Applied Natural Language Processing

Prof. Ramaseshan Ramachandran
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Lecture – 29
Naive Bayes Algorithm for Classification

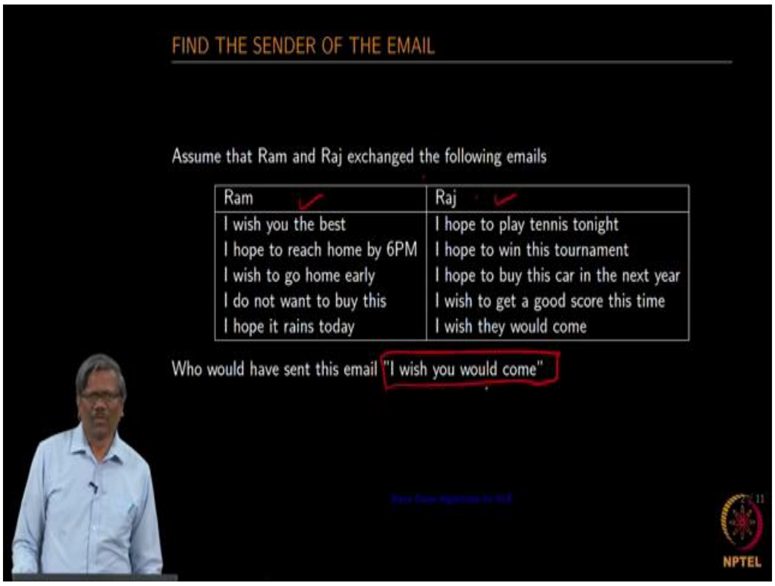
(Refer Slide Time: 00:15)

Naive Bayes Algorithms for NLP

Ramaseshan Ramachandran

So, we have been looking at the probability and statistics and then try to see how that could help us in terms of solving some of the problems in NLP. So, in continuation of that I am going to be looking at Naive Bayes Algorithms for NLP. So, I am going to give only the very high-level overview of this and you may want to go on and then read this in detail in the reference books, all right ok.

(Refer Slide Time: 00:45)



FIND THE SENDER OF THE EMAIL

Assume that Ram and Raj exchanged the following emails

Ram	Raj
I wish you the best	I hope to play tennis tonight
I hope to reach home by 6PM	I hope to win this tournament
I wish to go home early	I hope to buy this car in the next year
I do not want to buy this	I wish to get a good score this time
I hope it rains today	I wish they would come

Who would have sent this email "I wish you would come"

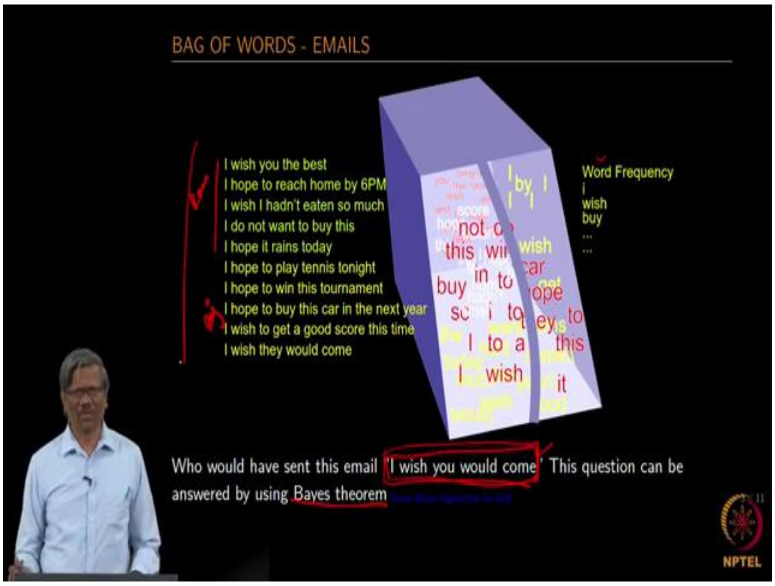
So, here what we are going to show is we are going to be showing how our Naive Bayes can be used for classification. So, it would act as a classifier and then we will see how we can utilize that in terms of classifying two different sets of data, all right. So, in this case I am going to be considering emails, ok. So, let us consider her two people Ram and Raj, who are exchanging emails. I have just taken some small portions of the email and I have listed here as documents, ok. So, this set is by Ram and this set is by Raj, ok. And then these are very simple text that considering as an email message. And then at the end, I want to find out who had sent this mail that contains so many words, ok, assuming that this mail has come without any name or anything and you are asked to find this.

