

National Language Processing
Prof. Pawan Goyal
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture – 43
Gibbs Sampling for LDA, Application

Hello everyone. Welcome to the third lecture of this week. So, we were talking about topic models and we have discussed; what is the formulation of latent Dirichlet allocation; and we say that what are the different parameters that we need to learn. We need to learn mainly three parameters that is what are my topic distributions given a topic what is the probability of each word, what are my per document topic proportion, so that is my θ s and then per document per word topic assignment that is by z . So, this I have to compute the posterior distribution of all this parameter given my observation, observation is all my corpus all the documents that I am seeing.

So, in this lecture, we will discuss one interesting method for doing that that is Gibbs sampling. And then in the end, we will also talk about some simple applications that once we had learnt topic models over a corpus what are the some of the simple things that you can do.

(Refer Slide Time: 01:21)

Approximating the posterior

Algorithms to approximate it fall in two categories:

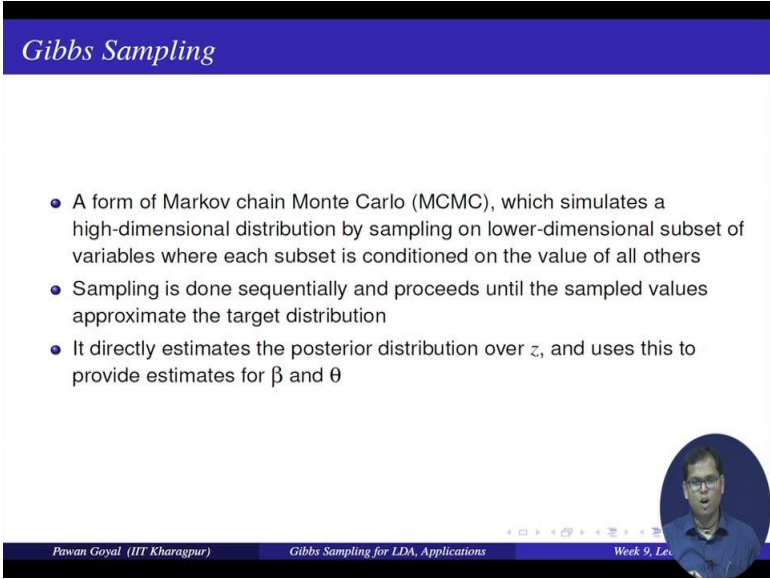
- Sampling-based Algorithms**
Collect samples from the posterior to approximate it with an empirical distribution
- Variational Methods**
 - Deterministic alternative to sampling-based algorithms
 - The inference problem is transformed to an optimization problem

Pawan Goyal (IIT Kharagpur) Gibbs Sampling for LDA, Applications Week 9, Lecture 3 2 / 13

So, for approximating the posterior probabilities of all these parameters, the algorithm generally fall in two categories. So, one are sampling based algorithms so that is from the

posterior you try to collect the samples of the distributions, and then approximate it with the empirical distribution and that is what we will focus on in Gibbs sampling. And there are also variational methods, so where we convert the inference problem to some sort of optimization problem and try to learn the parameters that we optimize that. So, both the methods are very much used in the literature. So, we will see Gibbs sampling that is an easier to understand method that I will say.

(Refer Slide Time: 02:30)



Gibbs Sampling

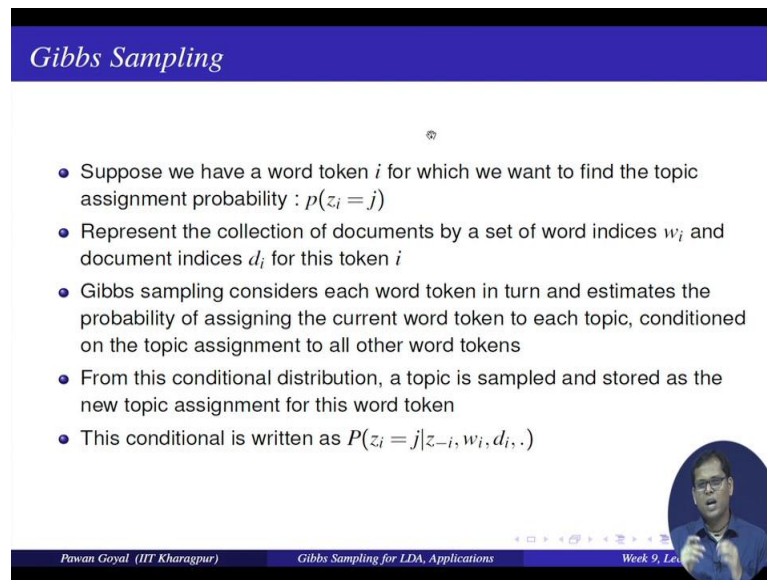
- A form of Markov chain Monte Carlo (MCMC), which simulates a high-dimensional distribution by sampling on lower-dimensional subset of variables where each subset is conditioned on the value of all others
- Sampling is done sequentially and proceeds until the sampled values approximate the target distribution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

