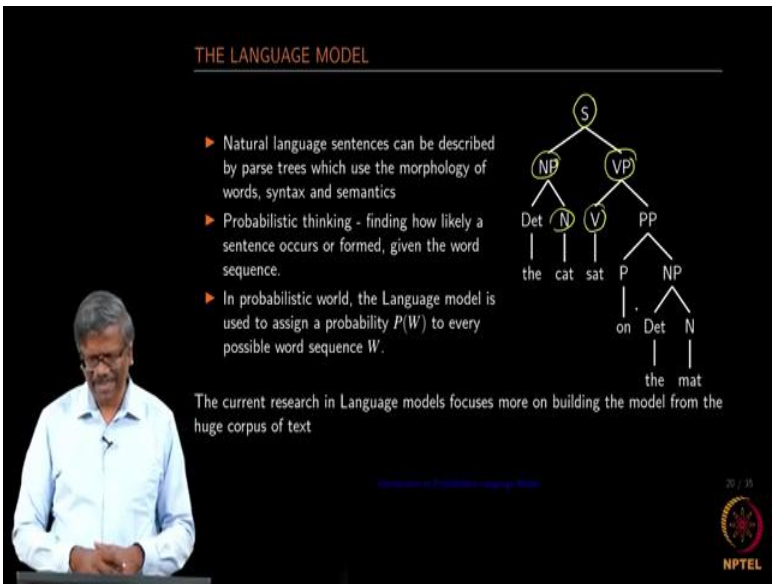
**Applied Natural Language Processing**
**Prof. Ramaseshan Ramachandran**
**Department of Computer Science and Engineering**
**Indian Institute of Technology, Madras**

**Lecture – 23**
**The definition of probabilistic language model**

(Refer Slide Time: 00:15)




So, here we are going to be talking about you know so far what we had seen is how do we really find the probability of word and event, and how do you find the joint probability of two events, how do you find the conditional probability one depends on the other and we have also seen some examples right.

So, now we need to figure out whether what we have learned would be useful in terms of constructing a sentence right. So, what is the normal way we construct this sentence? We can form a parse tree like what you see on the right side of the slide where a sentence is split into a noun phrase, a verb phrase where you will find a noun and then in the verb phrase you will find a verb and it goes on like this.

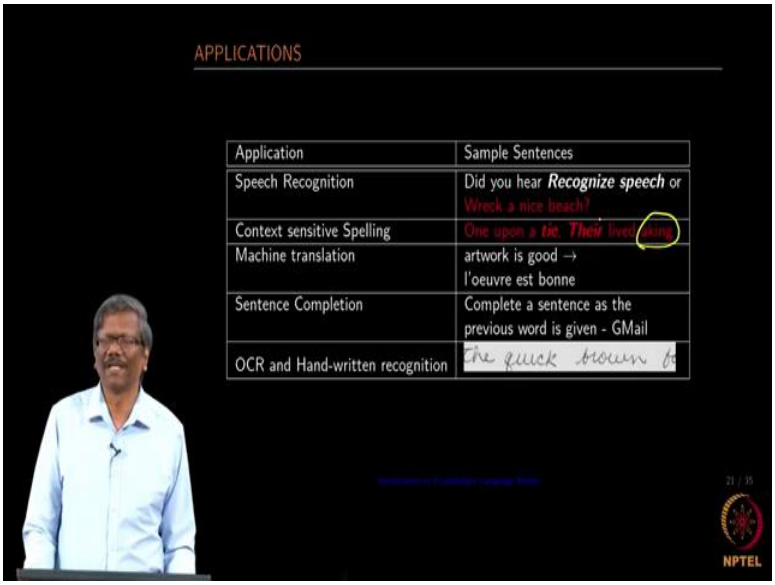
So, if you look at the cat sat on the mat you will see the parse tree as given on the right side ok. So, this is based on how this sentence is syntactically constructed ok. So, if you know how those rules for constructing this sentence you will be able to form a sentence this is one way of learning the language and that is how we have been taught when we moved to the higher classes to learn some languages right. Even though we would be

able to speak the language fluently, but we are taught how to really construct the language in a grammatical fashion using those parts of speech that we will that we have learned.

So, in this particular class we will not be dealing with the linguistics elements of how you really construct the sentence, we will try to see how these sentences can be constructed using the probability and the score that we compute by looking at a very large corpus. So, as we had seen earlier in the in this world of building a language model we use or assign a probability to every word sequence that is available in the document ok. The document contains lots of sentences and every sentence is not restricted by the size it could be from 5 words to 15 words, sometimes even to the twentieth thirty words longer correct.

So, we try to focus in terms of understanding the language sentence construction using the probability measure. Most of the computational linguistics right now focus on building the model from the huge corpus of text.