So, this is the situation in various other classification applications as well. So, instead of emails you may want to look at spam, right, so whether a mail is a spam or a or not spam. So, in that case you know when the incoming mail is not known to be spam initially, right, but by looking at the contents of the data you may want to target to spam or not spam.

So, in this fashion we are going to be considering, we are going to be considering this as the incoming email and we want to classify whether it was sent by Ram or Raj, ok. So, what do we do in the statistical modeling? We start looking at the numbers of or the occurrences of words, right. So, that is what we want to do. Can we just do it without any help from the probability and statistics to find out whether this mail has come from Ram or Raj?

There is a possibility. Since, the sizes of the emails are small, right, so in this class as well as in this class it is very small, so we may be able to find out how many times Ram had sent an email containing the word I wish you would come and so on, and then count how many times Raj had sent using those words. And maybe that would give approximately who had sent it. But that is not going to help us you know in the cases where the contents are huge. So, let us look at how this could be marked into the Naive Bayes and then let us see how we can solve step by step right, ok.

(Refer Slide Time: 03:42)



BAG OF WORDS - EMAILS

I wish you the best
I hope to reach home by 6PM
I wish I hadn't eaten so much
I do not want to buy this
I hope it rains today
I hope to play tennis tonight
I hope to win this tournament
I hope to buy this car in the next year
I wish to get a good score this time
I wish they would come

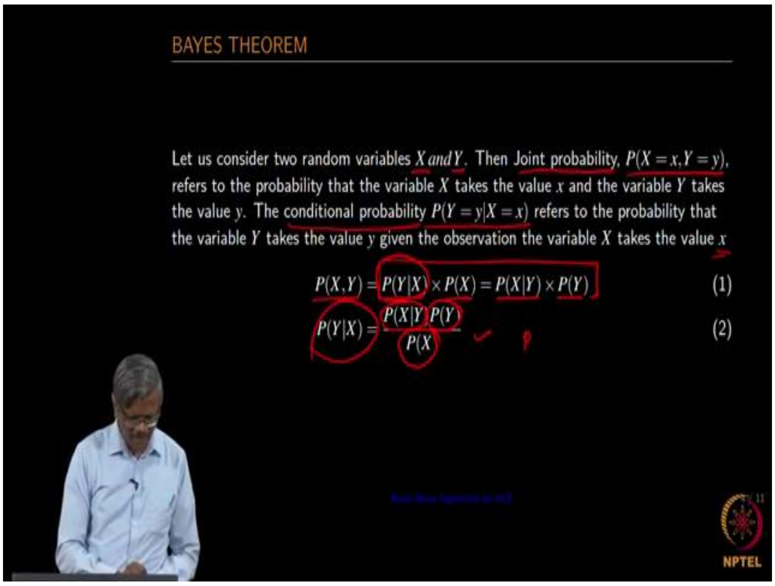
Word Frequency
i
wish
buy
...
...

Who would have sent this email 'I wish you would come' This question can be answered by using Bayes theorem

So, before that, I like you to remind you that we are going to be using a bag of words. I am sure you remember what bag of the word is, just represented this using some small simple graphics. So, we have the same set of emails coming in from Ram and Raj here, right. The bag of words you know well that it just takes each and every word and then we just dump it into the back, so there is no organization with respect to this sequence of words that are getting into the bag. So, it is all jumbled in some fashion.

The output of this would be for every word what is the frequency because that would be the output of the bag of words correct. So, again in this case we are going to be looking at the Bayes theorem to find out whether the mail had come from either Ram or Raj. So, a bag of words is going to be the part and parcel of this algorithm for us, ok.

