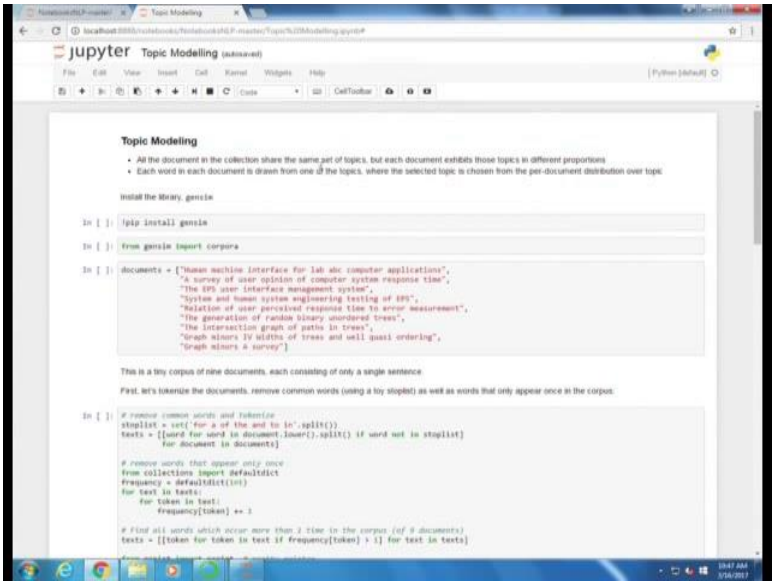
Natural Language Processing
Prof. Amrith Krishna
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 58
Tutorial

Hello everyone. So, welcome to today's tutorial session, today we will be discussing about topic modeling. So, for this tutorial, we assume that we already have Jupyter notebook installed and we will be starting with the topic modeling session.

(Refer Slide Time: 00:41)



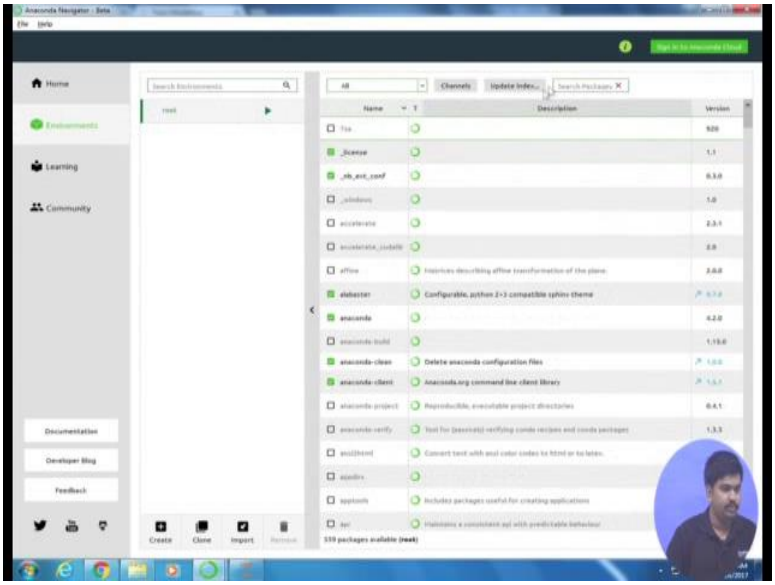
We have started Jupyter notebook and here we will be discussing about topic modeling. So, I assume you remember, what is topic modeling? So, it essentially is a way to model s corpus or a set of documents where all the documents in the collection share the same set of topics, but each document exhibits those topics in different proportions.

We essentially assume that we have a corpus and in that corpus that contains multiple documents, every document share a common set of topics. Now given those topics, they can be varying in different proportions in each of these documents. Now each word in the

given document is basically drawn from one of those topics. So, here we assume that every word that you see in a particular document that comes from the distribution of a particular topic and based on the document wise topic proportion you get to see those words.

Now, to infer all this, all we have is basically the observations or bay or the corpus itself. So, to run this codes that we are discussing today you might be requiring to install a library called Gensim. So, I assume that everyone is familiar with anaconda navigator.

