https://topicmodels.org
  “A tutorial on non-parametric methods”

Thank You ... Questions?
Alphabetic References


Alphabetic References