Latent Dirichlet Allocation

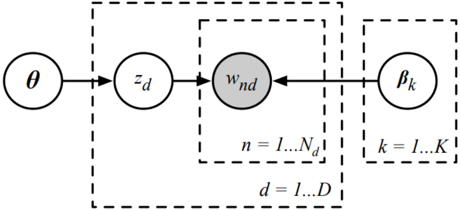
David Mareček

November 04, 2021

EUROPEAN UNION
European Structural and Investment Fund
Operational Programme Research, Development and Education

Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics

Mixture of Categoricals Model

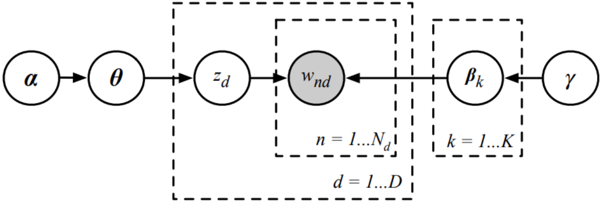


$$z_d \sim Cat(\theta)$$

$$w_{nd} | z_d \sim Cat(\beta_{z_d})$$

With the Expectation-Maximization algorithm we have essentially estimated θ and β by maximum likelihood.

Bayesian Mixture of Categoricals Model



$$z_d \sim Cat(\theta)$$
$$\theta \sim Dir(\alpha)$$
$$w_{nd} | z_d, \beta \sim Cat(\beta_{z_d})$$
$$\beta_k \sim Dir(\gamma)$$

An alternative, Bayesian treatment infers these parameters starting from priors, e.g.:

- $\theta \sim Dir(\alpha)$ is a symmetric Dirichlet over category probabilities,
- $\beta_k \sim Dir(\gamma)$ are symmetric Dirichlets over vocabulary probabilities.

What is different?

- We no longer want to compute a point estimate of θ and β.
- We are now interested in computing posterior distributions.

Collapsed sampling for Bayessian Mixture of Categoricals

We want to employ Gibbs Sampling to sample the model variables z_d, β, and θ.

Collapsed Gibbs Sampler: We will sample only the latent variables z_d. The model parameters β and θ are marginalized (integrated out).
In each step, we sample one latent variable z_d conditioned by all the other latent variables z_{-d}, all the documents w, and our hyperparameters γ and α.

$$p(z_d = k | \{w\}, \{z_{-d}\}, \gamma, \alpha)$$

We rewrite it using Bayes theorem.

$$= \frac{p(z_d = k | \{z_{-d}\}, \gamma, \alpha) \; p(\{w\} | z_d = k, \{z_{-d}\}, \gamma, \alpha)}{p(\{w\} | \{z_{-d}\}, \gamma, \alpha)}$$

The denominator is constant (does not depend on category k), the parts in the nominator also do not depend on both the hyperparameters.

$$\propto p(z_d = k | \{z_{-d}\}, \alpha) \; p(\{w\} | z_d = k, \{z_{-d}\}, \gamma)$$

Collapsed sampling for Bayessian Mixture of Categoricals [2]

We have:

$$p(z_d = k|\{w\}, \{z_{-d}\}, \gamma, \alpha) \;\propto\; p(z_d = k|\{z_{-d}\}, \alpha)\; p(\{w\}|z_d = k, \{z_{-d}\}, \gamma)$$

Probability of the document collection $p(\{w\})$ may be rewritten as $p(w_d|w_{-d})p(w_{-d})$. However $p(w_{-d})$ does not depend on z_d, so:

$$\propto\; p(z_d = k|\{z_{-d}\}, \alpha)\; p(\{w_d\}|w_{-d}, z_d = k, \{z_{-d}\}, \gamma)$$

$$\propto p(z_d = k|\{z_{-d}\}, \alpha)\; \prod_{n=1}^{N_d} p(w_{nd}|\{w_{-d}\}, z_d = k, \{z_{-d}\}, \gamma)$$

For computing $p(z_d|z_{-d})$ and $p(w_d|w_{-d})$, we integrate over all possible parameters θ and γ respectively.