(Refer Slide Time: 02:15)



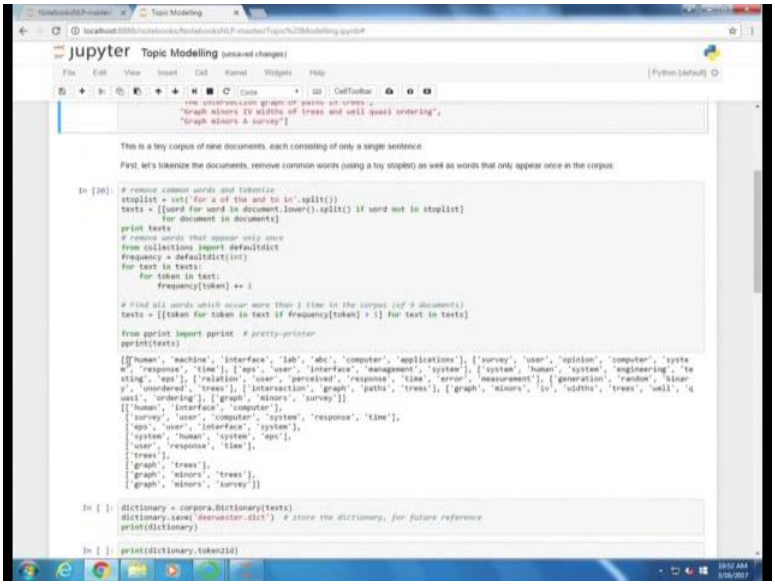
Once we use anaconda navigator have to browse to environment and look for the library section here. So, here just select on the all option and look for Gensim. So, we have already preinstalled Gensim. So, here you should ideally be not seeing the tick mark. So, select this and install the library.

Once the library is installed, we will import the library and for this tutorial, we will be assuming a very small corpus which is basically a collection of 9 different sentences. So, here is a tiny corpus of 9 documents and each document essentially contains only one sentence. So, in order to view this corpus to the model we have to do some essential pre processing. So, we will be showing what all pre processing needs to be done. So, we

need to convert this to a suitable bag of words format and we will be showing how to do this.

Here are some pre processing steps that we are going to do. So, for this simplistic corpus we assume the only set of stop words are these few words, but in practicality you might have to consider other stop words as well, but there are pre defined sets that you can obtain we will show how to obtain those lists. So, we make a list of all those stop words we also make sure that all the words are converted to lowercase. So, here what we do is that we take each sentence in the documents and then we split it based on the space. So, we do tokenization again based on a very simple notion that if there is a space between 2 characters we assume that they belong to 2 different words.

(Refer Slide Time: 04:37)



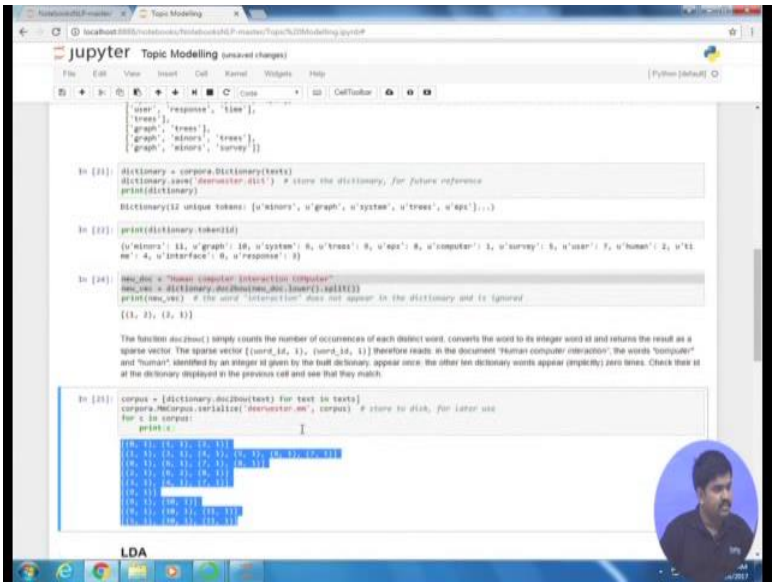
Since our corpus is something that we are doing for representation purposes these simplistic method will help you in doing this, but again you can rely on any standard Tokenizers for this task. So, text will essentially contain all those individual words see if you see here the word the sentence human machine interface for lab, A B C computer applications here that document is now represented as the list of words where you can find the H in human being has been turned in this small h or the lowercase h, the stop word for is removed and we essentially got a list of list where each list shows a document

with it is important words.