Pawan Goyal (IIT Kharagpur) Gibbs Sampling for LDA, Applications Week 9, Lec 1

So, Gibbs sampling are some sort of Markov chain Monte Carlo methods. So, what is the idea? So having a high dimensional distribution; think about all the possible values that that your parameter can take all your betas, theta, z , they can take so huge number of values. So, idea here is you sample on lower dimensional subset of variables and each subset is conditioned on whatever as it known.

So, you do not computing the joint probability of everything you are assuming sum to be known and computing probability for others and that you keep on doing in terms for all the all the variables. So, sampling is done sequentially and proceeds until the sampled value is approximate my target distribution. And it directly estimates the posterior distribution over z , and uses this to provide estimates for beta and theta. So, we will find out the distribution over z , and then use that to compute my theta and beta and we will see how do we do that.

(Refer Slide Time: 03:12)



Gibbs Sampling

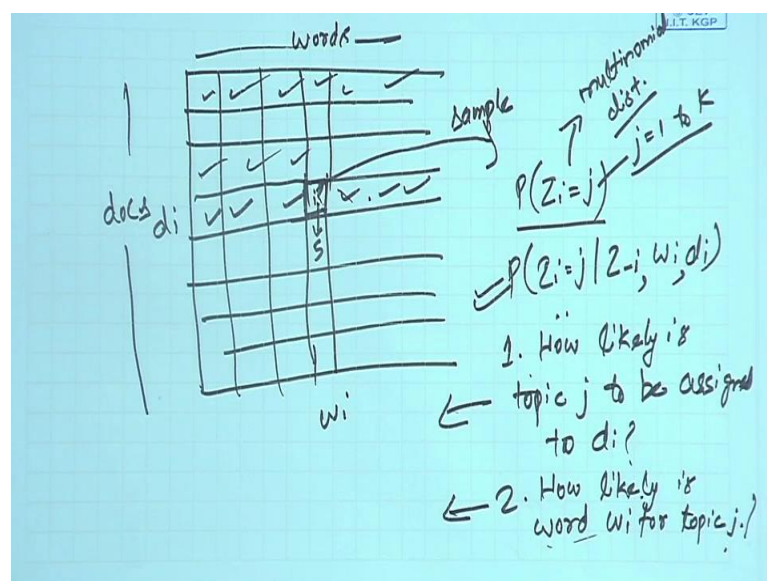
- Suppose we have a word token i for which we want to find the topic assignment probability : $p(z_i = j)$
- Represent the collection of documents by a set of word indices w_i and document indices d_i for this token i
- Gibbs sampling considers each word token in turn and estimates the probability of 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 $P(z_i = j | z_{-i}, w_i, d_i, .)$

Pawan Goyal (IIT Kharagpur) Gibbs Sampling for LDA, Applications Week 9, Lecture 1

So, what is the idea? So, assume that you are having a corpus; and in the corpus you have some documents and words. Now take so you will assume that in a in one of the iteration you are at a particular word that is the word token i . And what you want to find out what is the topic assignment probability, what is the probability that is i th token will be assigned to topic j probability $z_i = j$ that is what you want to find out. So, now, how do you do that? Now, you represent the collection of documents by a set of word indices w_i and document indices d_i for this token i . So, it simply says that you have a lot of words find out what is the corresponding word token for this i that will be called w_i w_i will give you the particular index and similarly d_i , d_i will be the particular document where you are having this token.