(Refer Slide Time: 04:52)



BAYES THEOREM

Let us consider two random variables $X\,and\,Y$. Then Joint probability, $P(X=x,Y=y)$, refers to the probability that the variable X takes the value x and the variable Y takes the value y. The conditional probability $P(Y=y|X=x)$ refers to the probability that the variable Y takes the value y given the observation the variable X takes the value x

$$P(X,Y)=P(Y|X)\times P(X)=P(X|Y)\times P(Y) \tag{1}$$

$$P(Y|X)=\frac{P(X|Y)P(Y)}{P(X} \tag{2}$$

So, before getting into our algorithm part, let us look at the theory aspect first and then see how we can map the theory into the email classification problem. And later start doing, later we start solving this particular problem, ok.

So, let us consider two random variables. So, I am just taking the theory part of that initially. Then you know what a joint probability is right. So, it is given like that, so this refers to the probability that the variable X, capital X takes the value small x and the variable captain Y takes the value small y and then you also know what a conditional probability is, ok. It is represented as this it refers to the probability that variable Y takes the value small y given the observation of the variable capital X taking the value small x, ok.

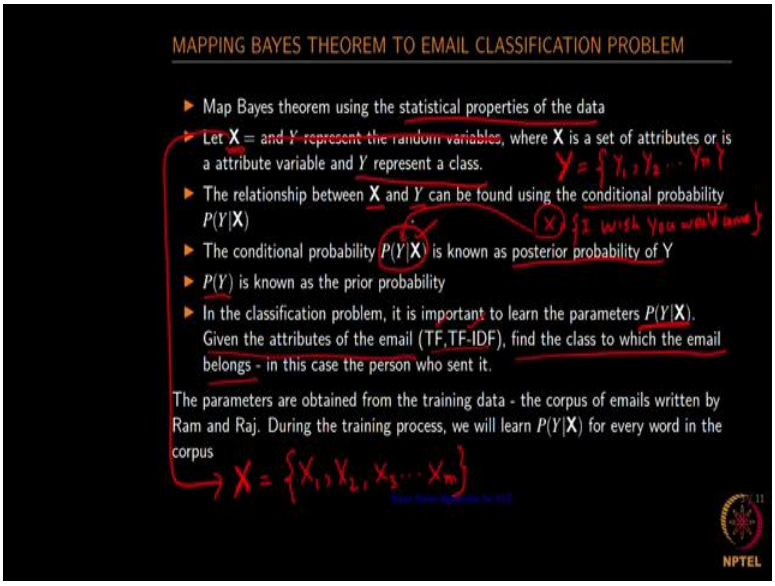
And you know that the conditional probability can be represented rather the joint probability can be represented as the product of conditional probability, right and so, same can be represented in this fashion. So, taking these two right and then trying to form an equation,

$$\mathrm{P(X,Y)} = \mathrm{P}(\frac{y}{x} \times P(X) = P(\frac{x}{y} \times P(y)$$

$$\mathrm{P}(\frac{Y}{X}) = \frac{P(\frac{X}{Y})\times P(Y)}{P(X)}$$

Where y= ($y_1\ y_2\ y_3 y_4\ \ \dots\dots\dots y_n$)

(Refer Slide Time: 06:46)



MAPPING BAYES THEOREM TO EMAIL CLASSIFICATION PROBLEM

- Map Bayes theorem using the statistical properties of the data
- Let **X** = ~~and Y represent the random variables~~, where **X** is a set of attributes or is a attribute variable and Y represent a class. $Y = \{Y_1, Y_2 \cdots Y_n\}$
- The relationship between **X** and Y can be found using the conditional probability $P(Y|\mathbf{X})$
- The conditional probability $P(Y|\mathbf{X})$ is known as posterior probability of Y
- $P(Y)$ is known as the prior probability
- In the classification problem, it is important to learn the parameters $P(Y|\mathbf{X})$. Given the attributes of the email (TF,TF-IDF), find the class to which the email belongs - in this case the person who sent it.

The parameters are obtained from the training data - the corpus of emails written by Ram and Raj. During the training process, we will learn $P(Y|\mathbf{X})$ for every word in the corpus

$X = \{X_1, X_2, X_3 \cdots X_m\}$

So, now how do you map this? So, now, we have obtained two parameters right, this. So, I am also going to let you figure out why I am not going to be using P (x) beyond this point, ok. So, I will not be considering P (x). So, you have to go and then look at the reference books to find out why a probability of X does not really matter in this case. So, we are going to be mapping this Bayes theorem to the email classification problem. Let us look at some of these statistical properties of the data. So, that is very important, right. So, we need to understand the data in order for us to really do some progress in terms of finding out what is in it what can I do with that and how can I solve the problem that is all I have.