After doing this what we find is that we find the count or the frequency of a particular word occurring in the entire corpus for example, the word human is appearing once in document one and it also appears one time in document 4. So, the total occurrence is 2 for the word human. So, this particular chunk of code basically tells you how to find the frequency of the individual words in your corpus once that is done, we filter those words which appear more than once. So, we keep those words only those words which appear more than once in our corpus and we remove them.

What you find here in the second printed statement are those words which are filtered after this particular criteria is applied. So, we can see that machine which was appearing only once in the entire corpus is removed and similarly other words like interface etcetera. So, once we have this, most often, what happens is that when we have large corp error, we may not be doing it every time like we may not be doing the pre processing every time because it itself might be time consuming.

(Refer Slide Time: 06:42)



Once we have this pre processed state, we save it into any of the suitable formats. So, here Gensim provides a suitable format for saving this text called dictionary and we save

it in that format, see if you see here in Gensim, we use the dictionary method to convert the text to a suitable internal representation and we find that there are 12 unique tokens for the entire corpus, once we have that for ease of handling the data we convert each word to a unique individual representation. So, every unique word will be given an individual ID.

There arbitrarily provided there is no rational for which number goes to which word. So, once we apply this operation dictionary dot token 2 ID which is an in built attribute for this model as given by Gensim you can find that the words like minus is given an id eleven graph is given an id ten and so on. Now we assume that whatever words are going that we are going to use will come from only one of these words.

Now, let us take a new document which is human computer interaction. So, I have a new sentence or a new document human computer interaction. So, I have to apply all those pre processing that I have applied to the corpus, since there are now stop words I do not apply the stop words filtering explicitly, but it is also assumed to be that. Now we convert it to something called as b o w or bag of words representation. So, a document is now going to be represented in terms of the bag of words representation which is very similar to this one.

We can find that you get a list with 2 petals which show 1 comma 1 and 2 comma 1. So, what essentially is represented here? So, essentially what you can find here is that we have 3 terms human computer and interaction, but unfortunately interaction is a word that is not available in our dictionary. So, we ignore that word when we convert this document to a vectorial representation. So, for all practical purposes, you can assume your document to be human computer as this is what it is going to be fed to the topic model.

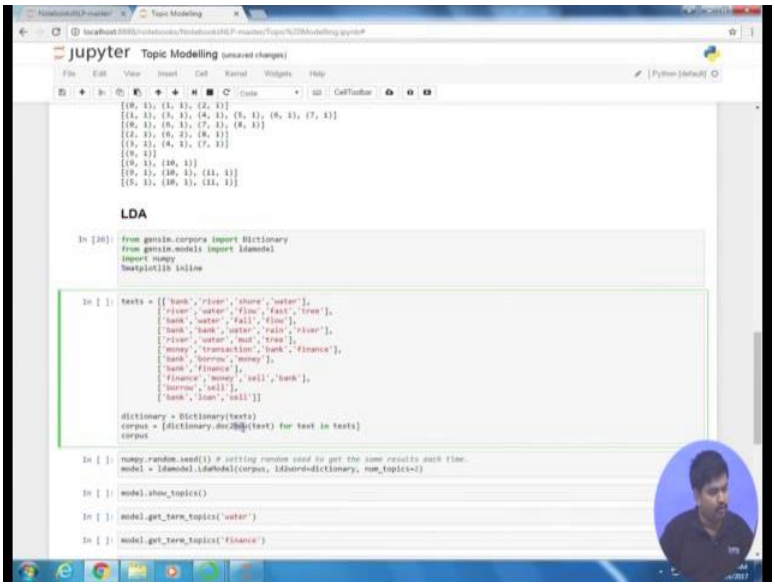
Now, what is this first entity that is 1 and 2 represent they basically represent each of the word human and computer see, if you see the identifier for human is 2 and the identifier for computer is 1 and what you see on the right is the number of times a particular word is appearing for example, if I add another term computer again. So, it is human computer

interaction computer the word with ID 1 which happens to be computer as increases frequency. So, this is at basically frequency count of each unique word that basically represents the documents.

As we have saved the dictionary, we can also save these representations also in a serializable format and we can store it so that we can load it, later whenever we require that so, these are all those lines sentences that we were dealing with here, they are first converted to a list format with it is relevant words. Now we convert it to the individual usage or the individual identifier along with it is frequency it. So, happen that in this corpus we do not have any of the word repeating twice other than this particular word that is the 6th word in the 3rd sentence. So, word 6 is system and in 3rd sentence, yes system; human system e p s. So, that is now reduced to 6 comma 2.