$$\propto \int p(z_d = k|\theta)p(\theta|z_{-d}, \alpha)d\theta \prod_{n=1}^{N_d} \int p(w_{nd}|\beta_k)p(\beta_k|\{w_{-d}\}, \{z_{-d}\}, \gamma)d\beta_k$$

Collapsed sampling for Bayessian Mixture of Categoricals [3]

We have:

$$\propto \int p(z_d = k|\theta)p(\theta|z_{-d}, \alpha)d\theta \prod_{n=1}^{N_d} \int p(w_{nd}|\beta_k)p(\beta_k|\{w_{-d}\}, \{z_{-d}\}, \gamma)d\beta_k$$

Both the integrals are expected values of Dirichlet distributions, therefore:

$$p(z_d = k|\{w\}, \{z_{-d}\}, \gamma, \alpha) \propto \frac{\alpha + c_d[k] - 1}{K\alpha + D - 1} \prod_{n=1}^{N_d} \frac{\gamma + c_w[w_{nd}][k]}{M\gamma + \sum_{m=1}^{M} c_w[m][k]}$$

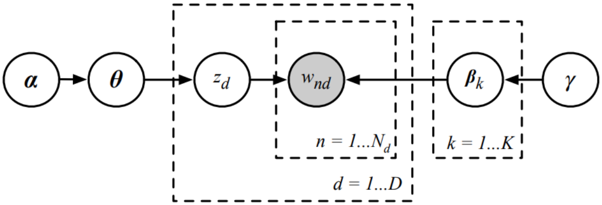
- $c_d[k]$ ... How many documents are assigned to topic k.
- $c_w[m][k]$ ... How many times the word m is in a document assigned to topic k.

Algorithm for Bayessian Mixture of Categoricals

```
initialize z_d randomly ∀d ∈ 1..D;
compute initial counts c_d[k], c_w[m][k], c[k]  ∀k ∈ 1..K, ∀m ∈ 1..M;
for i ← 1 to I do
    for d ← 1 to D do
        c_d[z_d]−−;
        for n ← 1 to N_d do
            c_w[w_nd][z_d]−−; c[z_d]−−;
        end
        for k ← 1 to K do
            p[k] = (α+c_d[k])/(Kα+D−1) ∏_{n=1}^{N_d} (γ+c_w[w_nd][k])/(Mγ+c[k]);
        end
        sample k from probability distribution p[k];
        z_d ← k; c_d[k]++;
        for n ← 1 to N_d do
            c_w[w_nd][z_d]++; c[z_d]++;
        end
    end
```

$$p[k] = \frac{\alpha+c_d[k]}{K\alpha+D-1} \prod_{n=1}^{N_d} \frac{\gamma+c_w[w_{nd}][k]}{M\gamma+c[k]}$$

Limitations of the mixture of categoricals model



$$z_d \sim Cat(\theta)$$
$$\theta \sim Dir(\alpha)$$
$$w_{nd}|z_d, \beta \sim Cat(\beta_{z_d})$$
$$\beta_k \sim Dir(\gamma)$$

A generative view of the mixture of categoricals model:

1. Draw a distribution θ over K topics from a $Dirichlet(\alpha)$.
2. For each topic k, draw a distribution β_k over words from a $Dirichlet(\gamma)$.
3. For each document d, draw a topic z_d from a $Categorical(\theta)$
4. For each document d, draw N_d words w_{nd} from a $Categorical(\beta_{zd})$

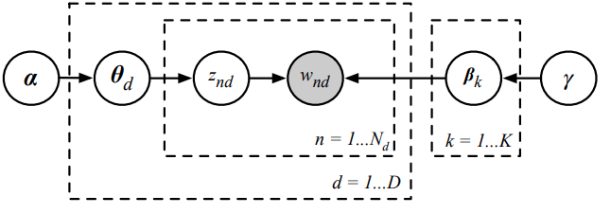
Limitations:

- All words in each document are drawn from one specific topic distribution.
- This works if each document is exclusively about one topic, but if some documents span more than one topic, then "blurred" topics must be learnt.