So, what Gibbs sampling does it consider each word token in turn and estimates the probability of assigning the current word token to each topic conditioned on the topic assignment of all other word tokens. So, what it is saying? That in the particular iteration assumes that you the topic assignment for all the other words, except this word. Now, based on this you try assign a topic to this word.

(Refer Slide Time: 04:38)



So, what will happen, you are having a set of documents. So, these are your documents and suppose you know your unique words these are your words. And you are at a i th token. So, the corresponding document will be called d_i , this is i token and the corresponding word will be called w_i . So, now what you have to find out probability that the topic assignment for this word will be z that is what we have to find out. And what is assumed is that in this iteration you know the topic assigned for everything else.

So, you know what are the topics that assigned for all different words you know all that, but you want to estimate to find out the topic for this particular i th word. So, how do you do that, you first estimate this probability and this you do condition on everything else. So, we write it like this probability z_i at j given the topic assignment for all the words other words minus sign is everything other than this i and what is the word index and what is the document index, this is what we want to find out.

Now, intuitively what should it depend on? So, what is the probability that this i th token will be given assigned the topic j . What should it depend on? So, we think in terms of topic models, this probability that the i th word should be assigned the topic j should depend on two things one is so we say that document consist of certain topics. So, if this topic is prevalent in the document, there might be a high chance that this word is assigned this topic j , so that is how likely is topic j to be assigned to d_i . What is the next thing, how likely is that this word occurs in topic j is word w_i for topic j . So, we have

two things I need to know. Now, how do I know this one, how likely is topic j to be assigned to document d_i . You see I am given the topic assignment for all other words in this document. So, I will find out what sections of words in this document are assigned this topic j divide by the number of words. So, this I can computationally by this topic assigned.

Now, how do you compute this how likely each word w_i for topic j , I will again see in topic j what are the different words that come and how many times word w_i comes in this topic j . And this I can use to compute this probability. And then I can multiply these two to find out what is the probability of a topic j been assigned to this word token i and this I will do for all topics j is equal to 1 to k . So, this will give me a multinomial distribution. Then to give a assignment to this word I will sample from this multinomial distribution.

And then from sampling, I will assign one topic and I will assign the topic here. So, suppose the topic is 5, I will put this is 5. And then I will move to the next word token, and then I will assume that this is not given, and I will take everything as given do that find the topic assignment and that is what I keep on doing. So, now once the intuition is clear, let us go back to the formulation. So, yes from this conditional distribution, you are sampling a topic and we are storing it as the new topic assignment for the word and this is written as this assignment probability that i th word has the has a topic j given everything else.

(Refer Slide Time: 09:08)

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}^{DT} contains the number of times topic j is assigned to some word token in document d , not including the current instance

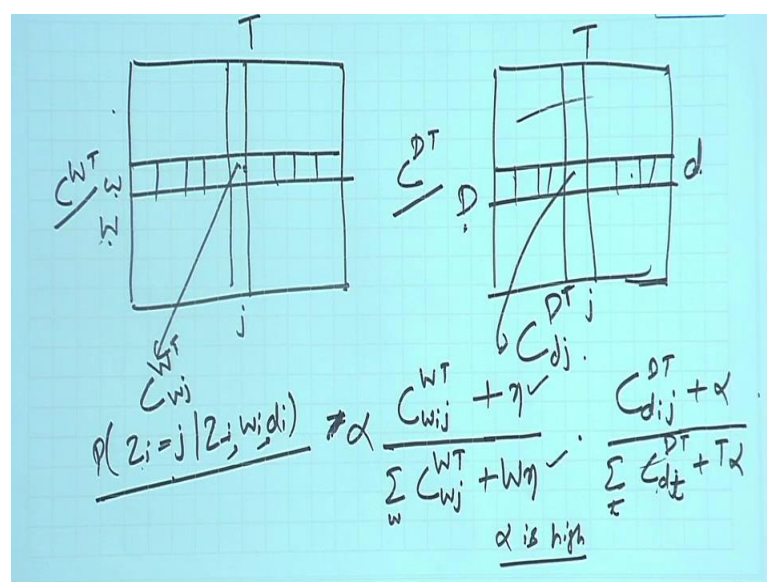
$$P(z_i = j | z_{-i}, w_i, d_i, \cdot) \propto \frac{C_{wj}^{WT} + \eta}{\sum_{w=1}^W C_{wj}^{WT} + W\eta} \frac{C_{dj}^{DT} + \alpha}{\sum_{t=1}^T C_{dt}^{DT} + T\alpha}$$

- 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)

Pawan Goyal (IIT Kharagpur) Gibbs Sampling for LDA, Applications Week 9, Lecture 3 5 / 13

So, how do we achieve this, how do we compute all these values. For that I need to keep two different matrices. So, one is C^{WT} of dimensions W times T another is C^{DT} of dimensions D times T .