Till now we have not done anything specific topic modeling or what we call as LDA for this particular instance, what we have done is that we have used a means to represent in a format that is much more easier to handle for the model.

(Refer Slide Time: 11:27)

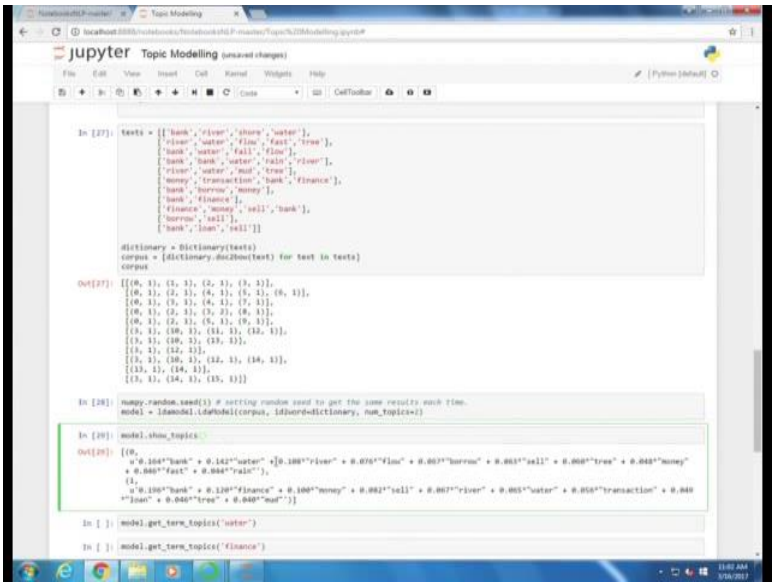


Now, we invoked the LDA model or the latent Dirichlet allocation model for the topic models as you remember, there are different variations of topic model that has been used

as you have seen different methods in your class like the sequential mode sequential topic model and the relation topic model here we will be showing only the simple LDA model.

I import this library now let us assume another corpus where all the pre processing are initially done. So, we have something like a new corpus. So, again a tiny corpus with few sentences and the stop words are removed all the pre processing like that we convert it to the dictionary and we convert it to the bag of words format for each of this document. So, here we look at the representation for this corpus just to make things clear we have created a new dictionary based on the words in this corpus.

(Refer Slide Time: 12:35)



Now, comes to the LDA model. So, in the LDA model what we have to essentially provide to the system is the number of topics. So, we assume that the user or the programmer has a he already he is aware of the number of topics that the certain collection of corpus is going to contain and this should be provided as a parameter or as an argument to the model. Now if we see what are the other possible options that you can use to customize this particular function, you can find what are the values the hyper parameter alpha can be given; the hyper parameter eta or the beta as you know it.

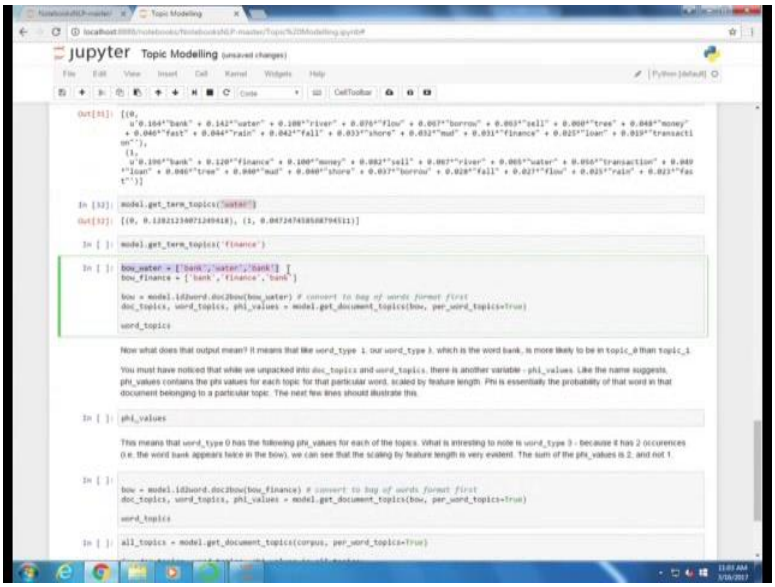
The decay and number of iterations and other convenient parameters that you can use, so

the alpha and eta are the important ones, others are from mostly from programming convenience. So, let us run this model. So, now, we have added this corpus into a doc 2 bow representation and we have provided that this purpose to the LDA model to get a model. This is stored in the variable named as model now if we see we can find since we have already told the system that there will be only 2 topics, it has provided 2 topics to us and it is showing the top terms that the topic represents. So, the topic is essentially a collection of words which is gathered from the different documents and you can see with what probability each word is belonging to that particular topic.

