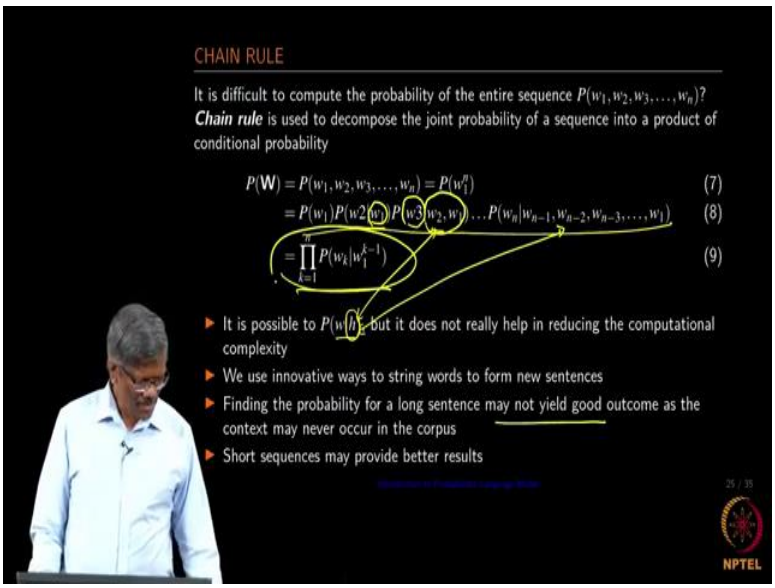
Applied Natural Language Processing
Prof. Ramaseshan Ramachandran
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Lecture – 24
Chain rule and Markov assumption

(Refer Slide Time: 00:15)



So, how do we compute this probability? So, if you look at the computation of the probability of the whole sequence, it is nothing, but a joint probability right. So, the probability of the first word with the entire one, probability of the second word with the rest of the word, and so on and so forth. If you look at it is a very long sequence of probability and you have to multiply each and every one of those to find the probability of the given sentence ok. What the chain rule does is it decomposes the joint probability into a product of conditional probability ok.

Actually, what I will do is, I will go through the theory part of that and then at the end of this I will take an example. And, then see why is it very hard to compute the joint probability and then why are we really using the chain rule first to create the product of conditional probability for the same sequence, and then later how it could be approximated so that the computation complexity can be reduced and so on ok.

So, right now let us stick on to the theory part of that and then let us define those equations and then later we will use those equations to see how those things can be

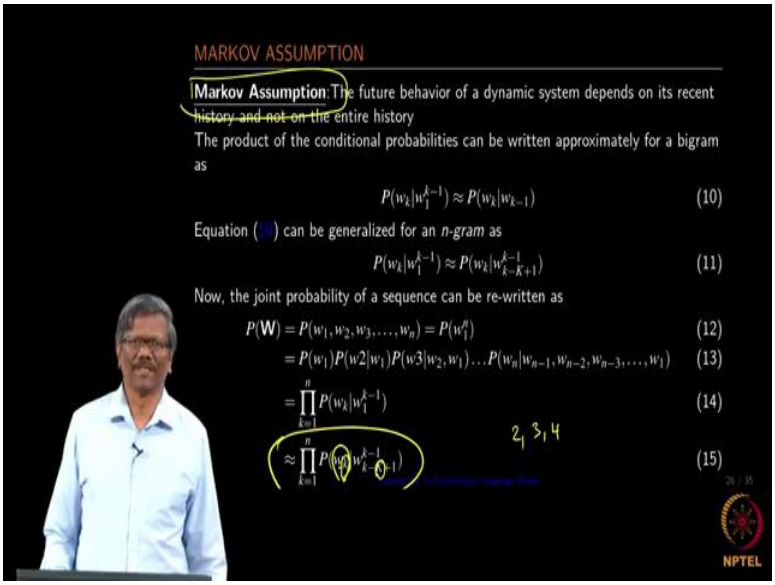
reduced in terms of complexity, and then how we can compute the probability of each sentence and so on. It is possible if now if you look at this chain rule, we can compute the probability using the conditional probabilities, so that what I have done in this case here right.

So, I have taken the first word and then the next one is computed based on the over w 1 and then if you look at the probability of finding this is conditioned by these two words and then and so on. So, it is a product of the conditional probabilities which can be written in a very simple form concise form in the mathematical notation.

So, we can compute I am just writing it as a history. History is nothing, but these two words right in a simplified form. It is not really reducing our computational complexity right. So, we have to use a different mechanism in order to reduce our complexity. Also we use a lot of innovative ways as I mentioned earlier to form new sentences. So, it is highly unlikely that a new sentence that we are forming differently would be available as part of that only parts of those sentences would be available as part of the carpel.

So, how do you deal with that situation when new sentences are formed and we are not able to really find the conditional probability for such a long sentence in the existing corpus? So, we need to split that into smaller chunks of words rather than a long chunk that we have seen earlier in the chain rule decomposition.

(Refer Slide Time: 03:58)



We can use the Markov assumption. Here what it does is, we can approximately change the equation into a different form, where we define that the future behavior of the dynamic system depends only on the recent history not on the entire history. We require only the recent set of words to compute the probabilities. So, we are saying that it is possible to approximately rewrite the statement that we had or the equation that we had seen earlier into an approximate form, where we would only use the recent history and not the entire history.

So, we are rewriting this in terms of n-grams which are smaller in number could be two bigrams, could be trigram or five-grams we are not using n-grams of size greater than 10 or 15 and so and so forth or considering the entire sentence. So, when we do that it is going to be only an approximate, not going to be an exact one like what we consider or when we consider the entire sentence. So, there could be some error in the computation which could be minimal, smaller and so on.

So, there is always a possibility that the sentence that we are forming using the Markov assumption need not be right every time ok. So, that is the covey that I want to place at this point in time, but it is a good approximation for most cases. Since we are considering only the nearest neighbors in terms of three words or four words as the history, it is possible that this gram this assumption may work 95 percent of the time ok; the 5 percent of the time we still have to deal with. Maybe when the computing power increases beyond what we have it may be possible for us to deal with that higher number in the later cases ok.

So, in this case what we do is we rewrite the entire sequence as follows. So, if you look at this the k is your n-gram could be 2, 3, 4 and so on. So, if you look at history it is going to be only between we are going to be considering the history for this word k using a bigram, trigram, or four-gram or a five-gram ok. So, this is the assumption that we are making and we are trying to reduce the complexity in terms of computation using this approach ok.