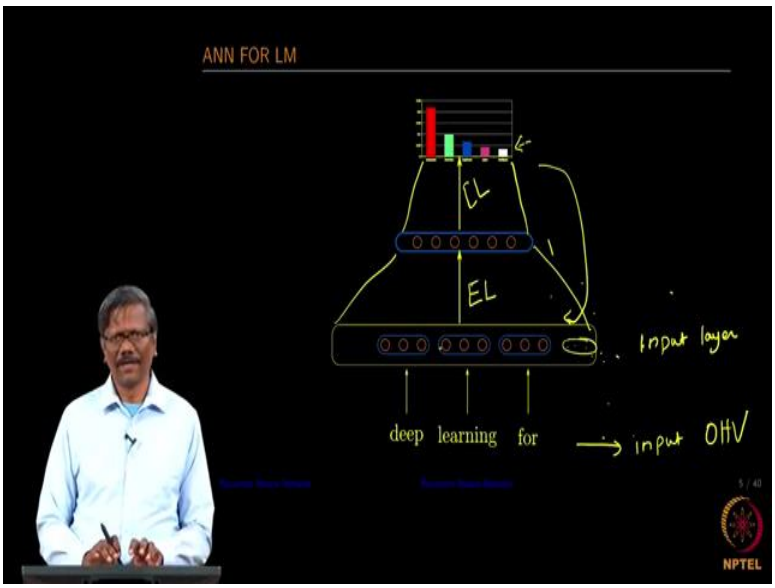
Applied Natural Language Processing
Prof. Ramaseshan Ramachandran
Department of Computer Science and Engineering
Chennai Mathematical Institute

Lecture - 52
ANN as an LM and its limitations

(Refer Slide Time: 00:15)



Alright. So, now, in the next phase of improvement, we want to find out if we can really use the neural networks all in the same problem. Again we know well that its possible to do that job right. So, we have the input layer here and then this is your input, in one hat vector form ok; and then we have the embedding layer and this is your context layer remember. We have used the terminologies earlier and then this is your output through the Softmax correct.

So, what we do here is, we have a fixed-sized input window where we can input three words. Let us say that we have a corpus, where we are going to be inputting all the words in this order. So, we are going to first start with three words and then say, what should be the next word so, we are going to train that. So, you do the backpropagation training and then we continue to update this until the network stabilizes right.

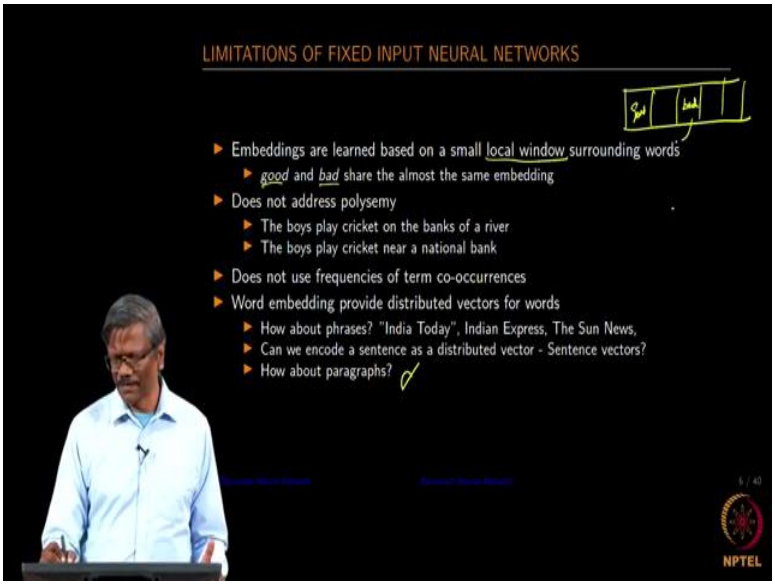
So, in this fashion you can input all the words in this network, let the network get trained on the prediction; and once it is done we can use the model to really predict the next

word here right. So, we not only learned the word embeddings but also have the capability to predict the next word through this.