We can also find the same word might belong to multiple topics here the term bank is having a probability of 0.164 and here the term bank is having a probability of 0.196, we can find other words also repeat here like water has 0.065 in topic 1 or the topic 2 and water has a probability of 0.142 in the first topic which is index at 0, now if you see ideally the probability should add up to 1. So, all the words which are represented here when we add up the probability it should give a probability of 1.

But if you add up these values it is not getting adding up to one. So, here we have an option to show, what are the numbers of words or topic to be shown? So, here the model has shown only top 10 words for the given topic. So, in practical scenario we often use number of topics which are more than 10 also like that it can be 50 or 100 topics in that case this function just shows 10 topics; some 10 topics out of the 100 or 50 topics now since we have a small corpus and I am aware that the number of words is going to be less than 20, I just add the parameter number of words is because the argument number of words is equal to 20 and we can find all almost all the words that is present in the particular topic if we increase the topic it does not change which means these are the only number of words in that particular topic.

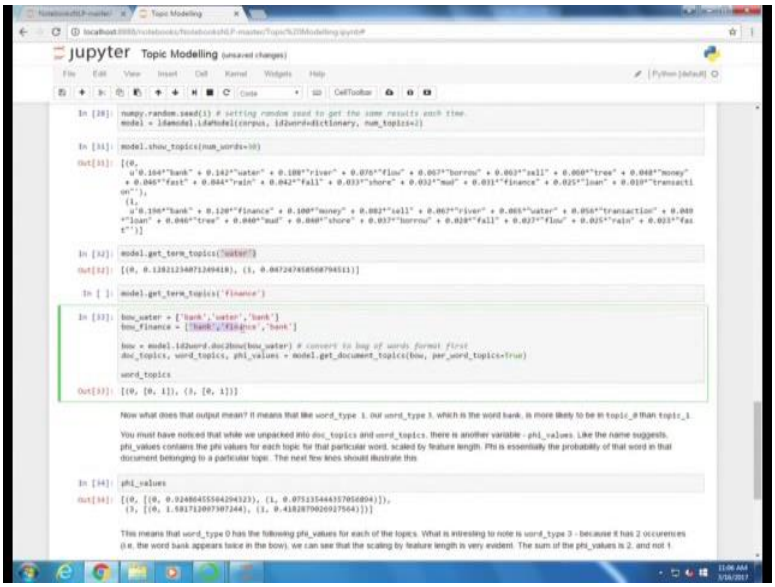
(Refer Slide Time: 16:24)



Now, if we want to know what is the probability of a particular word? So, we get the term topics from this function. So, we can find that water has a probability of 0.128 and a 0.047 in either of the topics, now let us see a new document that is bank; water bank which was not provided to the model while it was learning the probability distributions. So, we have a new test document or a new document where we have bank water bank and we first convert it to the b o w representation, once we have that here what we find is that the word 0 and word 3. So, both the terms which are like bank and water both the terms are more likely to belong to topic 0. So, here even if bank has a higher probability at topic one may be because of it is co occurrence with water which has as a high score in topic 0 both bank and water are given a higher probability of being in topic 0 for this particular document.

As you are aware in topic models given a word it can belong to multiple topic distributions and once you find the particular word in the document it might be coming from any one of those topics.

