Gibbs Sampling for LDA, Applications

Pawan Goyal

CSE, IIT Kharagpur

Week 9, Lecture 3

Approximating the posterior

Algorithms to approximate it fall in two categories:

Sampling-based Algorithms

Collect samples from the posterior to approximate it with an empirical distribution

Approximating the posterior

Algorithms to approximate it fall in two categories:

Sampling-based Algorithms

Collect samples from the posterior to approximate it with an empirical distribution

Variational Methods

- Deterministic alternative to sampling-based algorithms
- The inference problem is transformed to an optimization problem

Gibbs Sampling

- A form of Markov chain Monte Carlo (MCMC), which simulates a high-dimensional distribution by sampling on lower-dimensional subset of variables where each subset is conditioned on the value of all others
- Sampling is done sequentially and proceeds until the sampled values approximate the target distribution
- It directly estimates the posterior distribution over z, and uses this to provide estimates for β and θ

Gibbs Sampling

- Suppose we have a word token i for which we want to find the topic assignment probability : $p(z_i = j)$
- Represent the collection of documents by a set of word indices w_i and document indices d_i for this token i
- Gibbs sampling considers each word token in turn and estimates the probability of assigning the current word token to each topic, conditioned on the topic assignment to all other word tokens
- From this conditional distribution, a topic is sampled and stored as the new topic assignment for this word token
- This conditional is written as $P(z_i = j|z_{-i}, w_i, d_i, .)$

Gibbs Sampling

- Let us define two matrices C^{WT} and C^{DT} of dimensions $W \times T$ and $D \times T$ respectively.
- ${C_{wj}}^{WT}$ contains the number of times word w is assigned to topic j, not including the current instance
- ${C_{dj}}^{WT}$ contains the number of times topic j is assigned to some word token in document d, not including the current instance

Gibbs Sampling

- Let us define two matrices C^{WT} and C^{DT} of dimensions $W \times T$ and $D \times T$ respectively.
- ${C_{wj}}^{WT}$ contains the number of times word w is assigned to topic j, not including the current instance
- ${C_{dj}}^{WT}$ contains the number of times topic j is assigned to some word token in document d, not including the current instance

$$P(z_i = j | z_{-i}, w_i, d_i, .) \propto \frac{{C_{w_ij}}^{WT} + \eta}{\sum_{w=1}^{W} {C_{wj}}^{WT} + W\eta} \frac{{C_{d_ij}}^{DT} + \alpha}{\sum_{t=1}^{T} {C_{d_it}}^{DT} + T\alpha}$$

- The left part is the probability of word w under topic j (How likely a word is for a topic) whereas
- The right part is the probability of topic j under the current topic distribution for document d (How dominant a topic is in a document)

Algorithm

- Start: Each word token is assigned to a random topic in $[1\ldots T]$
- For each word token, a new topic is sampled as per $P(z_i = j|z_{-i}, w_i, d_i, .)$, adjusting the matrices C^{WT} and C^{DT}
- A single pass through all word tokens in the document is one *Gibbs sample*
- After the burnin period, these samples are saved at regularly spaced intervals, to prevent correlations between samples

Estimating θ and β

$$\beta_i^{(j)} = \frac{C_{ij}^{WT} + \eta}{\sum_{k=1}^{W} C_{kj}^{WT} + W\eta}$$

$$\theta_j^{(d)} = \frac{C_{dj}^{DT} + \alpha}{\sum_{k=1}^{T} C_{dk}^{DT} + T\alpha}$$

These values correspond to predictive distributions of

- sampling a new token of word i from topic j, and
- sampling a new token in document d from topic j

An Example

The algorithm can be illustrated by generating artificial data from a known topic model and applying the algorithm to check whether it is able to infer the original generative structure.

Example

- Let topic 1 give equal probability to MONEY, LOAN, BANK and topic 2 give equal probability to words RIVER, STREAM, and BANK

$$\beta_{MONEY}{}^{(1)} = \beta_{LOAN}{}^{(1)} = \beta_{BANK}{}^{(1)} = 1/3$$

$$\beta_{RIVER}{}^{(2)} = \beta_{STREAM}{}^{(2)} = \beta_{BANK}{}^{(2)} = 1/3$$

- We generate 16 documents by arbitrarily mixing two topics.