(Refer Slide Time: 03:21)




So, we are going to be building an application based on what we have learned. So, what are the different applications where we can utilize the knowledge that we have gained so far? One is on the speech recognition; like I am now presenting the lecture and you are hearing this in English, some of the speech recognizers that we have may not be able to decode what we have what I have spoken sometimes you know others set certain things

in a hurry. So, let us take one small example: if I say quickly recognize speech it could be also heard as wreck a nice beach right? You try to speak it a little faster you know recognize speech, it could be very similar to wreck a nice beach.

So, when the speech recognizer could not recognize words clearly the NLP could come into play based on the context it had been trained on. It would be able to say that we are talking about this speech technology all over, so, we are not going to be talking about the wreck on a beach, it is going to be recognizing speech or recognize speech. So, using the probability measure based on the corpus that we have been using to train the NLP system, we would be able to identify or the mission would be able to identify that it is not the right sentence the recognized speech is not the right sentence, but recognize speech is the right sentence because the probability of that particular sentence is a lot higher than this second one that you find ok.

And, then using the context-sensitive spelling: so, when you have written a long paragraph right you tend to make mistakes while typing or some of the auto-correction mechanism correct what you are typing based on a certain character that you have already typed and you keep going because you do not want to stop your flow of thoughts. So, in the end you know there will be some mistakes which the grammar, the spell checker would not have figured out because the spelling is right.

Now, in this case once upon a tie would be corrected as once upon a time and there lived a king would be corrected as t h e r e and then a king would be corrected as two different tokens ok. And, then another application would be on the machine translation ok. So, we want to translate from one language to another you know. We have mentioned this very briefly earlier in order for you to really do this you require a parallel corpus. A parallel corpus contains two corpora: one is for letting us say English, and the second one is for the other language French in this case ok.

So, we want to translate from one language to the other we require some training examples. Let us use those two corpora and then try to find out what would be the right translation for each other ok. So, even if this sentence is given wrongly. For example, I say, artwork good. It is you know in terms when I speak, but when you write it is not the right sentence. So, the system first would use the English model and then correct that, and then probably send it to the translation engine to translate it into the other language

ok. These otherwise could also be true for example, if they have a model created for French and if I made a mistake in the French sentence that model would be able to correct that sentence and then fail into the English translator ok.

And, then the next one would be the sentence completion. We spoke about that at the beginning of the session where you would be able to find out, what would be the next word or the sequence of words that would appear. I am sure most of you are familiar with the application Gmail. So, now, there is a sentence completion available. Unfortunately, it reads your mail and tries to understand the context and then tries to provide you certain words for you to complete which saves you from typing those words.

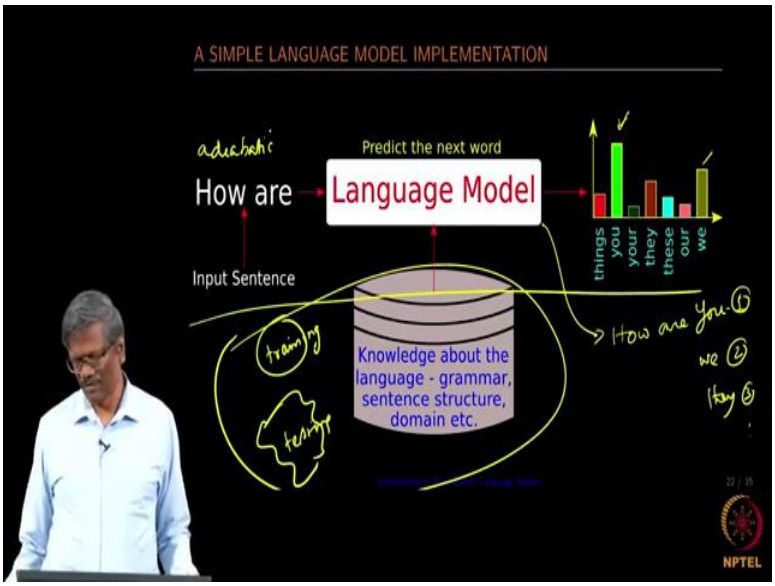
Another example would be OCR and handwritten recognition. There are a lot of handwritten texts that are available and some of the OCR engines even though they are very good with the printed words they are only about 95 to 98 percent accurate or even lower in many cases when the text was handwritten. So, in that case when it outputs a sentence we can run it through the model and then appropriately change the sentence right.

For example, in this case we quick brown we do not know what the next word is right. It is f o, it could be for, it could be the fox, it could be something else right. Also, if you look at this the quicker we can recognize as a human, but how will the machine understand because it is a sequence and if you if it does not find the dot at the top here you can see that there are u's right. So, the q if it recognizes q properly then may it would understand that the next character could be q and then it could give the possible combination of the word where q and u as the starting word starting characters right.

So, it could be quack, it could be quick, it could be anything right that starts with q and u right. It could also give you q u and then it since it does not recognize the rest of the world could say quit, it could also say queen as one of the probable word there. So, in this case since this is quick brown and so on, in many cases it could recognize as a quick brown fox if that particular sentence had been found several times in the corpus it had been trained on ok.