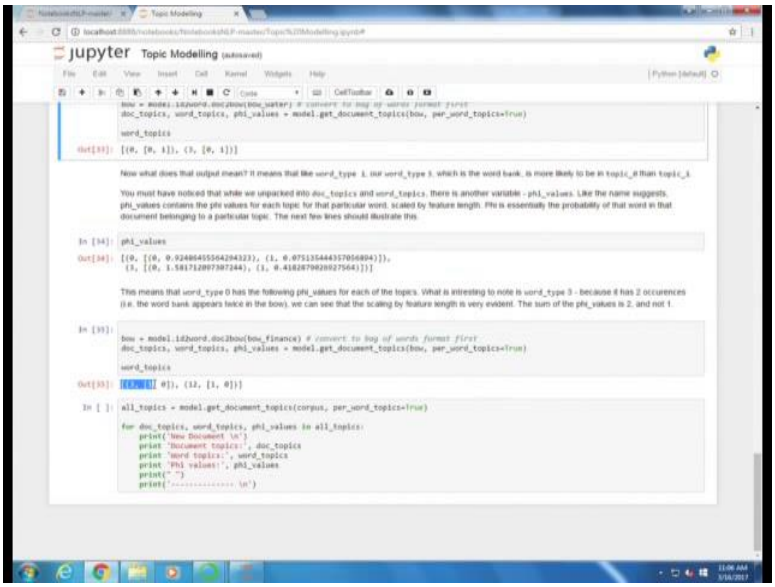
(Refer Slide Time: 18:22)



Now, if you see here the 5 values which is essentially the probability of a word in that document belonging to a particular doc topic. So, the 5 value essentially shows, what is the probability of each word belonging to each of those topics? So, if you see here the word type 3 when you add up the values here it has a probability it shows a score not a probability, but a score which adds up to 2.

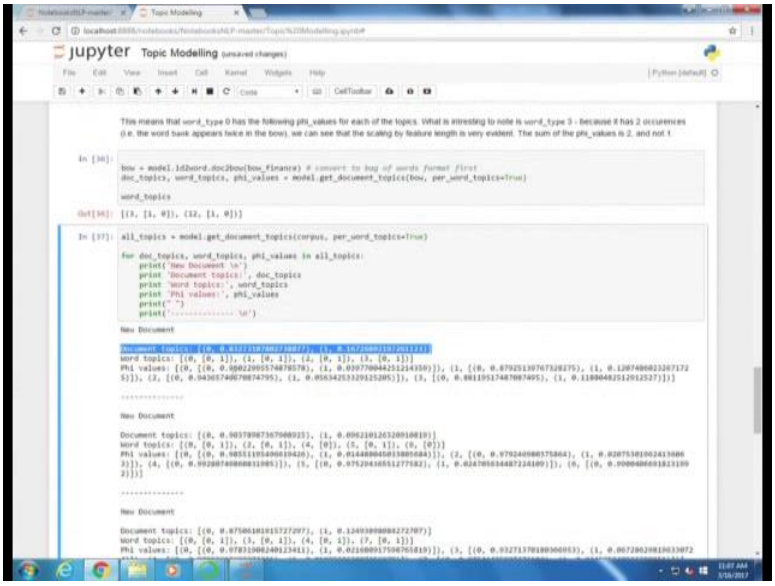
If the sum of the 5 value is 2 which is and not 1 and this is indicative of the scaling by feature length. So, this always does not happen, but it is an indicative aspect here now what we can see here is that we have taken the second document here which is b o w finance. So, once we have bank finance bank as our document we can find that both the words bank and finance have a higher probability of belonging to topic 1. So, here if you see the word 3 which was bank had a higher probability of belonging to topic 0.

(Refer Slide Time: 19:41)



But in the second document, this is the model assumes that the word bank comes from topic 1 not topic 0, this is again assumed to be because it co occurs with the word finance. So, we should be talking about the financial institution bank rather than the river side or river bank. So, this becomes evident using this model.

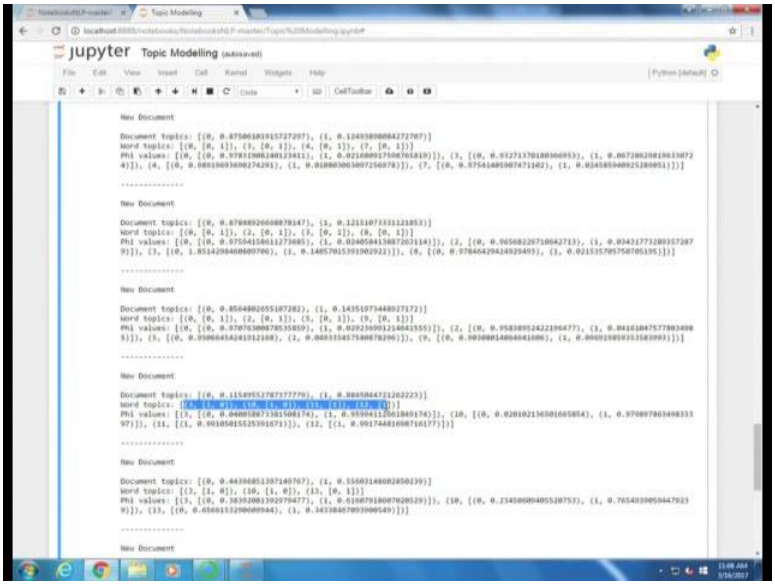
(Refer Slide Time: 20:08)



