Generative Latent Variable Models of Text

Jacob Eisenstein

Machine Learning Department, CMU

November 16, 2011
Generative models of text

Generative models are a powerful tool for understanding document collections.

- Classification/clustering (Naive Bayes)
- Discover latent themes (LDA)
- Distinguish latent and observed factors (e.g. Topic-aspect models)
Generative models of text

Generative models are a powerful tool for understanding document collections.

- Classification/clustering (Naive Bayes)
- Discover latent themes (LDA)
- Distinguish latent and observed factors (e.g. Topic-aspect models)

**Unifying idea**: a probability model over text, $P(w|z)$, where $z$ are labels or latent variables
Naive Bayes is a generative model for classification:

\[
\log P(w^{(d)} | z^{(d)}, \beta) = \prod_{n} P(w_n^{(d)} | \beta, z_n^{(d)}) \\
= \prod_{n} \beta_{z_n^{(d)}, w_n^{(d)}}
\]
Naive Bayes is a generative model for classification:

\[
\log P(w^{(d)}|z^{(d)}, \beta) = \prod_{n} P(w_n^{(d)}|\beta, z_n^{(d)})
\]

\[
= \prod_{n} \beta_{z_n^{(d)}, w_n^{(d)}}
\]

- **training:**

\[
\hat{\beta} = \arg \max_{\beta} \prod_{d} P(w^{(d)}|z^{(d)}, \beta)
\]

- **prediction:**

\[
\hat{z}^{(d)} = \arg \max_{y} P(w^{(d)}|z, \beta)
\]
The Dirichlet-Multinomial pair

- Each $\beta_i$ is a distribution over words, typically a multinomial distribution.
The Dirichlet-Multinomial pair

- Each $\beta_i$ is a distribution over words, typically a multinomial distribution.
- If we want to “be Bayesian,” we can place a prior distribution on $\beta$. Then we are solving,

$$\hat{\beta} = \arg \max_{\beta} \prod_d P(w^{(d)}|z^{(d)}, \beta)P(\beta)$$
The Dirichlet-Multinomial pair

- Each $\beta_i$ is a distribution over words, typically a **multinomial** distribution.
- If we want to “be Bayesian,” we can place a prior distribution on $\beta$. Then we are solving,

$$
\hat{\beta} = \arg \max_{\beta} \prod_d P(w^{(d)}|z^{(d)}, \beta) P(\beta)
$$

- The conjugate prior for the multinomial is the **Dirichlet** distribution.

Conjugacy means we can do collapsed Gibbs sampling, analytically marginalizing the parameter $\beta$. This trick gets used a lot.
An aside

- Using priors (or not) is a key tenet of some people’s world view!
An aside

- Using priors (or not) is a key tenet of some people’s world view!
- But there are also practical reasons to use priors.
Using priors (or not) is a key tenet of some people’s world view!

But there are also practical reasons to use priors.

- They perform smoothing, improving performance when data is limited or the number of parameters is very large.
Using priors (or not) is a key tenet of some people’s world view!
But there are also practical reasons to use priors.
- They perform smoothing, improving performance when data is limited or the number of parameters is very large.
- Priors also make it possible to incorporate domain knowledge.

Spoiler: I’ll have a lot more to say about whether the Dirichlet-Multinomial pair is the best possible choice for generative models.
Using priors (or not) is a key tenet of some people’s world view!

But there are also practical reasons to use priors.
- They perform smoothing, improving performance when data is limited or the number of parameters is very large.
- Priors also make it possible to incorporate domain knowledge.

Spoiler: I’ll have a lot more to say about whether the Dirichlet-Multinomial pair is the best possible choice for generative models.
Naive Bayes

\[
\log P(w^{(d)}|z^{(d)}, \beta) = \prod_n P(w_n^{(d)}|\beta, z_n^{(d)}) = \prod_n \beta_{z_n^{(d)}, w_n^{(d)}}
\]

**training:**

\[
\hat{\beta} = \arg \max_\beta \prod_d P(w^{(d)}|z^{(d)}, \beta)
\]

**prediction:**

\[
\hat{z}^{(d)} = \arg \max_y P(w^{(d)}|z, \beta)
\]
Example: Political ideology classification on Twitter

Training data:

Messages containing #p2

Messages containing #tcot
Example: Political ideology classification on Twitter

Training data:

**Messages containing #p2**

- ![Image 1](image1)
- ![Image 2](image2)
- ![Image 3](image3)
- ![Image 4](image4)

- $\beta_{#p2}$ emphasizes *protest, unconstitutional, fascism*

**Messages containing #tcot**

- ![Image 5](image5)
- ![Image 6](image6)
- ![Image 7](image7)
- ![Image 8](image8)

- $\beta_{#tcot}$ emphasizes *nobama, solyndra, socialism*
Lin et al (2006) applied Naive Bayes to the “bitter lemons” corpus of text about the Palestinian-Israeli conflict:

<table>
<thead>
<tr>
<th>Model</th>
<th>Data Set</th>
<th>Accuracy</th>
<th>Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td></td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>Editors</td>
<td>0.9724</td>
<td></td>
</tr>
<tr>
<td>NB-M</td>
<td>Editors</td>
<td>0.9895</td>
<td>61%</td>
</tr>
<tr>
<td>NB-B</td>
<td>Editors</td>
<td>0.9909</td>
<td>67%</td>
</tr>
<tr>
<td>SVM</td>
<td>Guests</td>
<td>0.8621</td>
<td></td>
</tr>
<tr>
<td>NB-M</td>
<td>Guests</td>
<td>0.8789</td>
<td>12%</td>
</tr>
<tr>
<td>NB-B</td>
<td>Guests</td>
<td>0.8859</td>
<td>17%</td>
</tr>
</tbody>
</table>

Palestinian: palestinian, israel, state, politics, peace, international, people, settle, occupation, sharon, right, govern, two, secure, end, conflict, process, side, negotiate

Israeli: israel, palestinian, state, settle, sharon, peace, arafat, arab, politics, two, process, secure, conflict, lead, america, agree, right, gaza, govern
When the label $z$ is not observed, it can be imputed. This is a method for probabilistic clustering:

$$P(w|\theta, \beta) = \sum_z P(z|\theta) \prod_n P(w_n|\beta_z)$$

where $\theta$ is a prior on $z$. 
Unsupervised Naive Bayes

When the label $z$ is not observed, it can be imputed. This is a method for probabilistic clustering:

$$P(w | \theta, \beta) = \sum_z P(z | \theta) \prod_n P(w_n | \beta_z)$$

where $\theta$ is a prior on $z$.

Typically we optimize using expectation-maximization:

- In the **e-step** we compute the distribution $Q(z)$
- In the **m-step** we update the parameter $\beta$
Latent Variable Models

- Imagine we have additional data $y^{(d)}$: for each author on Twitter,
  - $y^{(d)}$ is their geographical location,
  - $w^{(d)}$ is the set of all words in all their tweets,
  - $z^{(d)}$ is a latent variable which must explain both $y^{(d)}$ and $w^{(d)}$.

- We want to learn to predict $y$ from $w$.
  (Eisenstein, O’Connor, Smith, and Xing. EMNLP 2010)
Imagine we have additional data $y^{(d)}$: for each author on Twitter,
- $y^{(d)}$ is their geographical location,
- $w^{(d)}$ is the set of all words in all their tweets,
- $z^{(d)}$ is a latent variable which must explain both $y^{(d)}$ and $w^{(d)}$.

We want to learn to predict $y$ from $w$. (Eisenstein, O’Connor, Smith, and Xing. EMNLP 2010)
Imagine we have additional data $y^{(d)}$: for each author on Twitter,

- $y^{(d)}$ is their geographical location,
- $w^{(d)}$ is the set of all words in all their tweets,
- $z^{(d)}$ is a latent variable which must explain both $y^{(d)}$ and $w^{(d)}$.

We want to learn to predict $y$ from $w$. (Eisenstein, O’Connor, Smith, and Xing. EMNLP 2010)

In training, we maximize:

$$P(y, w | \theta, \beta, \mu, \sigma^2) = \sum_z P(z | \theta) P(y | \mu_z, \sigma^2_z) \prod_n P(w_n | \beta_z)$$
Latent Variable Models

**training**: Expectation-maximization, alternating between updates to $Q(z)$ and the parameters $\{\beta, \theta, \mu, \sigma^2\}$
**training**: Expectation-maximization, alternating between updates to $Q(z)$ and the parameters $\{\beta, \theta, \mu, \sigma^2\}$

**prediction**:

$$\hat{y} = \arg \max_y P(y|w)$$

$$P(y|w) = \sum_z P(y|\mu_z, \sigma^2_z)P(z|w, \theta)$$

$$P(z|w, \theta) = P(w|\beta_z)P(z|\theta)/P(w)$$
Quantitative Results

<table>
<thead>
<tr>
<th>error in km:</th>
<th>mean</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean location</td>
<td>1148</td>
<td>1018</td>
</tr>
<tr>
<td>text regression</td>
<td>948</td>
<td>712</td>
</tr>
<tr>
<td>mixture model</td>
<td>947</td>
<td>644</td>
</tr>
</tbody>
</table>
Each author in our dataset is a point; cluster membership is indicated by color and shape.\(^1\)

\(^1\)Figure by Brendan O’Connor
Qualitative results

For each cluster, we rank words by log-odds: $\log \beta_i - \log \frac{1}{K} \sum_j \beta_j$:

- **New York**: brib, lml, wassupp, uu, werd, deadass, flatbush, odee, dha
- **So. Cal**: disneyland, cuh, fucken, af, fasho, faded, wyd, freeway, bomb
- **No. Cal**: sac, oakland, sf, hella, warriors, pleasure, bay, koo
- **Atlanta**: atlanta, atl, georgia, ga, $1$, waffle, af, nun, shawty
- **Cleveland/Detroit**: ctfu, detroit, foolin, .!!., cleveland, geeked, salty, ikr
- **Pac. Northwest**: seattle, portland, oregon, olympic, heh, canada, stoked
Discovering latent themes

Topic models like latent Dirichlet allocation discover latent themes or topics in document collections:

- Each $\beta_k$ is a topic, a distribution over words.
- Each $\theta_d$ represents the topic proportions for document $d$.
- Each $z_n$ is the latent topic which generates the word $w_n$.