So this we have not seen earlier, we only saw how this central word can be predicted in the case of a CBOW model or in the case of the skip-gram, how the context words can be found out. So, in this case if I want to really increase the size of the input; that means, from 3 to 4 words.

So, I have to start changing this and then my weights here that I have will have to be changed correspondingly we will also change this here. So, it is not easy to really reuse whatever we have done earlier for a longer sequence of words. So, we have to really go for another model in this fashion you understand this. So, a traditional neural network can be used to train, but unfortunately there is a restriction that, the input layer sizes fixed and start ok. And this does not really bother about the sequence of words that is available in the time series that is another ok.

(Refer Slide Time: 03:33)



So, what are the limitations as I mentioned earlier? We learned the embeddings through these models using a local window right its a very small window that we move it. It is possible that the word good and bad could appear together ok, with an of two words or three words and so on. So, what is going to happen when you have that, suppose if you have the word window you have good here and then bad here good or bad take it and

then start evaluating it later when you train the network. So, we are going to be having this as the context for us correct; so, that means the embedding is not learned properly.

So, there is a possibility it is not the problem with the network it is the problem with the way we have constructed the sentence we probably should construct the sentence very carefully. So, that these kinds of opposite words do not occur together is it possible? I think in a natural language it is not possible. So, we have to give it to that the network would not be able to really do a good job if these words occur very close to each other right.

This is one of the very important aspects that we wanted to address right from the beginning of polysemy. We were able to capture the similarity in some form right we are not able to address the polysemy again even through the neural network that we have studied earlier. So, to again refresh your memory of what a polysemy is. The boys play cricket on the banks of a river right so, look at this the boys play cricket near a national bank. So, the same word has a different meaning right. So, how do we really address this? So, how will I interpret that, this is not like a place where I can do the financial transaction or this is not a place where there is a river nearby right.

So, we need to be able to distinguish those words when we start processing the sentence correctly. So, how do we tackle this polysemy problem? Is it really possible to solve this in the case of natural language processing? There are few application or the mechanism that has cropped up in this. So, one is to use another language where there is a distinction made right.

Say for example, I can say from the point of view of Tamil a bank near the river is called a [FL] right. And then the bank where we do the financial transaction is called [FL]. So, there is a very clear distinction made in the language in the Tamil language. So, we can actually use that aspect and then see if we can distinguish that. So, for example, if I translate that into the Tamil language if this one this particular one sentence and then translate this into Tamil again.

So, you have two different meanings. So, then you can say that from the interpretation, you can say that this particular word refers to really the banks of a river and then this relates to the place where you do the financial transaction. So, it is possible and for us to really understand that again a small local window will not suffice. So, we require a

longer window where we should be able to understand the entire meaning of the sentence.

One thing that you must have noticed so far you know in the case of the neural network rare we used CBOW model as well as skip-gram, we completely ignored, the frequency of the words in the vocabulary right. So, that is another aspect that we want to consider, we are able to fit the embeddings of the words in a distributed fashion right. So, are you able to also use the same thing for phrases the phrases such as you know India today is a magazine oh?

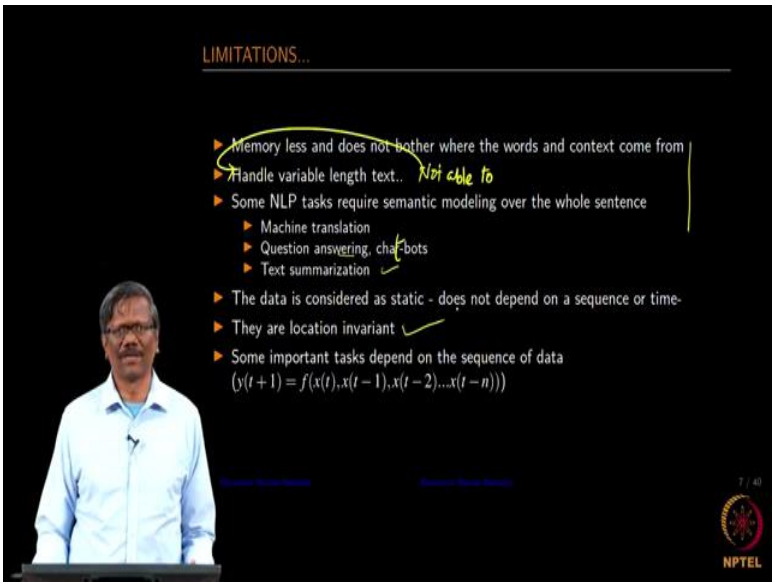
But these two are two different worlds, but they are joint here right. So, this phrase you should be able to understand. So, how can we do that? One way to do it is to take all those words where you see India today appearing to make it as one word in the entire corpus. And then use it as one word and then train it; so, that is one way of doing it right the same fashion here. So, can we learn these phrases?