So, let us X be a variable that contains variable parameters x_1 ,x_2 ,x_3 ,x_4 …… x_n .

So, let us Y be a variable that contains variable parameters y_1 y_2 $y_3 y_4$ ……… y_n

So, in this case since we are going to be finding this is the posterior probability of Y and P (Y) is known as the prior probability, ok. So, we are doing it from here, ok. So, in the classification problem it is important to learn the parameters of this conditional probability. So, given the attributes of the email. So, you can either use the term frequency TF-IDF or any combination of weighted TF-IDF or just TF alone and so on, right. It could be just the term or you can use Ingram also to find out the parameters or the attributes of the email, ok. So, what we want to find out is given these set of data and of having all the attributes for each of the data set that we are going to be using, we want to find out to which class a new document that comes in belonged, right.

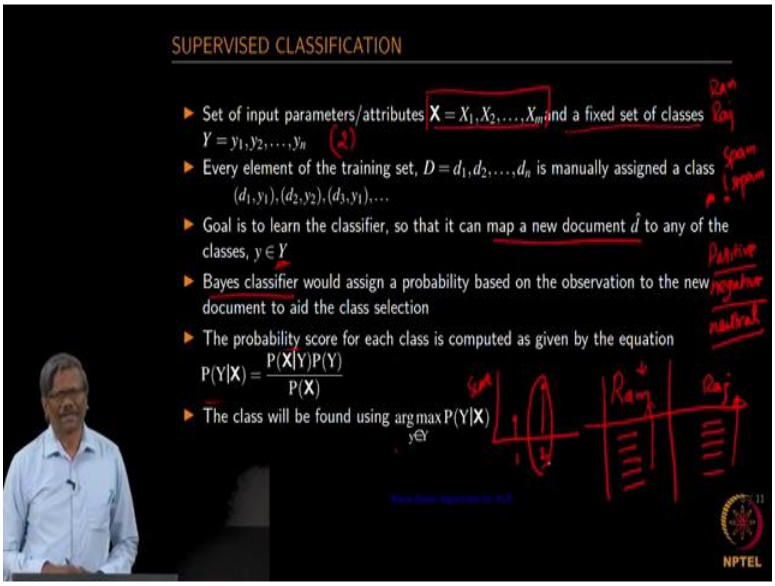
So, given the attributes of the email right and a given the prior probability, so we want to find the class to which the email belongs, that is in this case who is going to be sending it. Getting the point, right. So, now, what we have? We have a set of data let us call it as the training data set which contains all the emails from both parties or we can call those as two different classes of emails and then every email that you have contains a list of words right that we can call as the attributes.

And then based on the combinations of the words that are present in the email we want to classify whether it belongs to the class one which is Ram or Raj for class two, right. So, that is what we want to find out. So, now, looking at those contents of the email how can I take the attributes out of that that is their TF and TF-IDF or bigram frequencies all those things would come into play, right.

So, now we have the data set that is been extracted from the training samples. So, now, how can I now go and then find out who had sent an email in that case. Which one is that? So, we want to find out based on the training data that we have or the labels that we have created we want to find out who had sent this particular mail, ok.

So, the assumption is that the words that are there in this are part of this, part of the data set, right. So, if they are not there, so what do we do? So, we look at that little later, ok. So, now, is this clear. So, we want to map the Bayes theorem to the classification problem. So, it is pretty simple. So, we are trying to find the conditional probability of a set of attributes belonging to a class, that is this X could be you were, ok. So, we want to use this X and then find out which class it belongs to, ok, all right.