Mathematically:

$$P(w | \theta, \beta) = \prod_{n} P(z_n | \theta) P(w_n | \beta_{z_n})$$
Key point is that individual authors are *admixtures* of these topics, e.g., my Twitter feed is 60% chit-chat, 30% basketball, 10% emoticons.
Combining topics and labels

Recall the Twitter political ideology problem:

**Messages containing #p2**

- **LCranston1939** (LaMont Cranston)
  Arrant Fascism: Coordinated #OWS evictions make this clear; the military and the police exist to protect the 1%.
  bit.ly/tpsn5m #p2
  1 hour ago

- **rtherealitycheck** RH Reality Check
  Thanks to @BarbaraBCrane (of @IpasOrg) for donating & helping us stop the right-wing effort to #OccupyYourWomb! ow.ly/7ueej #p2...
  2 hours ago

- **BuddyRoemer** (Gov. Buddy Roemer)
  Bloomberg’s actions in the midnight hours against Occupy Wall Street protestors were unjust, uncalled for, and unconstitutional.
  #p2 #ows
  3 hours ago

- **peterothberg** Peter Rothberg
  thenation.com/blog/164612/occ...
  #ows #p2
  3 hours ago

- **miffMedia Matters**
  Don’t worry, folks: #FoxNews has enough conspiracy theorists to go around for anyone willing to believe them!
  bit.ly/tutOfhM #p2
  3 hours ago

**Messages containing #tcot**

- **AG_Conservative**
  I miss having a president who loves America and Americans. RT if you agree.
  #tcot
  13 Nov
  Retweeted 100+ times

- **SaintRPh** Matt Dawson
  Febreeze should film a commercial at Zuccotti Park.
  #ows #tcot
  11 minutes ago

- **PatDollard** Patrick Dollard
  Breaking: JUDGE RULES #OCCUPYWALLSTREET CAMP CANNOT RETURN TO ZUCCOTTI PARK
  bit.ly/uXymp7 #tcot
  16 minutes ago
  Favorite 6 Retweet 6 Reply

- **PatDollard** Patrick Dollard
  Team Obama Pressured Solyndra To Hide Layoffs Until After Elections bit.ly/wveqaK #tcot
  16 minutes ago

- **FoxieNews** Debbie
  Dear #Occupy Protesters, You can’t have your Sleepovers at Zuccott Park anymore. Go Home and Please Occupy a Shower and a Job STAT! #tcot
  18 minutes ago
Authors don’t just express ideological viewpoints, they discuss topics: health care, taxes, regulation, ...
Authors don’t just express ideological viewpoints, they discuss topics: health care, taxes, regulation, ...

In prediction, these topical differences make learning harder. Left-wing and right-wing perspectives on a single topic may share more words than a single perspective on multiple topics.
Authors don’t just express ideological viewpoints, they discuss topics: health care, taxes, regulation, ...

In prediction, these topical differences make learning harder. Left-wing and right-wing perspectives on a single topic may share more words than a single perspective on multiple topics.

In analysis, we often want to understand topic-specific differences: e.g., how do the left-wing and right-wing perspectives differ with respect to foreign policy
Switching models

We can combine topics and labels by adding a “switch” for each word, which determines if the word is generated from a topic or the label:

- Each $s_n$ determines whether $w_n$ is generated by the topic $z_n$ or the label $y$.
- Each $\beta_k^{(T)}$ is a word distribution associated with a latent topic.
- Each $\beta_j^{(A)}$ is a word distribution associated with a label.
Switching models

We can combine topics and labels by adding a “switch” for each word, which determines if the word is generated from a topic or the label:

- Each $s_n$ determines whether $w_n$ is generated by the topic $z_n$ or the label $y$.
- Each $\beta_k^{(T)}$ is a word distribution associated a latent topic.
- Each $\beta_j^{(A)}$ is a word distribution associated with a label.
- Each $\beta_{k,j}^{(TA)}$ is a word distribution associated with a topic-label interaction.
Switching models: a schematic