(Refer Slide Time: 09:29)



So, what are these matrices? So, one is C^{WT} W times T , and next is C^{DT} D times T . T is the number of topics, W is the number of words and D is the number of documents. So, what do they store? So, C_{wj} , WT that is I take the word w and topic j , so wj WT . So, this element is called C_{wj}^{WT} this contains the number of times word w is

assigned to topic j not including the current instance. So, we are taking at the current instance except that how many times this word is assigned this topic that you can find out from the your whole data, you will actually keep this stable. So, we will just update the values.

So, at any point, you know how many times this word is assigned topic j . So, we will do that for all the values in this matrix. How many times this word is assigned this topic? Yes, similarly an element $C_{D T}$ small $d j$, this will contain for the document d topic j number of times topic j is assigned to some word token in document d that is how many words in the document d are assigned to topic j in that again you can have all the values. So, this will again be not included in the current instance, we will find out how many times any word in document d is assigned this topic and we will do it for all the documents. So, this you will have these two matrices $C_{W T}$, $C_{D T}$ and you are understand now what is element is.

Now once you have these two matrices how do I compute probability Z_i is equal to j given z minus i w_i , d_i and research may depend on two different things. So, let us see that. So, depends on two parts, one is what is the probability of word w under topic j . Remember we are talking about two parts. So, what will depend on? So, we are saying that it will depend on how likely is this word to come under topic j ; second is how likely is this topic to be assigned to document d - two things. How likely is this word to come on to topic j ? How likely is this topic to come under document d ?

Now, how do I write this probabilities in terms of this matrix elements how likely is word w to come under topic $d j$. So, that will be $C_{i w}$ word topic j $W T$ summation over all the words. So, this will give p probability for this word, summation over all words $C_{w j} W T$. Similarly, for how likely is topic Z to come on the document d this would be $C_{d i}$ for the current instance j $D T$ summation over, now for all the topic that are assigned in this document. So, this will be summation over all my topics $C_{d t}$, $C_{d t}$ capital $D T$. So, this will be the simple formulation.

So, there are certain priors that you take here. So, you will have some sort you can these are some sort of smoothing. So, here you have some smoothing parameter η , similarly it will be plus W times η , here α plus T times α , so smoothing parameters. And this will again to make it a probability; it is normalized, so instead of calling it equal to

you will say proportional to this. So, now, this gives you the formulation probability that the i th word token will be assigned topic j given all this.


So, now, you can say that what are these η and α . So, this is my Dirichlet parameters that we were seeing earlier. And you can also correlate this with their values. So, suppose your α is high, if your α is high, what would happen? This value will not matter and all the topics will be assigned roughly equal probability and that is what was happening. If you keep on increasing α you are going towards a distribution where all topics have some probabilities, but if your α is very, very small then what will matter is only this count and that is why you are going towards only a few topics. So, these are the intuitions.

Same thing you can do with η . So, if η is high, all the words will have equal probability of coming into topic if it is low then certain distributions will be preferred. So, now this is the formulation for probability that Z_i is equal to j . So, left part here is the probability of word W under the topic j , so that is how likely the word is for a topic; and the right part is the probability of topic j under the current topic distribution for the topic. And this is what we had seen also in the previous. So, when we are doing it on the paper in the previous slide. Now, this will give you the probability distribution over all the topics given this token. Now, what is the next thing, you have this multinomial distribution, you sample a topic from here.