(Refer Slide Time: 13:40)



SUPERVISED CLASSIFICATION

- Set of input parameters/attributes $\mathbf{X} = X_1, X_2, \ldots, X_m$ and a fixed set of classes $Y = y_1, y_2, \ldots, y_n$
- Every element of the training set, $D = d_1, d_2, \ldots, d_n$ is manually assigned a class $(d_1, y_1), (d_2, y_2), (d_3, y_1), \ldots$
- Goal is to learn the classifier, so that it can map a new document $\hat{d}$ to any of the classes, $y \in Y$
- Bayes classifier would assign a probability based on the observation to the new document to aid the class selection
- The probability score for each class is computed as given by the equation $$P(Y|\mathbf{X}) = \frac{P(\mathbf{X}|Y)P(Y)}{P(\mathbf{X})}$$
- The class will be found using $\arg\max_{y \in Y} P(Y|\mathbf{X})$

So, as I mentioned earlier now we have a set of input parameters X which can be represented like this and we have a fixed set of classes, so in this case it is two, right. So, if you look at these spam based, one it is going to be spam not spam. And then if you look at the movie reviews it is going to be positive, negative, or if it is about the reviews of any product it is going to be positive or negative, the third class also could be neutral, ok. So, the same thing could be extended to multi-class as well, ok.

So, now we are going to be looking at various attributes of X with varying lengths and then we have a fixed set of classes Y which is $y_1\ y_2\ y_3 y_4\ \ldots\ldots\ldots y_n$)

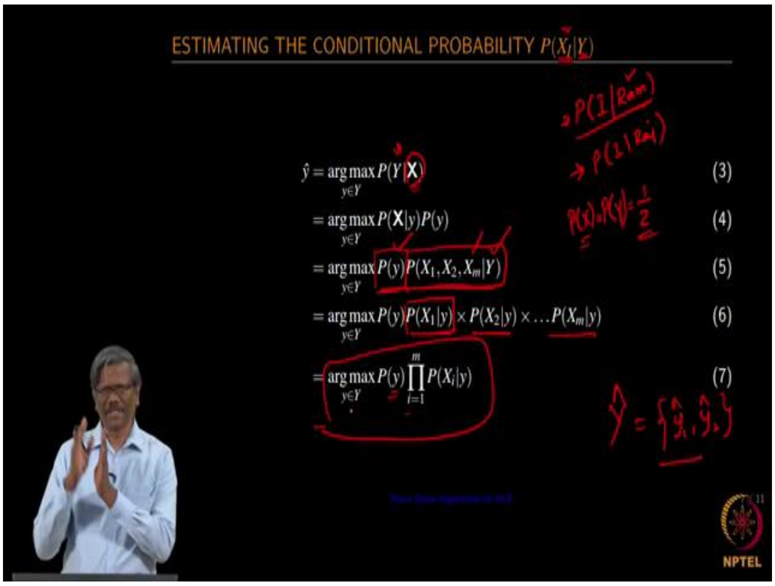
in this case as I mentioned we have only two classes, ok. So, every element of the training set. So, we can consider that as a document 1, 2, 3, 4, 5 and so on. This is manually assigned a class, right. So, we have a set of emails. So, we already have manually assigned those values, right, to this class. So, we have set of emails and we have assigned each one a class, so that the manual assignment that we have made, ok

So, the goal is to learn the classifier based on the data that we are going to be having, so that it can map a new document d to any of the classy belonging to capital Y, right. So, when we get the new document, the document is I wish you would come here we are going to be mapping that particular document into one of these classes. So, that is the idea.

So, what we are going to be doing this? In this case we are going to assign a probability to each of the classes. So, we need to find out what is the probability of that particular document belonging to Ram or Raj, and then based on the values we are going to be picking up whether it belongs to the class Ram or Raj, ok. So, we are going to be computing the probability score for each class using this equation, right. So, this class will be found using the arc max. For example, in this case there are only two values.

So, we will have values corresponding to the let us say this is the probability score, this is the score, ok. So, based on this code we will find out. So, this belongs to 1 and 2. So, we will find out which one has a higher value in this function and then pick up that index as our a class, ok, all right.