So, these are the applications where you can apply the probabilistic language model and then use it to correct the incoming sentences based on the probability ok.

(Refer Slide Time: 10:56)




So, I am going to be giving a very simple implementation. There could be several other modules that are part of this I am not showcasing them. Let us assume that the model has been built and then the knowledge about the domain has been captured, knowledge about the language has been captured, the grammar knowledge has been captured, the sentence structure has been captured and they have kept in some form for the language model to use.

So, now you are inputting the input since I am just using the same sentence that we had used earlier in our first slide. If the sentence is input as to how are the language model we will look at the at words that are incoming and then look at the corpus and then tries to figure out what could be the possible outcome for this sentence. Here it I am explaining or other I am drawing it very clear that this language model would output let us use the Ngram model of Google.

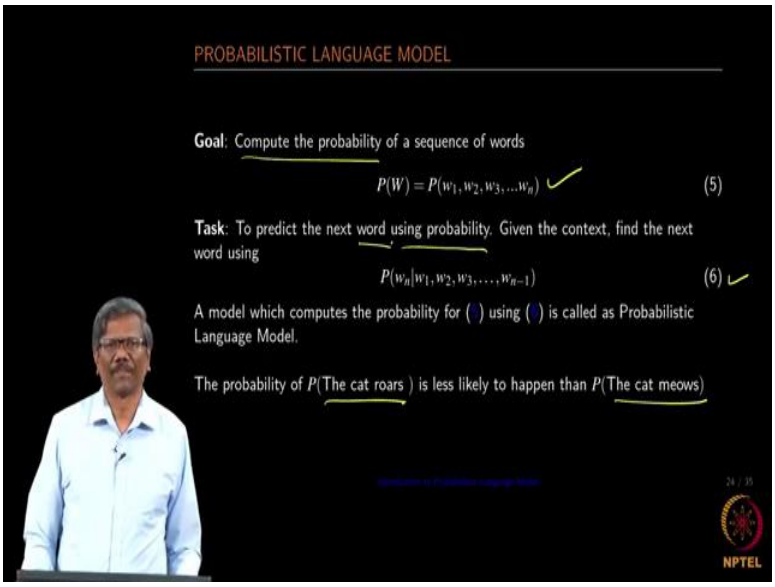
In fact, it outputs you like the first sentence; that means, the probability of you occurring after how are is very high they highly likely that you could be the next word after they gave how and are. So, we can see the value of that being very high there. So, now, we can say that this sentence could be as the first right and then this we as the second one right and then they the third one and so.

So, this model implementation would have various other small modules that we are not really showing. For example, it could use our training phase and then there is testing

involved or to make sure that the model really outputs the right sentence and the words and so on. And, then these are all part of the which I am not really showing. So, I am only showing the input side of that. So, all these things are built and there is a language model available, when you feed in a new sentence it should be able to tell you what should be the next world with the probability score attached to each of those ok.

So, what happened I am going to ask you a question here is what happens if the word that I am inputting is might not part of the corpus that we have? So, this is highly likely that a new word is an input and the corpus had not seen that. For example, if I say adiabatic, the language model is going to fail right. If it had not been trained on this particular domain and if you had not had seen this it is going to fail. So, we will talk about this how to really take care of this situation a little later ok.

(Refer Slide Time: 14:19)




So, now let us get on to the formal definition of how do you define a probabilistic language model. So, so far we have been only seeing bits and pieces of the applications of probability in terms of how do you apply conditional probability to find out whether those two words occur together or not and so on, correct? So, now, we really want to compute the probability of a given sentence. As we had said earlier the sentence is seen as a sequence of words or you can also look at it as Ngrams ok. The goal is to find the probability of this sequence of words. How do you do it? By predicting the next word

using the probabilities; given the context of the history or the history of the previous words I should be able to estimate the next word. Is explained below

$$P(w) = p(w_{1,}w_{,2}\, w_3, \ldots\ldots w_n)$$

$$P(w_1 \,/\, w_1\, w_2\, w_3\, \ldots . w_{n-1})$$

So, starting with this starting word; so, let us say that I am going to start with I how many times I occurred as the starting word, is what we need to find out based on the counter that we had done in the corpus and then move on to the next word. So, after the starting symbol and the word what could be by what would be the next word? So, in this way we can keep building our sentences together or if the words are given we can find out what is the likelihood of this particular sentence to occur ok.

So, how do you do that? So, it is a very complex process in terms of the computation if we look at it. We need to predict every word and then find out based on the previous history of that particular sentence. As I mentioned earlier we are constructing newer and newer sentences innovatively, and then this sentence that we are forming now would not be available as part of the corpus. So, in those cases we are going to fail right.

So, it is also impossible for us to use the entire history in terms of the context in order to find the next word. So, let us move on to the definition here. A model that computes the probability for this equation using 6 is called the probabilistic model. So, as a simple example would be the probability of The cat roars is less likely to happen than The cat meow right. So, a language model is one that tries to find or compute the probability of a sequence of words using the word probability.