(Refer Slide Time: 16:11)

Algorithm

- Start: Each word token is assigned to a random topic in $[1 \dots T]$
- For each word token, a new topic is sampled as per $P(z_i = j | z_{-i}, w_i, d_i, \cdot)$, 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



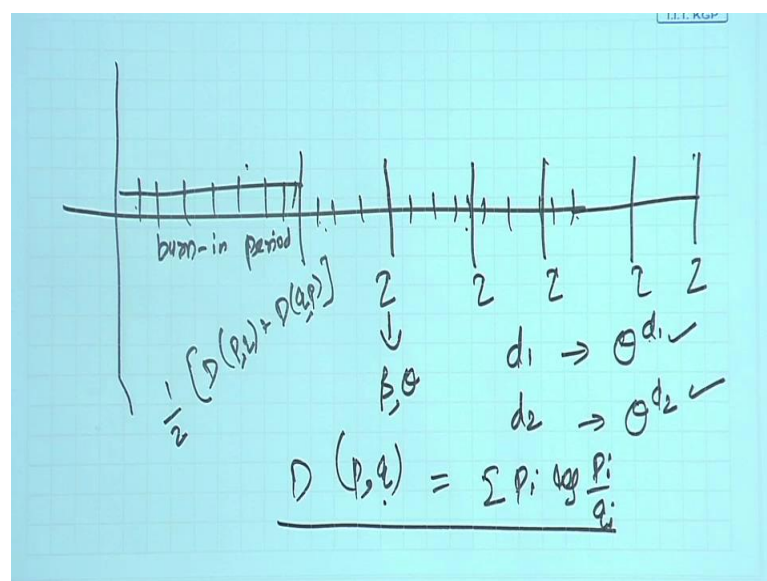
Pawan Goyal (IIT Kharagpur) Gibbs Sampling for LDA, Applications Week 9, Lecture 1

So, this is the whole algorithm in a nutshell. So, is how do you start. So, each word token is assigned to a predefined so random topic, so a random topic in 1 to T. So, you have this whole collection and you assign some random topic to each word. Now, you compute your two matrices C D T and C W T from that assignment. Now for each word token, in each iteration, what you will do for each word token you sample a new topic as per this distribution. And we have seen that once you have the matrices formulated you can find out this probability distribution and you can sample a topic by sampling from a multinomial distribution. And when you will sample a topic and put that topic in that, for that word accordingly adjust your two matrices.

So, now then you make complete single pass through all your words in your corpus that is called one Gibb sample. So, this is your one Gibb sample, and then you will do it again and again and again. So, what happens is that initially for certain iterations, you can call it as a burnin period, initially burnin period. So, where you will not store those samples, you will use them to update the values, but you will not store, but after some word burnin period you will start storing these values.

Now, you will not store every conjugative value. So, what might happen because you are just using a previous values to compute the next one, they may be very, very correlated. So, you will have some regularly spaced at some regularly expressed interval, you will store these samples.

(Refer Slide Time: 17:57)



So, something like; so, you are doing these iterations over the full corpus. So, there will be some initial burnin period. So, you are computing Gibb sample. After burnin in period you will have reliable Gibb sample, but what will happen those that are very, very close, they will be highly correlated. So, you will say I will store some regularly space intervals. So, say I will store it up every 100, after every 100 iteration, I will store this. So, we will have cube multiple Gibb samples now. So, each sample contains all your z, all your z, all your assignment is contains. And then and from here you can compute your beta and theta. And then finally, you can take an expectation over all these values. And this will give your one particular approximation of your parameters. If we take a expectation over various samples that you are getting from Gibb sample.

(Refer Slide Time: 19:09)


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



Pawan Goyal (IIT Kharagpur)
Gibbs Sampling for LDA, Applications
Week 9, Lec 9

So, once you have this Z, how do you compute your betas and theta? That is very easy; we were actually using that to compute Z. So, betas are the probability that a topic j is assigned to the ith word. So, this will be from the matrix C W T. So, I will take C i j W T plus eta divide over all words C k j W T plus W eta. Similarly, what is theta j d that is what is probability of topic j under document d; it will be C d j D T plus alpha divide by summation over all topics C d k D T plus T alpha. So, once we have the Z, then we can compute here beta and theta also.

So, these values will correspond to the distribution of sampling a new token of word i from topic j; and sampling a new token in document d from topic j.