A topic-perspective-background model:
<table>
<thead>
<tr>
<th>Topic 1</th>
<th>Topic 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>fashion</td>
<td>food</td>
</tr>
<tr>
<td>style</td>
<td>add</td>
</tr>
<tr>
<td>look</td>
<td>chicken</td>
</tr>
<tr>
<td>dress</td>
<td>recipe</td>
</tr>
<tr>
<td>wear</td>
<td>cooking</td>
</tr>
<tr>
<td>new</td>
<td>taste</td>
</tr>
<tr>
<td>collection</td>
<td>rice</td>
</tr>
<tr>
<td>accessories</td>
<td>recipes</td>
</tr>
<tr>
<td>black</td>
<td>sugar</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UK</th>
<th>India</th>
<th>Singapore</th>
</tr>
</thead>
<tbody>
<tr>
<td>shoes</td>
<td>fashion</td>
<td>price</td>
</tr>
<tr>
<td>fashion</td>
<td>women</td>
<td>posted</td>
</tr>
<tr>
<td>clothing</td>
<td>indian</td>
<td>earrings</td>
</tr>
<tr>
<td>high</td>
<td>designer</td>
<td>length</td>
</tr>
<tr>
<td>designer</td>
<td>sarees</td>
<td>item</td>
</tr>
<tr>
<td>style</td>
<td>leather</td>
<td>sgd</td>
</tr>
<tr>
<td>love</td>
<td>girls</td>
<td>silver</td>
</tr>
<tr>
<td>london</td>
<td>china</td>
<td>clothes</td>
</tr>
<tr>
<td>shirts</td>
<td>jewellery</td>
<td>shop</td>
</tr>
<tr>
<td>bag</td>
<td>jewelry</td>
<td>code</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UK</th>
<th>India</th>
<th>Singapore</th>
</tr>
</thead>
<tbody>
<tr>
<td>food</td>
<td>recipe</td>
<td>coffee</td>
</tr>
<tr>
<td>wine</td>
<td>recipes</td>
<td>cup</td>
</tr>
<tr>
<td>restaurant</td>
<td>powder</td>
<td>oil</td>
</tr>
<tr>
<td>coffee</td>
<td>indian</td>
<td>comments</td>
</tr>
<tr>
<td>cheese</td>
<td>salt</td>
<td>fried</td>
</tr>
<tr>
<td>soup</td>
<td>tsp</td>
<td>add</td>
</tr>
<tr>
<td>eat</td>
<td>rice</td>
<td>restaurant</td>
</tr>
<tr>
<td>chef</td>
<td>masala</td>
<td>rice</td>
</tr>
<tr>
<td>English</td>
<td>oil</td>
<td>tea</td>
</tr>
<tr>
<td>drink</td>
<td>coriander</td>
<td>seafood</td>
</tr>
</tbody>
</table>

From ccLDA (Paul and Girju, 2009)
### Example output: topics and perspectives

<table>
<thead>
<tr>
<th>Aspect A</th>
<th>Aspect B</th>
</tr>
</thead>
<tbody>
<tr>
<td>palestinian</td>
<td>war</td>
</tr>
<tr>
<td>israeli</td>
<td>violence</td>
</tr>
<tr>
<td>israel</td>
<td>palestinians</td>
</tr>
<tr>
<td>military</td>
<td>occupation</td>
</tr>
<tr>
<td>civilians</td>
<td>resistance</td>
</tr>
<tr>
<td>attacks</td>
<td>intifada</td>
</tr>
<tr>
<td></td>
<td>violent</td>
</tr>
<tr>
<td></td>
<td>non</td>
</tr>
<tr>
<td></td>
<td>force</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Israeli</th>
<th>Palestinian</th>
</tr>
</thead>
<tbody>
<tr>
<td>jewish</td>
<td>palestinians</td>
</tr>
<tr>
<td>arab</td>
<td>return</td>
</tr>
<tr>
<td>israeli</td>
<td>right</td>
</tr>
<tr>
<td>jews</td>
<td>refugees</td>
</tr>
<tr>
<td>population</td>
<td>problem</td>
</tr>
<tr>
<td>jordan</td>
<td>refugee</td>
</tr>
<tr>
<td>west</td>
<td>rights</td>
</tr>
<tr>
<td>south</td>
<td>resolution</td>
</tr>
</tbody>
</table>

From TAM (Paul and Girju, 2010); added unsupervised and semi-supervised learning to ccLDA.
Results: ideology prediction

From Multiview-LDA (Ahmed and Xing, 2010)
## Results: geography prediction

<table>
<thead>
<tr>
<th>error in km:</th>
<th>mean</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean location</td>
<td>1148</td>
<td>1018</td>
</tr>
<tr>
<td>text regression</td>
<td>948</td>
<td>712</td>
</tr>
<tr>
<td>mixture model</td>
<td>947</td>
<td>644</td>
</tr>
<tr>
<td><strong>mixture model + topics</strong></td>
<td>900</td>
<td>494</td>
</tr>
</tbody>
</table>
Capabilities of generative models:

- Classification and clustering (Naive Bayes)
- Discovering latent topics (LDA)
- Combining topics and labels (ccLDA, TAM, Multiview-LDA)
Capabilities of generative models:

- Classification and clustering (Naive Bayes)
- Discovering latent topics (LDA)
- Combining topics and labels (ccLDA, TAM, Multiview-LDA)

We have focused on text, but there are many, many applications of these models to vision and computational biology.
Taking stock

Generative models models have many advantages:

- Interpretability
- Can combine multiple modalities
- Relatively simple semi-supervised extensions
- Easy to incorporate domain-specific insights in model design
Taking stock

Generative models have many advantages:

- Interpretability
- Can combine multiple modalities
- Relatively simple semi-supervised extensions
- Easy to incorporate domain-specific insights in model design

But they also have problems! (Eisenstein et al., ICML 2011)
A naïve Bayes classifier must estimate the parameter $Pr(w = \text{“the”}|y)$ for every class $y$. 
A naïve Bayes classifier must estimate the parameter $Pr(w = “the” | y)$ for every class $y$.

The probability $Pr(w = “the”)$ is a fact about English, not about any of the classes (usually).
A naïve Bayes classifier must estimate the parameter $Pr(w = “the” | y)$ for every class $y$.

The probability $Pr(w = “the”)$ is a fact about English, not about any of the classes (usually).

Heuristic solutions like stopword pruning are hard to generalize to new domains.
A naïve Bayes classifier must estimate the parameter $Pr(w = \text{“the”} | y)$ for every class $y$.