Now, let us see how what are the different values that we get for each of these documents. So, we can find that the document topics. So, the first document which is essentially bank river shore water if the document representation has a higher tendency to or has a higher proportion of topic 0 rather than topic 1.

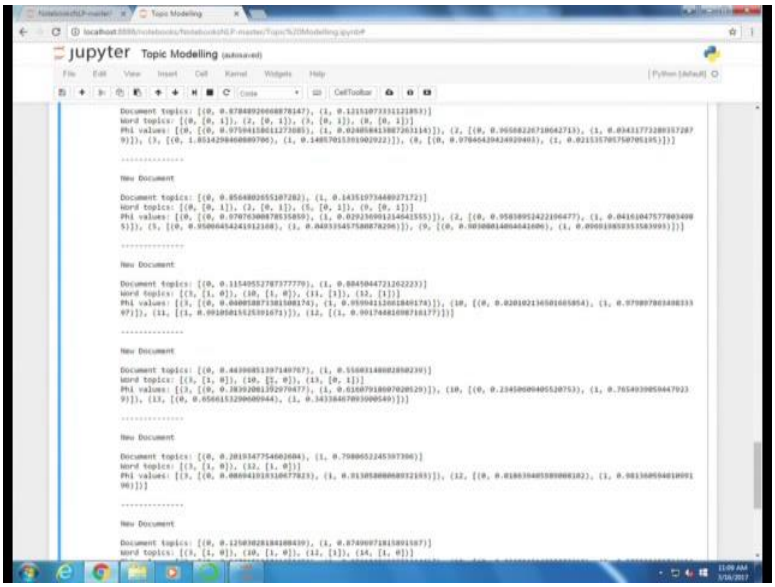
Now, individual words were all the 4 words 0, 1, 2 and 3 have a higher tendency to belong the topic 0 the 5 values which is like per the per word distribution for each word in that document also shows the similar branch. Now if you see next word per document which is the document river water flow fast tree what we can find is that again it is having a higher probability of belonging to topic 0 than topic one and we can find that some of the words have no, no occurrence or no probability of occurrence in topic one like the 4 and 6.

(Refer Slide Time: 21:28)



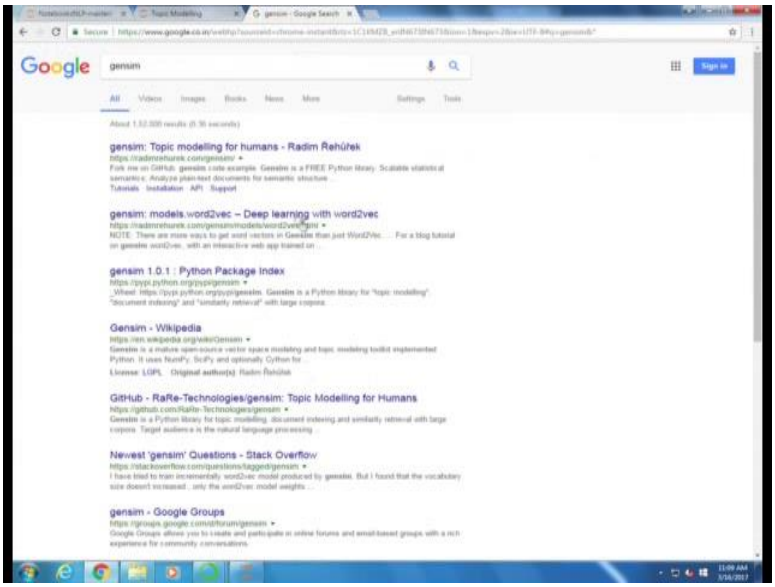
Now, if you look into say then one of those documents where the it is probability of belonging to topic 1 is 0.88 while that of belonging to topic 0 is 0.11 and we can find almost all of the words have a higher probability to belong to a topic one it is also possible that suppose similarly for this document.