(Refer Slide Time: 20:02)

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

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

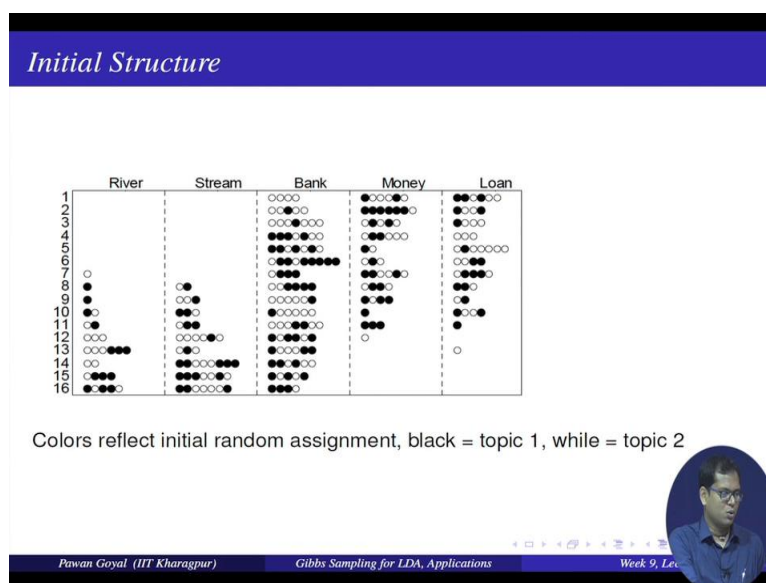
- We generate 16 documents by arbitrarily mixing two topics.

Pawan Goyal (IIT Kharagpur) Gibbs Sampling for LDA, Applications Week 9, Le...

Now, it is just an example to explain what it means to use Gibbs sampling. So, what is an example? So, this is like So, we are taking in a artificial data and for a known topic model and applying the algorithm, we will check if we can come back to the same topic distribution that we started with. So, what is done here, let us say we have two topics. So, we are doing a generation now, and then we will see whether Gibbs sampling can infer back the original topic distributions.

So, what is done in generative part let us say I have two topics - topic 1, topic 2. And simply we are saying topic one assigns equal probability to three words money, loan, and bank; and topic two assigns equal probability word river, steam and bank. So, all these three are assigned the probability of 1 by 3 each. So, these are topic distributions.

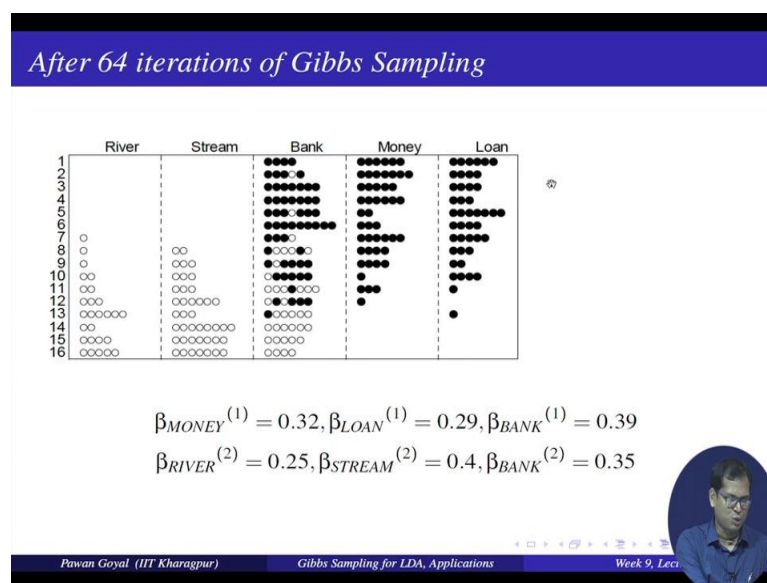
(Refer Slide Time: 21:10)



Now, by arbitrary mixture of these two topics we are generating 16 documents, so that is how these 16 documents look like. So, black means topic 1, and white means topic 2. So, there are two topics. So, each document has some number of words from different topics. So, I am sorry you should initial look at only the number of balls that is how many words are there in the document. Now, to apply Gibb sampling, so that is why you generated all these documents. So, this makes sense that you are generating documents that are having bank, money, loan lot of documents; then some document that contain only river, stream, bank. So, these are documents from only one topic these are document only from another topic, and there are some documents that are mix in these two topics, so that is how you are doing generating 16 documents.