The probability $Pr(w = \text{“the”})$ is a fact about English, not about any of the classes (usually).

Heuristic solutions like stopword pruning are hard to generalize to new domains.

It would be better to focus computation on parameters that distinguish the classes.
Overparametrization

- An LDA **model** with $K$ topics and $V$ words requires $K \times V$ parameters.
- An LDA **paper** shows 10 words per topic.
An LDA model with $K$ topics and $V$ words requires $K \times V$ parameters.

An LDA paper shows 10 words per topic.

What about the other $V - 10$ words per topic??
An LDA model with $K$ topics and $V$ words requires $K \times V$ parameters.

An LDA paper shows 10 words per topic.

What about the other $V - 10$ words per topic??

- These parameters affect the assignment of documents...
- But they may be unnoticed by the user.
- And there may not be enough data to estimate them accurately.
Latent topics may be combined with additional facets, such as sentiment and author perspective.

“Switching” variables decide if a word is drawn from a topic or from another facet.

Twice as many latent variables per document!
**Multinomial generative models**: each class or latent theme is represented by a distribution over tokens, $P(w|y) = \beta_y$.
**Multinomial generative models**: each class or latent theme is represented by a distribution over tokens, $P(w|y) = \beta_y$

**Sparse Additive Generative models (SAGE)**: each class or latent theme is represented by its deviation from a background distribution.

$$P(w|y,m) \propto \exp (m + \eta_y)$$
**Multinomial generative models**: each class or latent theme is represented by a distribution over tokens, \( P(w|y) = \beta_y \)

**Sparse Additive Generative models (SAGE)**: each class or latent theme is represented by its deviation from a background distribution.

\[
P(w|y, m) \propto \exp(m + \eta_y)
\]

- \( m \) captures the background word log-probabilities
- \( \eta \) contains **sparse** deviations for each topic or class
- additional facets can be added in log-space
A topic-perspective-background model using Dirichlet-multinomials:
Sparse Additive Generative Models

A topic-perspective-background model using SAGE:
Sparse Additive Generative Models

A topic-perspective-background model using SAGE:
Sparsity deviation of log probabilities

- Sparsity: $\eta_i = 0$ for many $i$
Sparsity deviation of log probabilities

- Sparsity: $\eta_i = 0$ for many $i$
- Due to normalization, the generative probabilities will not be identical, $Pr(w = i | \eta + m) \neq Pr(w = i | m)$, even if $\eta_i = 0$. Different notion of sparsity from sparseTM (Wang & Blei, 2009), which sets $Pr(w = i | y) = 0$ for many $i$. 
Sparsity deviation of log probabilities

- Sparsity: $\eta_i = 0$ for many $i$
- Due to normalization, the generative probabilities will not be identical, $Pr(w = i | \eta + m) \neq Pr(w = i | m)$, even if $\eta_i = 0$.
- But for most pairs of words, $\frac{Pr(w = i | \eta + m)}{Pr(w = j | \eta + m)} = \frac{Pr(w = i | m)}{Pr(w = j | m)}$
Sparsity deviation of log probabilities

- Sparsity: $\eta_i = 0$ for many $i$
- Due to normalization, the generative probabilities will not be identical, $Pr(w = i|\eta + m) \neq Pr(w = i|m)$, even if $\eta_i = 0$.
- But for most pairs of words,  
  \[
  \frac{Pr(w=i|\eta+m)}{Pr(w=j|\eta+m)} = \frac{Pr(w=i|m)}{Pr(w=j|m)}
  \]
Sparsity deviation of log probabilities

- Sparsity: \( \eta_i = 0 \) for many \( i \)
- Due to normalization, the generative probabilities will not be identical, \( Pr(w = i | \eta + m) \neq Pr(w = i | m) \), even if \( \eta_i = 0 \).
- But for most pairs of words, \( \frac{Pr(w = i | \eta + m)}{Pr(w = j | \eta + m)} = \frac{Pr(w = i | m)}{Pr(w = j | m)} \)

Different notion of sparsity from sparseTM (Wang & Blei, 2009), which sets \( Pr(w = i | y) = 0 \) for many \( i \).
The $L_1$ regularizer is equivalent to a Laplace prior distribution:

$$\eta \sim \mathcal{L}(0, \sigma)$$
The $L_1$ regularizer is equivalent to a Laplace prior distribution:
\[ \eta \sim \mathcal{L}(0, \sigma) \]

The Laplace distribution is equal to the integral:
\[ \mathcal{L}(\eta; 0, \sigma) = \int \mathcal{N}(\eta; 0, \tau)\text{Exp}(\tau; \sigma)\,d\tau \]  
(Lange & Simsheimer, 1993)
The $L1$ regularizer is equivalent to a Laplace prior distribution: 
\[ \eta \sim \mathcal{L}(0, \sigma) \]

The Laplace distribution is equal to the integral: 
\[ \mathcal{L}(\eta; 0, \sigma) = \int \mathcal{N}(\eta; 0, \tau) \exp(\tau; \sigma) d\tau \quad \text{(Lange & Simsheimer, 1993)} \]

Other integrals also induce sparsity, e.g. 
\[ \int \mathcal{N}(\eta; 0, \tau) \frac{1}{\tau} d\tau \quad \text{(Figueiredo, 2001; Guan & Dy, 2009)} \]
The $L_1$ regularizer is equivalent to a Laplace prior distribution: 
\[ \eta \sim L(0, \sigma) \]