(Refer Slide Time: 17:13)



ESTIMATING THE CONDITIONAL PROBABILITY $P(X_i|Y)$

$$\hat{y} = \arg\max_{y \in Y} P(Y|\mathbf{X}) \quad (3)$$

$$= \arg\max_{y \in Y} P(\mathbf{X}|y)P(y) \quad (4)$$

$$= \arg\max_{y \in Y} P(y)P(X_1, X_2, X_m|Y) \quad (5)$$

$$= \arg\max_{y \in Y} P(y)P(X_1|y) \times P(X_2|y) \times \ldots P(X_m|y) \quad (6)$$

$$= \arg\max_{y \in Y} P(y)\prod_{i=1}^{m} P(X_i|y) \quad (7)$$

$P(I|Ram)$

$P(I|Raj)$

$P(x) \cdot P(y) = \frac{1}{2}$

$\hat{y} = \{\hat{y}_1, \hat{y}_2\}$

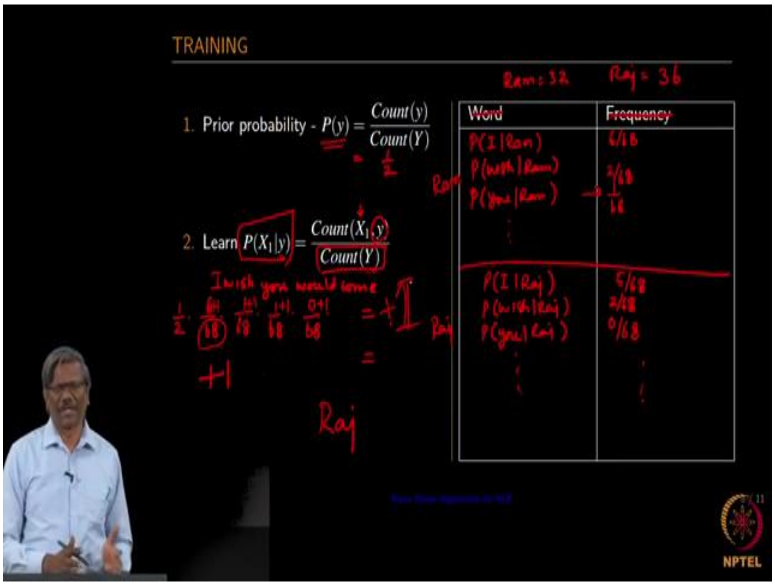
So, how do we estimate the conditional probability? So, for each attribute that we have given the class we will find the conditional probability, ok. So, for example, in this case we have I right, probability of I given Ram. So, this is what we are going to be finding every time for every word in the cards, as I mentioned earlier. So, this is going to be for one word, right and then for every class since it contains about 4 or 5 words, so we have to find out that conditional probability and then multiply that.

So, when you do that you have estimated from the data all of these right for every class and which can be written in a simplified fashion and they say, right. For us to find that, so we have a few parameters that we can estimate from the data, so one is the prior probability for that class. So, we are going to be finding. Since we are going to be having only two classes here, so and then the sets of emails are the same, so we have P of Y equal to half, ok. So, if you have more number of classes that depend on how you have actually classified the documents and then it goes by the document counter as well then you take this out.

So, in this case since the number of documents is the same for both classes Ram and Raj, this is half, ok. So, this can be estimated. And then since the class is known, so we are going to be assigning the probability for each and every variable. When you expand this is nothing, but the conditional probability for that was given the class. So, for every class and every word in that class, we are going to find the conditional probability and then multiply that.

So, we get in this case we are going to get Y hat will have two values got it. So, that is where, so than for every class, once you find that and then use the arc max function to find out which index has the highest value and pick that up, ok.

(Refer Slide Time: 20:21)



TRAINING

1. Prior probability - $P(y) = \frac{Count(y)}{Count(Y)}$ $= \frac{1}{2}$

2. Learn $P(X_1|y) = \frac{Count(X_1, y)}{Count(Y)}$

I wish you would come

$\frac{1}{2} \cdot \frac{6+1}{68} \cdot \frac{1+1}{68} \cdot \frac{1+1}{68} \cdot \frac{0+1}{68}$ =

+1

Raj

Ram: 32 Raj = 36

~~Word~~	~~Frequency~~
P(I | Ram)	6/68
P(with | Ram)	2/68
P(you | Ram)	1/68
⋮	
P(I | Raj)	5/68
P(with | Raj)	2/68
P(you | Raj)	0/68
⋮	⋮

So, let us take this example and then see how we can compute this. As I mentioned earlier this is going to be half, right. So, we need to learn the parameters given the class for every word to find the probability, ok. So, it is nothing but the count of x_1, in that class and then the total number of words in the entire corpus. So, what do you do here is there are two sets here, right. So, we have 1 to 5, 1 to 5. So, you combine all of this and then count the number of words that is what going to be the count here. And then for x_1 count the number of x_1 in one class, for example, in this case for Ram compute this and then later for Raj compute the same thing, but this reminds the same for the computation until, ok. So, I need the space.