Now, your task is can you use Gibbs sampling to find out what are the two topic distributions here. So, what is done for that, initialize all the words to some topic; so randomly that is why you are seeing the random assignments. So, black is topic 1 and white is topic 2 - some random assignment. From this random assignment, you can have the two matrices $C \times W \times T$, $C \times D \times T$. And then what you will do in each iteration, you will go to each word find out what is the probability that this word will be assigned to topics j sample from the distribution update your matrices and you get some Gibb samples.

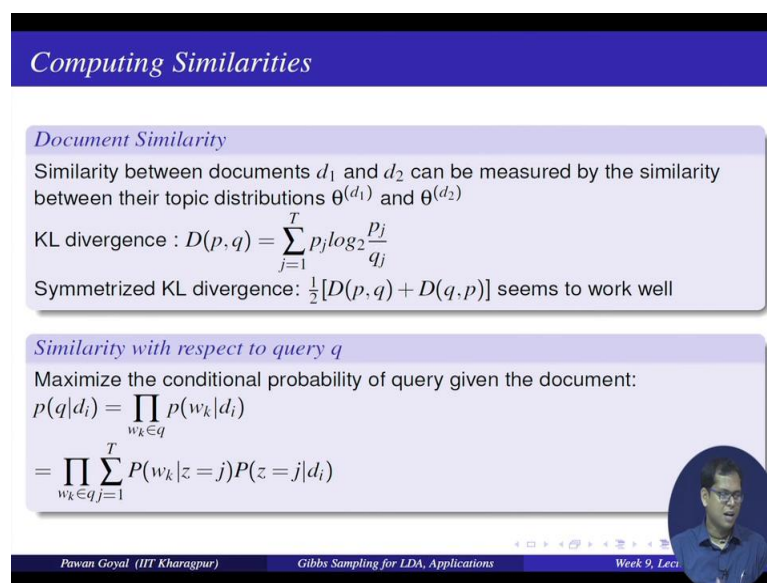
(Refer Slide Time: 22:36)



So, once you do that, so this is what you see after 64 iterations of Gibbs sampling. Can you see that? It actually looks very, very close to what we had initially. So, all these words are assigned to topic 1 and all these words are assigned to the next topic. And you can see that bank, money, loan are been assigned the same topic in a given document and that makes lot of sense. And from this particular sample, you can also compute your betas. And if you compute the betas, they come out to be very, very close to what you started with. So, started with each word having a probability of 1 by 3, and that is what is roughly what we obtain.

And this is just an explanation that how Gibbs sampling can help you to recover what is a original topic distribution this is from artificial data. But now you can do that for any real corpus, we have the real corpus and we want to find out what is the topic distributions apply Gibbs sampling and find out that. So, there are various tool kits available that I also disused in the last lecture. So, where you can give a corpus, you can define the number of topics and they can give you the all these values what are theta, what are betas, what are per topic per document per word topic distribution.

(Refer Slide Time: 24:00)



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:

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

Pawan Goyal (IIT Kharagpur) Gibbs Sampling for LDA, Applications Week 9, Lec. 1

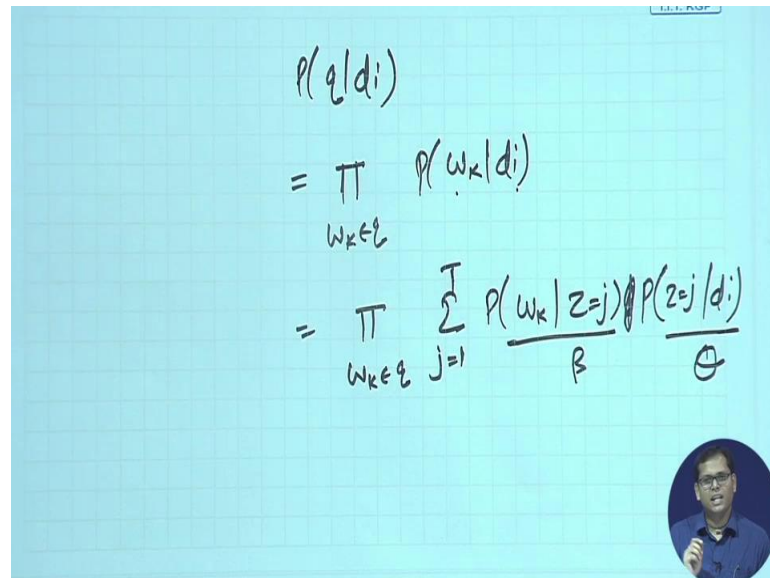
Now, once you have that what are sort of simple tasks that you can do with this. So, for example, one very important task is can you compute similarity between two documents and how will you do that. So, suppose I have two documents given in d_1 and d_2 . To compute the similarity between that I will see what is the topic distributions for d_1 and d_2 . So, if they are similar, I will say that the two documents are similar, but if their topic distributions are different, I will say they are different plus. So, I have two documents d_1 and d_2 , I find out what is θ for d_1 , what is θ for d_2 .