Initial Structure

	River	Stream	Bank	Money	Loan
1			○○○○	●○○○●○	●●○●○○
2			○○●○○	●●●●●●○	●○○●
3			○○○●○○○	○●○●○	●○○○
4			●●●○●○○	○●●○○○	○○○
5			●●○●○●○	●○	○●○○○○○
6			○●●○●●●●●	○●○	○○●●
7	○		○●●●	●●○○●○	○●●●○
8	●	○●	○○●●●●	○●●○	●●○
9	●	○○●	○○○○○●	●○●●	○●
10	●○	●●○	●○○○○○	●	●○○●
11	○●	○●●	○○○●●○○	●●●	●
12	○○○	○○○○●○	●○●●○●	○	
13	○○○●●●	○●○	●○○○●●		○
14	○○	●●○○○●●●	●●○●○○		
15	○●●●	●●●○○●○	●○●○●		
16	●○●●○	●●○○○○●	●●●○		

Colors reflect initial random assignment, black = topic 1, while = topic 2

After 64 iterations of Gibbs Sampling

	River	Stream	Bank	Money	Loan
1			●●●●	●●●●●●	●●●●●●
2			●●●○●	●●●●●●●	●●●●
3			●●●●●●●	●●●●●	●●●●
4			●●●●●●●	●●●●●●	●●●
5			●●●○●●●	●●	●●●●●●●
6			●●●●●●●●●	●●●	●●●●
7	○		●●●○	●●●●●●	●●●●●
8	○	○○	●○○○●○	●●●●	●●●
9	○	○○○	●○●●●●	●●●●	●●
10	○○	○○○	○●●●●●	●	●●●●
11	○○	○○○	○○○●○○○	●●●	●
12	○○○	○○○○○○	○●○●●●	●	
13	○○○○○○	○○○	●○○○○○		●
14	○○	○○○○○○○○	○○○○○○		
15	○○○○	○○○○○○○	○○○○○		
16	○○○○○	○○○○○○○	○○○○		

$$\beta_{MONEY}{}^{(1)} = 0.32, \beta_{LOAN}{}^{(1)} = 0.29, \beta_{BANK}{}^{(1)} = 0.39$$

$$\beta_{RIVER}{}^{(2)} = 0.25, \beta_{STREAM}{}^{(2)} = 0.4, \beta_{BANK}{}^{(2)} = 0.35$$

Computing Similarities

Document Similarity

Similarity between documents d_1 and d_2 can be measured by the similarity between their topic distributions $\theta^{(d_1)}$ and $\theta^{(d_2)}$

KL divergence : $D(p,q) = \sum_{j=1}^{T} p_j log_2 \frac{p_j}{q_j}$

Symmetrized KL divergence: $\frac{1}{2}[D(p,q)+D(q,p)]$ seems to work well

Similarity with respect to query q

Maximize the conditional probability of query given the document:

$$p(q|d_i) = \prod_{w_k \in q} p(w_k|d_i)$$

$$= \prod_{w_k \in q} \sum_{j=1}^{T} P(w_k|z=j)P(z=j|d_i)$$

Computing Similarities

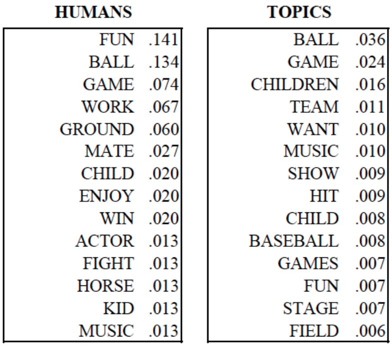
Similarity between two words

Having observed a single word in a new context, what are the other words that might appear in the same context, based on the topic interpretation for the observed word?

$$p(w_2|w_1) = \sum_{j=1}^{T} p(w_2|z=j)p(z=j|w_i)$$

Example

Observed and predicted responses for the word ‘PLAY’



HUMANS	
FUN	.141
BALL	.134
GAME	.074
WORK	.067
GROUND	.060
MATE	.027
CHILD	.020
ENJOY	.020
WIN	.020
ACTOR	.013
FIGHT	.013
HORSE	.013
KID	.013
MUSIC	.013

TOPICS	
BALL	.036
GAME	.024
CHILDREN	.016
TEAM	.011
WANT	.010
MUSIC	.010
SHOW	.009
HIT	.009
CHILD	.008
BASEBALL	.008
GAMES	.007
FUN	.007
STAGE	.007
FIELD	.006