Jump...

Jump to http://mlg.eng.cam.ac.uk/teaching/4f13/1617/lda.pdf

Bayesian Latent Dirichlet Allocation



$$z_{nd} \sim Cat(\theta_d)$$

$$\theta_d \sim Dir(\alpha)$$

$$w_{nd} | z_{nd}, \beta \sim Cat(\beta_{z_{nd}})$$

$$\beta_k \sim Dir(\gamma)$$

An alternative, Bayesian treatment infers these parameters starting from priors, e.g.:

- $\theta \sim Dir(\alpha)$ is a symmetric Dirichlet over category probabilities,
- $\beta_k \sim Dir(\gamma)$ are symmetric Dirichlets over vocabulary probabilities.

What is different?

- We no longer want to compute a point estimate of θ and β.
- We are now interested in computing posterior distributions.

Collapsed sampling for Latent Dirichlet Allocation

$$p(z_{nd} = k | \{w\}, \{z_{-nd}\}, \gamma, \alpha) =$$

(rewrite using Bayes theorem)

$$= \frac{p(z_{nd} = k | \{z_{-nd}\}, \gamma, \alpha) \; p(\{w\} | z_{nd} = k, \{z_{-nd}\}, \gamma, \alpha)}{p(\{w\} | \{z_{-nd}\}, \gamma, \alpha)}$$

(the denominator is constant with respect to z_{nd}; generation of topics does not depend on γ; generation of words for given topic does not depend on γ)

$$\propto p(z_{nd} = k | \{z_{-nd}\}, \alpha) \; p(\{w\} | z_{nd} = k, \{z_{-nd}\}, \gamma)$$

(probability of data $p(w)$ can be also rewritten as $p(w_{nd}|w_{-nd})p(w_{-nd})$ and $p(w_{-nd})$ is constant with respect to z_{nd})

$$\propto p(z_{nd} = k | \{z_{-nd}\}, \alpha) \; p(w_{nd} | \{w_{-nd}\}, z_{nd} = k, \{z_{-nd}\}, \gamma)$$

Collapsed sampling for Latent Dirichlet Allocation [2]

$$p(z_{nd} = k | \{w\}, \{z_{-nd}\}, \gamma, \alpha) \propto$$
$$\propto p(z_{nd} = k | \{z_{-nd}\}, \alpha)\ p(w_{nd} | \{w_{-nd}\}, z_{nd} = k, \{z_{-nd}\}, \gamma)$$

(for each predictive distribution, we integrate over all possible parameters β_k and θ_d)

$$\propto \int p(z_{nd} = k | \theta_d) p(\theta_d | z_{-nd}, \alpha) d\theta_d \int p(w_{nd} | \beta_k) p(\beta_k | \{w_{-nd}\}, \{z_{-nd}\}, \gamma) d\beta_k$$

(these integrals can be easily computed; see predictive distribution for Dirichlet posteriors)

$$= \frac{\alpha + c_d[d][k]}{K\alpha + N_d - 1} \frac{\gamma + c_w[w_{nd}][k]}{M\gamma + \sum_{m=1}^{M} c_w[m][k]}$$

Where:

- $c_d[d][k]$ = how many words in document d are assigned to topic k.
- $c_w[m][k]$ = how many times the word m is assigned to topic k (across all documents).

The current position z_{nd} is always excluded from the counts.