So, these are all important things as part of the natural language understanding right. So, we should be able to really understand what are phrases and how they occur and what kind of words occur together and then, completely have a different meaning for those correct. Like we have done in the case of the word so, can we really create an embedding for sentence.

Suppose if I give a long sentence, can I create a word vector for that sentence? So, why is that useful? So, we will come to that when we talk about the applications ok. So, how about a paragraph? Suppose if we want to really identify paragraphs that talk about the same thing in a similar context; we should be able to bring them together in one go right. For example, if I want to find the papers that talk about a backpropagation through time the b PTT right.

So, I want to be able to get all those papers aligned along in some fashion so that I could compare who is really giving me the right description, which one gives me a better meaning we normally do that right. So, when we have two different books, we try to see what is described in one book and then try to find out how the other book also describes the same thing right. So, in the same fashion, we should be able to do something with the paragraph as well can we really get to that level in the language understanding through the model that we are talking about alright.

(Refer Slide Time: 10:51)



So, the limitation that we have gotten into so far with their traditional model is they are truly memoryless; they do not really bother where the context is coming from alright. So, for example, when you train the network the weights are adjusted right. So, for a given context or a given central word you get the context word script.

$$y\,(t+1) = f(\,x(t-1), x(t_2-1) \ldots\ldots\ldots (t_n-1))$$

So, when you do the weight updation every time we lose what was there before, we do not really care about it we just update so that the network gets into the equilibrium state very quickly. We do not also really bother where these words came from. For example, if I train a word in the skip-gram model the word could be at the start of the sentence in the middle of the sentence or wherever it is and then the context could be appearing in. So, many different locations of the same corpus we do not really bother about that the location and so on. And we saw that we are not able to handle variable-length text right.

So, if we want to do that we need to really change the architecture of the network that we have. We require really some semantic modeling over the whole sentence. The two words do not really give us you know bigrams or trigrams do not really give us it only gives you the context in which the next word or the central word appears beyond that it does not give you any meaning related to that. So, our aim was also only to find the word embedding not the semantic part ok.

So, if we have to really get to the semantic level, we really have to look at the whole sentence. So, for example, in the mid-case of the machine translation I need the whole sentence to be able to translate from one language to the other; and then question answering. I need to be able to understand what somebody is asking you to know if you are asking me a question I need to understand that question so that I should be able to understand the background of your question understand what you are asking and then try to explain the right answer correct. And then the chatbots; so, you should be able to seamlessly communicate with the chatbot you know without knowing that the machine is answering your queries text summarization.

So, we mentioned earlier right, we have a long text and then we want to be able to provide the abstract of the long text in terms of ten sentences or five sentences you would normally find this as an abstract in many technical papers right; even many newspapers give you one paragraph or the summary right. So, can we do the text summarization using the same or using the neural networks? Whatever we are talking about could not be achieved using the traditional model of neural networks ok.

So, those are the limitations of the. And then we consider this data static we do not really care about the sequence as I mentioned earlier. We do not care about whether it is a sequence of words or it is a time series one right. And then the location of the word is invariant the location is not really a problem when you want to train, but in certain cases locations are important we should be able to process this sequence which is like a time series.

So, can we get to this level of solving all these problems that we are talking about, in the model that we are going to be discussing today?