So, what I am going to do is I am going to just copy some of those computations that I have done already. So, let us start with, ok. So, this value is going to be 6 by 67. I think the number of words for Ram is 32 and Raj is 36, ok. So, in the same fashion, you do for all the words and this will be 2 by 67, let me use a smaller sorry 68 and so on, right. So, you do this for Ram and then now do this for Raj. So, this will be 5 by 67, sorry, this should be 68, we got one 0 here, right and so on. So, once you learn these parameters, so now, we have a table that contains all those values, right.

So, now, how can I use it to find who had sent that mail. So, I; so, if you want to find out this see we are going to be using this formula, right. So, we are going to be multiplying all the conditional probability along with the prior probability. So, when you do this we are going to have in the first case we have half that is equal to $\frac{P}{Y}$, right. I am sorry that is the prior probability for this class and then we have $\frac{6}{68}$, $\frac{1}{68}$ and for you this is going to be $\frac{1}{68}$ again. There is no word, I guess in this, so it is going to be 0 and so on, right. So, the value is going to be 0 because we have encountered a word that is not part of the dictionary that we have, right.

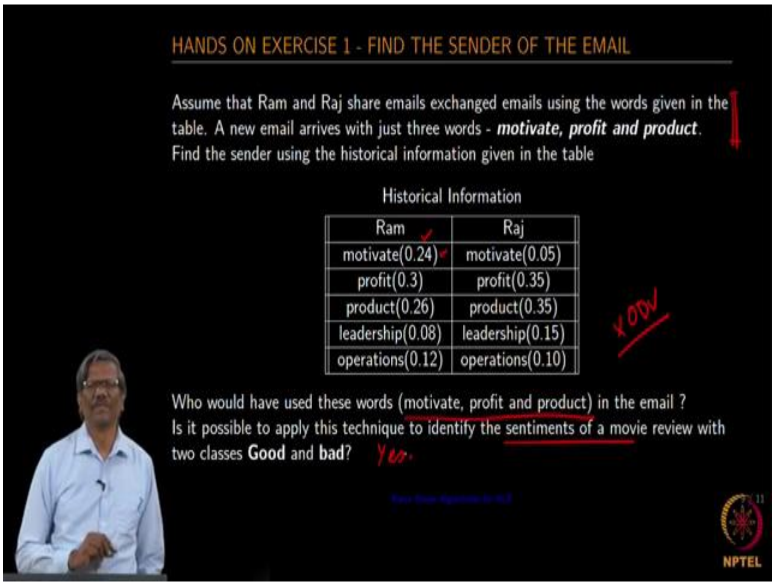
So, again if you look at the other one, so you will have the same problem. So, how can we solve this problem? So, using this it is not possible for us to find out who had sent the mail. So, in order to solve this problem what we do is, we do add one more thing to each one of the variables. So, in this case we will add instead of 6, we will add 1 here, 1 here, 1 here and so on. So, when you do that it is not going to become 0. So, we will always have some value associated with that. See the most important thing is to add 1, plus 1. So, for every word count you are 1 to that but do not change the count at the bottom, ok. So, this is a very practical problem that you will encounter.

So, it is not always possible to satisfy a constraint every time, right. So, in this case word is not part of the carpus, so, but it has come, but we still have to take care of that and then solve it. So, that is why you know these kinds of smoothing is done where you add plus 1 to each of the words, ok.

So, in this case, you know once you compute all this you will find that this mail had come from Raj, ok, all right. So, it is clear. So, every, you encounter always these kinds of problems in natural language processing. So, every corpus is not complete by itself, in terms of the vocabulary, in terms of the size, in terms of the lexical variety and so on ok. So, we deal with these kinds of problems as and when we proceed and it is very important to understand the domain in order for us to solve this problem.