LDA Algorithm

```
initialize z_nd randomly ∀d ∈ 1..D, ∀n ∈ 1..N_d;
compute initial counts c_d[d][k], c_w[m][k], c[k]  ∀d ∈ 1..D, ∀k ∈ 1..K, ∀m ∈ 1..M;
for i ← 1 to I do
    for d ← 1 to D do
        for n ← 1 to N_d do
            c_d[d][z_nd]--; c_w[w_nd][z_nd]--; c[z_nd]--;
            for k ← 1 to K do
                p[k] = (α+c_d[d][k])/(Kα+N_d−1) · (γ+c_w[w_nd][k])/(Mγ+c[k]);
            end
            sample k from probability distribution p[k];
            z_nd ← k;
            c_d[d][k]++; c_w[w_nd][k]++; c[k]++;
        end
    end
end
```

where

$$p[k] = \frac{\alpha+c_d[d][k]}{K\alpha+N_d-1}\ \frac{\gamma+c_w[w_{nd}][k]}{M\gamma+c[k]}$$

LDA Algorithm - topics assignment on a new data

initialize z_{nd} *randomly* $\forall d \in 1..D,\ \forall n \in 1..N_d$;
fix the counts $c_w[m][k]$ *and* $c[k]$ *obtained during training*;
compute initial counts $c_d[d][k]$ $\forall d \in 1..D,\ \forall k \in 1..K$;
for $i \leftarrow 1$ **to** I **do**
    for $d \leftarrow 1$ **to** D **do**
        for $n \leftarrow 1$ **to** N_d **do**
            $c_d[d][z_{nd}]$−−;
            for $k \leftarrow 1$ **to** K **do**
                $p[k] = \frac{\alpha + c_d[d][k]}{K\alpha + N_d - 1} \frac{\gamma + c_w[w_{nd}][k]}{M\gamma + c[k]}$;
            end
            sample k *from probability distribution* $p[k]$;
            $z_{nd} \leftarrow k$;
            $c_d[d][k]$++;
        end
    end
end

Entropy of text

- joint probability $p(T) = \prod_{i=1}^{N} p(w_i) = \prod_{m=1}^{M} p(m)^{c_m}$
- log probability $\log p(T) = \sum_{i=1}^{N} \log p(w_i) = \sum_{m=1}^{M} c_m \log p(m)$
- entropy $H(T) = -\frac{1}{N} \sum_{i=1}^{N} \log p(w_i) = -\sum_{m=1}^{M} \frac{c_m}{N} \log p(m) = \frac{-\log p(T)}{N}$
- perplexity $PP(T) = 2^{H(T)}$

A perplexity of g corresponds to the uncertainity associated with a die with g sides, which generates each new word.

All the logarithms used here are binary (with base 2)

Entropy of the text for a topic in LDA

Probability of word w given a topic k is

$$p(w|k) = \frac{\gamma + c_w[w][k]}{M\gamma + \sum_{m=1}^{M} c_w[m][k]},$$

where the counts c_w are taken from the training data, M is the size of the vocabulary. The entropy of a topic is computed as follows:

$$H(k) = -\sum_{w=1}^{M} p(w|k) \log_2 p(w|k)$$

Perplexity is $PP(k) = 2^{H(k)}$.

Perplexity of the LDA model on test data

Probability of word w in document d is

$$p(w|d) = \sum_{k=1}^{K} p(w|k)p(k|d) = \sum_{k=1}^{K} \frac{\gamma + c_w[w][k]}{M\gamma + \sum c_w[m][k]} \frac{\alpha + c_d[d][k]}{K\alpha + N_d},$$

where the counts c_w are taken from the training data, and counts c_d and N_d are taken from the test data.
The entropy is computed as the average of the log probabilities over all words in the test data.

$$H = -\frac{1}{N_{test}} \sum_{d=1}^{D_{test}} \sum_{n=1}^{N_d} \log_2 p(w_{nd}),$$

where N_{test} is the total number of words in the test data. Perplexity is $PP = 2^H$.

Perplexity of a simple model without topics

Probability of word w in the test data given the training data is

$$p(w) = \frac{\gamma + c_w[w]}{M\gamma + \sum c_w[m]}$$

where the counts c_w are taken from the training data.
The entropy is computed as the average of the log probabilities over all words in the test data.

$$H = -\frac{1}{N_{test}} \sum_{d=1}^{D_{test}} \sum_{n=1}^{N_d} \log_2 p(w_{nd}),$$

where N_{test} is the total number of words in the test data. Perplexity is $PP = 2^H$.