Now, I can say distance between two distribution p, q , I can use the KL divergence summation $p_i \log p_i$ by q_i . So, now, I compute the KL divergence between the two topic distributions for d_1 and d_2 and that will get me what is the distance between the two documents that is one every standard measure for finding out how similar two documents are. Because this is asymmetric, you can also do something like $\frac{1}{2} D(p, q) + D(q, p)$. So, we can also use that as some sort of distance symmetric, so that is you have now the corpus you find know the topic distributions. Now, you can use that to compare the similarity between any pair of documents, so that is one very, very important application of topic models.

Then you can find out what is similarity of the document with (Refer Time: 25:46) query, what is the probability that query is generated from a document that is what we study in information (Refer Time: 25:53). That you will have a lot of documents, you

want to find out given a query which document should be ranked higher this will be computed using what is the probability that the query is generated from this document. So, whichever documents give the highest probability of generating the query is given the highest score. Now, what is the formulation you want to find out probability of query given the document? The query is nothing but a set of words.

(Refer Slide Time: 26:22)



$$\begin{aligned}
 P(q|d_i) &= \prod_{w_k \in q} P(w_k|d_i) \\
 &= \prod_{w_k \in q} \sum_{j=1}^J \frac{P(w_k|z=j)}{\beta} \frac{P(z=j|d_i)}{\theta}
 \end{aligned}$$


So, if we take it simply easily multiplication over all the words in my query probability w_k given d_i . Now, how do I compute probability of word given d_i there I am using the LDA, the LDA model. So, I will say, so I will marginalize it over all the topics. So, this will be covered all words in query summation over all topics probability w_k given topic is j given document d this is nothing that comes from your beta directly that comes here from your theta directly. And use that to find out what is the probability of this query given this document. So, this is again a very interesting use.

(Refer Slide Time: 27:22)

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?

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


Pawan Goyal (IIT Kharagpur) Gibbs Sampling for LDA, Applications Week 9, Lec.


Then you can also use it to find out which two words are similar, what is the probability of word w_2 given w_1 that is nothing but again you marginalize over all the topics, and this you will compute from either your beta or by using Bayesian theorem. So, this will give me given a word w_1 what are some of the likely words in my vocabulary. And this you can find from computing words similarity and all that you have to using doing using distributional similarity and other stuff. So, that is why again a nice method for capturing semantics between words, documents, even sentences.

(Refer Slide Time: 28:02)

Example

Observed and predicted responses for the word 'PLAY'

HUMANS	TOPICS
FUN .141	BALL .036
BALL .134	GAME .024
GAME .074	CHILDREN .016
WORK .067	TEAM .011
GROUND .060	WANT .010
MATE .027	MUSIC .010
CHILD .020	SHOW .009
ENJOY .020	HIT .009
WIN .020	CHILD .008
ACTOR .013	BASEBALL .008
FIGHT .013	GAMES .007
HORSE .013	FUN .007
KID .013	STAGE .007
MUSIC .013	FIELD .006



Pawan Goyal (IIT Kharagpur) Gibbs Sampling for LDA, Applications Week 9, Lec.

So, how do you validate that this works, so this is very simple experiment. So, you have the word play. By using topic model, you find out which words have the highest probability given the word play. So, probability of x given play, find out some top words. Then you ask some humans to say that when you hear the word play what words come into your mind that is humans. So, words like fun, ball, game, work, ground, mate, child etcetera they come to their mind. Then you try to see whether these two lists are similar. And you find that many words are similar like ball, game they come on top even in topic model. So, this is the interest interesting way of evaluating whether a model is doing well for capturing words in that.

So, we discussed how do we use Gibbs sampling to estimate these parameters and some simple applications. Next, we will also talk about some non-parametric Bayesian model and what are the different applications they can be used for.

Thank you.