So, that is why we always keep looking at the content before we really extract parameters whether it is the probability-based approach or neural net-based approach it is very important for you to have a good handle on the data, ok. And then it is also very important for you to find out what kind of parameters you want to use for any problem that you may encounter in the natural language processing.

(Refer Slide Time: 28:45)



HANDS ON EXERCISE 1 - FIND THE SENDER OF THE EMAIL

Assume that Ram and Raj share emails exchanged emails using the words given in the table. A new email arrives with just three words - ***motivate, profit and product***. Find the sender using the historical information given in the table

Historical Information

Ram ✓	Raj
motivate(0.24) ✓	motivate(0.05)
profit(0.3)	profit(0.35)
product(0.26)	product(0.35)
leadership(0.08)	leadership(0.15)
operations(0.12)	operations(0.10)

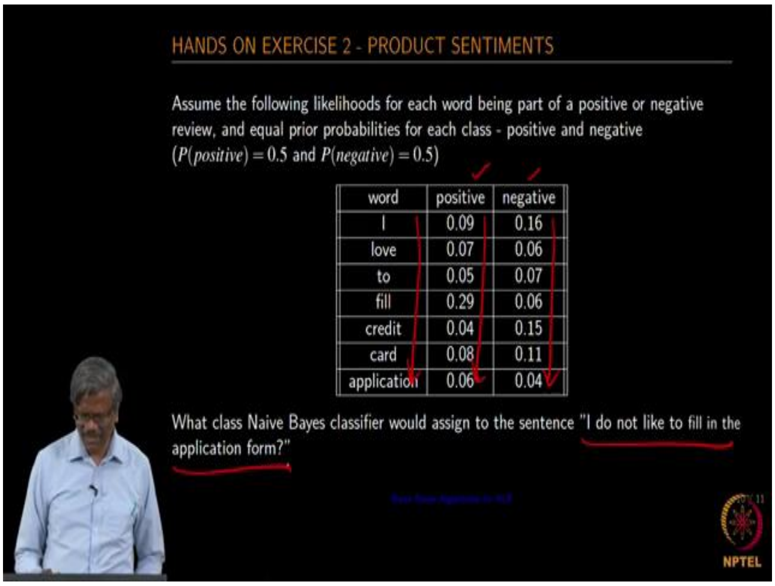
x 100

Who would have used these words (motivate, profit and product) in the email ?
Is it possible to apply this technique to identify the sentiments of a movie review with two classes **Good** and **bad**? Yes.

So, in this next slide what I have shown is a problem, ok. So, the problem for you. This, you may have to go and then either solve it by hand or use some python programming to solve this problem and I have given the attribute values. So, instead of you counting and computing all that, so I have compressed and then reduced the work for you. I have computed some values and then provided them here and this is the historical information that you may want to use.

Again there are only two classes, ok. So, what you want to find out is when the email comes with these three words, who all send this mail. So, it is very similar to what we have done earlier and I think you would be able to solve this quickly. So, it does not contain any OOV in this case, ok. So, you can also apply the same mechanism as I mentioned earlier that we can find the sentiments of a movie review or product and so on, right, ok.

(Refer Slide Time: 30:07)



HANDS ON EXERCISE 2 - PRODUCT SENTIMENTS

Assume the following likelihoods for each word being part of a positive or negative review, and equal prior probabilities for each class - positive and negative ($P(positive) = 0.5$ and $P(negative) = 0.5$)

word	positive	negative
I	0.09	0.16
love	0.07	0.06
to	0.05	0.07
fill	0.29	0.06
credit	0.04	0.15
card	0.08	0.11
application	0.06	0.04

What class Naive Bayes classifier would assign to the sentence "I do not like to fill in the application form?"

So, there is one more problem also that I have given. So, in this case again it is a sentiment analysis or a positive or a negative review. So, again to make your life easier I have given two classes and the vocabulary and the values corresponding to each class, ok. So, at the end of this, you need to find out what is the class, the Bayesian classifier would design this particular sentence too is it positive or negative. So, you may want to use either hand calculation or you may want to write a small program and then later extended to multi classes.