(Refer Slide Time: 22:01)



We have more or less similar weights for each of the topic like topic 0 it is 0.44 and topic 1, it is 0.55 where the confidence of this document belong to a particular topic is not that high to be decisive enough we can find that 2 of the words have a probability of belonging to topic 1 while that last third word has a higher probability of belonging to topic 0 and we can find the comparative values of each of those doc words with this topic proportion. So, this is how we analyze a corpus using topic modeling or the LDA.

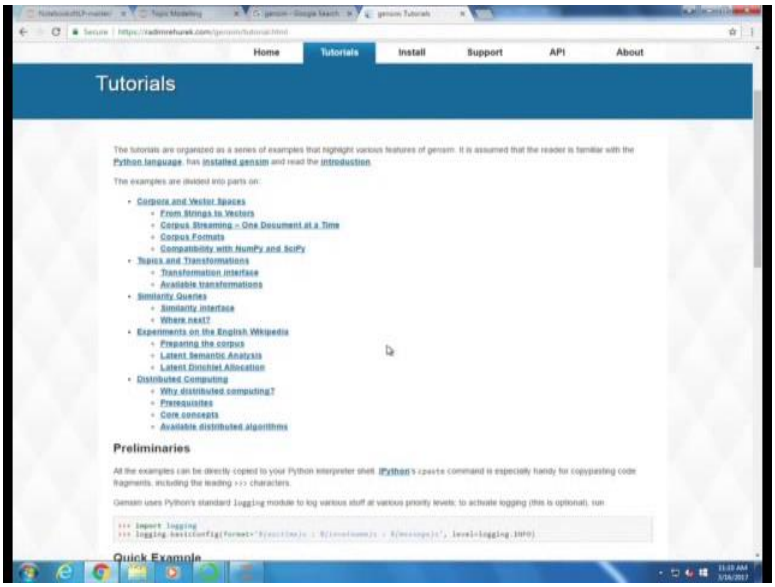
(Refer Slide Time: 22:55)



Gensim essentially provides a suit of packages that helps in different topic modeling task and they have a set of comprehensive tutorials. So, the tutorial that you have been seen here has been combined from multiple tutorials from Gensim.

(Refer Slide Time: 23:11)

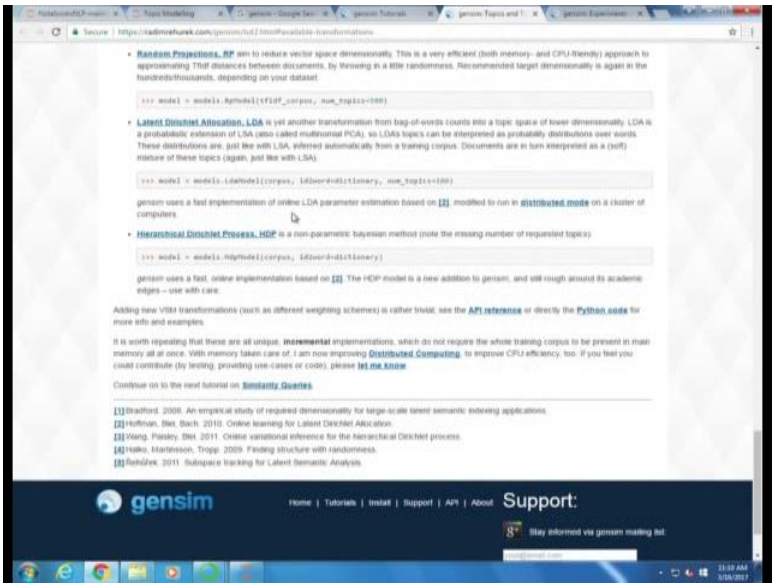
(Refer Slide Time: 23:13)



You can have a further look into different topics here, then in addition to LDA provide other topic models as well they have they show topic models e pitching load HDP which is hierarchical display process and they also have a model for sequential LDA. So, we can have a quick look into one of those.

(Refer Slide Time: 23:51)

(Refer Slide Time: 23:48)



Here we can find different transformations or different ways of representing a document what DF, IDF and LSA or LSI are some of the pops they use to be some of the popular ways of representing a document in a vectorial representation and we have LDA which is what we have seen we have HDP. So, HDP is a non parametric version where you do not need to provide the number of topics as well that is it is inferred by the model itself. So, this is pretty much for the topic modeling tutorial.

Thank you.