The Laplace distribution is equal to the integral: 
\[ L(\eta; 0, \sigma) = \int N(\eta; 0, \tau) \text{Exp}(\tau; \sigma) d\tau \]  
(Lange & Simsheimer, 1993)

Other integrals also induce sparsity, e.g. 
\[ \int N(\eta; 0, \tau) \frac{1}{\tau} d\tau \]  
(Figueiredo, 2001; Guan & Dy, 2009)

We solve this integral through coordinate ascent (EM), updating:
Sparsity through integration

- The $L_1$ regularizer is equivalent to a Laplace prior distribution:
  \[ \eta \sim \mathcal{L}(0, \sigma) \]

- The Laplace distribution is equal to the integral:
  \[ \mathcal{L}(\eta; 0, \sigma) = \int \mathcal{N}(\eta; 0, \tau) \exp(\tau; \sigma) d\tau \quad \text{(Lange & Simsheimer, 1993)} \]

- Other integrals also induce sparsity, e.g.
  \[ \int \mathcal{N}(\eta; 0, \tau) \frac{1}{\tau} d\tau \quad \text{(Figueiredo, 2001; Guan & Dy, 2009)} \]

- We solve this integral through coordinate ascent (EM), updating:
  - The distribution $Q(\tau)$
Sparsity through integration

- The $L_1$ regularizer is equivalent to a Laplace prior distribution: \( \eta \sim \mathcal{L}(0, \sigma) \)

  - The Laplace distribution is equal to the integral:
  \[
  \mathcal{L}(\eta; 0, \sigma) = \int \mathcal{N}(\eta; 0, \tau)\text{Exp}(\tau; \sigma) d\tau \quad (\text{Lange & Simsheimer, 1993})
  \]

  - Other integrals also induce sparsity, e.g.
  \[
  \int \mathcal{N}(\eta; 0, \tau) \frac{1}{\tau} d\tau \quad (\text{Figueiredo, 2001; Guan & Dy, 2009})
  \]

- We solve this integral through coordinate ascent (EM), updating:
  - The distribution $Q(\tau)$
  - A point estimate of $\eta$
Applications

- Document classification
- Topic models
- Multifaceted topic models
Each document $d$ has a label $y_d$.

Each token $w_{d,n}$ is drawn from a multinomial distribution $\beta$, where

$$\beta_i = \frac{\exp(\eta_{y_d,i} + m_i)}{\sum_j \exp(\eta_{y_d,j} + m_j)}$$

Each parameter $\eta_{k,i}$ is drawn from a distribution equal to $\mathcal{N}(0, \tau_{k,i})$, with $P(\tau_{k,i}) \sim 1/\tau_{k,i}$.
We maximize the variational bound

\[
\ell = \sum_{d} \sum_{n} \log P(w_{n}^{(d)}|m, \eta_{y_{d}}) + \sum_{k} \langle \log P(\eta_{k}|0, \tau_{k}) \rangle \\
+ \sum_{k} \langle \log P(\tau_{k}|\gamma) \rangle - \sum_{k} \langle \log Q(\tau_{k}) \rangle,
\]

where \( c_{k} \) are the observed counts for class \( k \),

\( C_{k} = \sum_{i} c_{ki} \) \( \beta_{k} \propto \exp(\eta_{k} + m) \).
Inference

- We maximize the variational bound

\[
\ell = \sum_d \sum_n^{N_d} \log P(w_n^{(d)} | m, \eta_{yd}) + \sum_k \langle \log P(\eta_k | 0, \tau_k) \rangle \\
+ \sum_k \langle \log P(\tau_k | \gamma) \rangle - \sum_k \langle \log Q(\tau_k) \rangle,
\]

- The gradient wrt \( \eta \) is,

\[
\frac{\partial \ell}{\partial \eta_k} = c_k - C_k \beta_k - \text{diag} \left( \langle \tau_k^{-1} \rangle \right) \eta_k,
\]

where

- \( c_k \) are the observed counts for class \( k \)
- \( C_k = \sum_i c_{ki} \)
- \( \beta_k \propto \exp(\eta_k + m) \)
We maximize the variational bound

\[ \ell = \sum_{d} \sum_{n} \log P(w_n^{(d)} | m, \eta_{y_d}) + \sum_{k} \langle \log P(\eta_k | 0, \tau_k) \rangle \\
+ \sum_{k} \langle \log P(\tau_k | \gamma) \rangle - \sum_{k} \langle \log Q(\tau_k) \rangle, \]
Inference

- We maximize the variational bound

\[
\ell = \sum_d \sum_n \log P(w_n^{(d)}|m, \eta_{y_d}) + \sum_k \langle \log P(\eta_k|0, \tau_k) \rangle \\
+ \sum_k \langle \log P(\tau_k|\gamma) \rangle - \sum_k \langle \log Q(\tau_k) \rangle,
\]

- We choose \( Q(\tau_{k,i}) = \text{Gamma}(\tau_{k,i}; a_{k,i}, b_{k,i}) \)
Inference

- We maximize the variational bound

\[ \ell = \sum_{d} \sum_{n} \log P(w_{n}^{(d)} | m, \eta_{y_{d}}) + \sum_{k} \langle \log P(\eta_{k} | 0, \tau_{k}) \rangle \]

\[ + \sum_{k} \langle \log P(\tau_{k} | \gamma) \rangle - \sum_{k} \langle \log Q(\tau_{k}) \rangle, \]

- We choose \( Q(\tau_{k}, i) = \text{Gamma}(\tau_{k}, i; a_{k}, i, b_{k}, i) \)

- Iterate between a Newton update to \( a \) and a closed-form update to \( b \)
Document classification evaluation

- 20 newsgroups data: 11K training docs, 50K vocab

![Graph showing accuracy vs. proportion of training set]

- Adaptive sparsity:
  - 10% non-zeros for full training set (11K docs)
  - 2% non-zeros for minimal training set (550 docs)
Document classification evaluation

- 20 newsgroups data: 11K training docs, 50K vocab

- Adaptive sparsity:
  - 10% non-zeros for full training set (11K docs)
  - 2% non-zeros for minimal training set (550 docs)
The gradient for $\eta_k$ now includes expected counts:

$$\frac{\partial \ell}{\partial \eta_k} = \langle c_{k,i} \rangle - \langle C_k \rangle \beta_k - \text{diag}(\langle \tau_{k,i} - 1 \rangle)\eta_k,$$

where

$$\langle c_{k,i} \rangle = \sum_n Q(z_n | k) \delta(w_n = i).$$
The gradient for $\eta$ now includes expected counts:

$$\frac{\partial \ell}{\partial \eta_k} = \langle c_k \rangle - \langle C_k \rangle \beta_k - \text{diag}(\langle \tau_k^{-1} \rangle) \eta_k,$$

where $\langle c_{ki} \rangle = \sum_n Q_{zn}(k) \delta(w_n = i)$. 

The gradient for $\eta$ now includes **expected** counts:
Sparse topic model results

- NIPS dataset: 1986 training docs, 10K vocabulary

- Number of topics vs. perplexity

- System: Dirichlet-Multinomial, SAGE

- Adaptive sparsity:
  - 5% non-zeros for 10 topics
  - 1% non-zeros for 50 topics
Sparse topic model results

- **NIPS dataset:** 1986 training docs, 10K vocabulary

  - **Number of topics vs. perplexity**
    - Bar chart showing perplexity for different number of topics (10, 25, 50)
    - Systems compared: Dirichlet–Multinomial, SAGE
    - Lower perplexity indicates better performance

- **Adaptive sparsity:**
  - 5% non-zeros for 10 topics
  - 1% non-zeros for 50 topics
Sparse topic model analysis

Total variation $= \sum_i |\beta_{k,i} - \bar{\beta}_i|$

Standard topic models assign the greatest amount of variation for the probabilities of the words with the least evidence!
Multifaceted generative models

- Combines latent topics $\beta^{(T)}$ with other facets $\beta^{(A)}$, e.g. ideology, dialect, sentiment
Multifaceted generative models

- Combines latent topics $\beta^{(T)}$ with other facets $\beta^{(A)}$, e.g. ideology, dialect, sentiment
- Typically, a **switching variable** determines which generative facet produces each token (Paul & Girju, 2010; Ahmed & Xing, 2010).
Multifaceted generative models

- Combines latent topics $\beta^{(T)}$ with other facets $\beta^{(A)}$, e.g. ideology, dialect, sentiment
- Typically, a **switching variable** determines which generative facet produces each token (Paul & Girju, 2010; Ahmed & Xing, 2010).
- There is one switching variable per token, complicating inference.
In SAGE, switching variables are not needed.

\[ P(w|z, y) \propto \exp(\eta(T)z + \eta(A)y + m) \]

The gradient for \( \eta(T) \) is now
\[
\frac{\partial \ell}{\partial \eta(T)_k} = \langle c(T)_k \rangle - \sum_j \langle C_{jk} \rangle \beta_{jk} - \text{diag}(\langle \tau - 1_k \rangle) \eta_k, \theta_{z\eta(T)_k \tau(T)_k}, \theta_{w\eta(A)_j \tau(A)_j}, \theta_{y m_i}, \theta_{x \alpha} \rangle
\]

\( i \in \{1, \ldots, W\} \)
In SAGE, switching variables are not needed.

Instead, we just sum all the facets in log-space:

\[ P(w|z, y) \propto \exp(\eta_T z + \eta_A y + m) \]
In SAGE, switching variables are not needed.

Instead, we just sum all the facets in log-space:

\[
P(w|z, y) \propto \exp \left( \eta^{(T)}_z + \eta^{(A)}_y + m \right)
\]

The gradient for \( \eta^{(T)} \) is now

\[
\frac{\partial \ell}{\partial \eta^{(T)}_k} = \langle c^{(T)}_k \rangle - \sum_j \langle C_{jk} \rangle \beta_{jk} - \text{diag} \left( \langle \tau^{-1}_k \rangle \right) \eta_k,
\]
Evaluation: Ideology prediction

- Task: predict blog ideology
- Model: latent topics, observed ideology labels
- Data: six blogs total (two held out), 21K documents, 5.1M tokens

Results match previous best of 69% for Multiview LDA and support vector machine (Ahmed & Xing, 2010).
Evaluation: Ideology prediction

- Task: predict blog ideology
- Model: latent topics, observed ideology labels
- Data: six blogs total (two held out), 21K documents, 5.1M tokens

Results match previous best of 69% for Multiview LDA and support vector machine (Ahmed & Xing, 2010).
Evaluation: Geographical Topic Model

- **Task**: location prediction from Twitter text
- **Model**: latent “region” generates text and locations
- **9800 weeklong twitter transcripts; 380K messages; 4.9M tokens**
Evaluation: Geographical Topic Model

- Task: location prediction from Twitter text
- Model: latent “region” generates text and locations
- 9800 weeklong twitter transcripts; 380K messages; 4.9M tokens

<table>
<thead>
<tr>
<th>error in km:</th>
<th>mean</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean location</td>
<td>1148</td>
<td>1018</td>
</tr>
<tr>
<td>text regression</td>
<td>948</td>
<td>712</td>
</tr>
<tr>
<td>mixture model</td>
<td>947</td>
<td>644</td>
</tr>
<tr>
<td>mixture model + topics</td>
<td>900</td>
<td>494</td>
</tr>
</tbody>
</table>

SAGE (5K vocab) 845 501
SAGE (22K vocab) 791 461
Evaluation: Geographical Topic Model

- **Task:** location prediction from Twitter text
- **Model:** latent “region” generates text and locations
- **9800 weeklong twitter transcripts; 380K messages; 4.9M tokens**

<table>
<thead>
<tr>
<th>error in km:</th>
<th>mean</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean location</td>
<td>1148</td>
<td>1018</td>
</tr>
<tr>
<td>text regression</td>
<td>948</td>
<td>712</td>
</tr>
<tr>
<td>mixture model</td>
<td>947</td>
<td>644</td>
</tr>
<tr>
<td>mixture model + topics</td>
<td>900</td>
<td>494</td>
</tr>
<tr>
<td>SAGE (5K vocab)</td>
<td>845</td>
<td>501</td>
</tr>
<tr>
<td>SAGE (22K vocab)</td>
<td>791</td>
<td>461</td>
</tr>
</tbody>
</table>
**Evaluation: Geographical Topic Model**

- **Task**: location prediction from Twitter text
- **Model**: latent “region” generates text and locations
- **9800 weeklong twitter transcripts; 380K messages; 4.9M tokens**

<table>
<thead>
<tr>
<th>error in km:</th>
<th>mean</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean location</td>
<td>1148</td>
<td>1018</td>
</tr>
<tr>
<td>text regression</td>
<td>948</td>
<td>712</td>
</tr>
<tr>
<td>mixture model</td>
<td>947</td>
<td>644</td>
</tr>
<tr>
<td>mixture model + topics</td>
<td>900</td>
<td>494</td>
</tr>
<tr>
<td>SAGE (5K vocab)</td>
<td>845</td>
<td>501</td>
</tr>
<tr>
<td>SAGE (22K vocab)</td>
<td><strong>791</strong></td>
<td><strong>461</strong></td>
</tr>
</tbody>
</table>
The Dirichlet-multinomial pair is computationally convenient, but does not adequately control model complexity.
The Dirichlet-multinomial pair is computationally convenient, but does not adequately control model complexity.

The **Sparse Additive GEnerative model (SAGE):**
- gracefully handles extraneous parameters,
- adaptively controls sparsity without a regularization constant,
- facilitates inference in multifaceted models.
Generative models provide powerful tools for understanding natural language data.

Capabilities include prediction, clustering, and discovering latent topics, as well as more exotic models that combine latent and observed aspects.

As always, controlling model complexity is critical.
  - SAGE improves on the Dirichlet-Multinomial pair by modeling sparse deviations in log-odds.
Generative models provide powerful tools for understanding natural language data.

Capabilities include prediction, clustering, and discovering latent topics, as well as more exotic models that combine latent and observed aspects.

As always, controlling model complexity is critical.

- SAGE improves on the Dirichlet-Multinomial pair by modeling sparse deviations in log-odds.

Thanks!
Example Topics

20 Newsgroups, Vocab=20000, K=25

LDA (perplexity = 1131)

- health insurance smokeless tobacco smoked infections care meat
- wolverine punisher hulk mutants spiderman dy timucin bagged marvel
- gaza gazans glocks glock israeli revolver safeties kratz israel
- homosexuality gay homosexual homosexuals promiscuous optilink male
- god turkish armenian armenians gun atheists armenia genocide firearms
Example Topics

20 Newsgroups, Vocab=20000, K=25

**LDA (perplexity = 1131)**

- health insurance smokeless tobacco smoked infections care meat
- wolverine punisher hulk mutants spiderman dy timucin bagged marvel
- gaza gazans glocks glock israeli revolver safeties kratz israel
- homosexuality gay homosexual homosexuals promiscuous optilink male
- god turkish armenian armenians gun atheists armenia genocide firearms

**SAGE (Perplexity = 1090)**

- ftp pub anonymous faq directory uk cypherpunks dcr loren
- disease msg patients candida dyer yeast vitamin infection syndrome
- car cars bike bikes miles tires odometer mavenry altcit
- jews israeli arab arab israel objective morality baerga amehdi hossien
- god jesus christians bible faith atheism christ atheists christianity