

Applied Natural Language Processing
Prof. Ramaseshan Ramachandran
Visiting Professor at Chennai Mathematical Institute
Indian Institute Technology, Madras

Lecture - 16
Collocations, Dense Word Vectors

(Refer Slide Time: 00:15)

COLLOCATIONS

Collocations is a juxtaposition of two or more words that more often occur together than by chance.

- ▶ Poverty is a **major problem** for many countries
- ▶ Ram has a **powerful computer**
- ▶ I had a **brief chat** with Raj
- ▶ I could not see anything in the room, it was **pitch dark** inside
- ▶ The crime was committed in **broad daylight** - We don't use wide, large, big daylight
- ▶ I wish I had a **strong tea** - we don't use powerful, tough
- ▶ The **heavy rain** prevented us from playing outside - We don't use strong rain
- ▶ Someone **knocked on the front door**

14 / 25
NPTEL

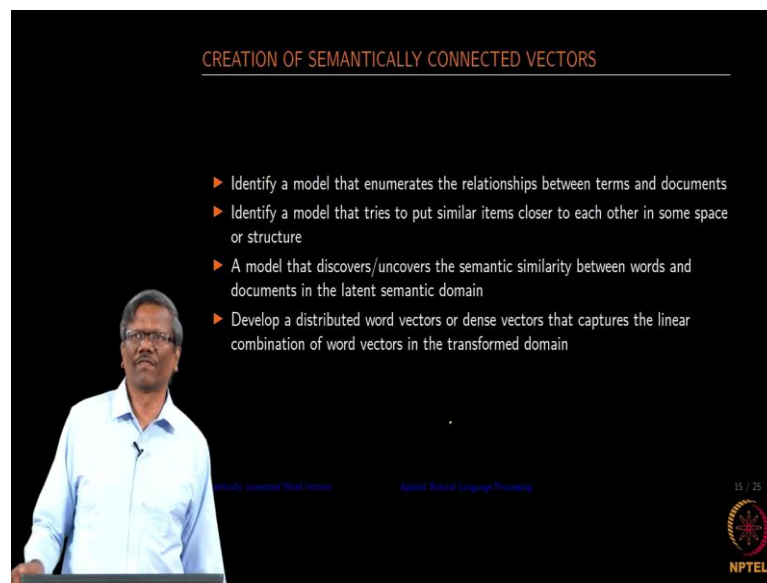
So, we always use the word major problem, poverty is a major problem for many countries right. Ram has a powerful computer; nobody will say Ram has a strong computer right. these two words always occur in that form. And then I had a brief chat with Raj ok. I could not see anything in the room it was pitch dark inside right. And then the crime was committed in broad daylight; we do not use wide, broad, you know the synonyms are wide, large, big and so on. we do not use wide daylight, large daylight and so on.

So, these two words occur in the corpus not by chance, but because of the usage ok. these are the collocations ok. sometimes they do not occur together, sometimes they occur away from each other. For example, someone knocked on the front door ok. now, this knocked and door, I could have occurred in several ways ok, somebody knocked on the door right. it could happen after the knocked and door you know there are three in this case, three words in between; sometimes it could be just somebody knocked on the

door or knocked on the door. the distance between the word knocked and the door could be varying.

So, we should be able to figure out how these two things could be picked up, and how these two things occur only together in this fashion. For example, nobody will say somebody hit on the front door type right. we should be able to capture these types of words as well in the corpus ok.

(Refer Slide Time: 02:29)



CREATION OF SEMANTICALLY CONNECTED VECTORS

- ▶ Identify a model that enumerates the relationships between terms and documents
- ▶ Identify a model that tries to put similar items closer to each other in some space or structure
- ▶ A model that discovers/uncovered the semantic similarity between words and documents in the latent semantic domain
- ▶ Develop a distributed word vectors or dense vectors that captures the linear combination of word vectors in the transformed domain

15 / 25

NPTEL

So, now we found out the method of capturing a word that is co-occurring together using bigram, trigrams or engrams. What are we going to do with that, so why do we need those things? You know are they really going to give me some additional information about the word? Yes, it is going to give us additional information about the word and its meaning.

Is it going to be in the same domain of operation where I can find the similarities of the word or synonyms of those words and so on? Possible, it is also possible that it is not going to be in the same domain where the term, document vector, or the matrix up here, it could be in a transformed domain. we need to identify a model that really finds the relationship between terms and documents not necessarily in the same domain that we have created, it could be in a transform domain.

As you are aware in the earlier cases and in the domains that I have shown, all the operations happen in the same domain ok, so there is no transformed domain and so on. Let us talk about that or see how those domains can be transformed and how we can actually retrieve information from the transformed domain a little later. We need to identify a model that tries to put similar items closer to each other in some space or structure. Remember we spoke about the dimensionality problem right.

So, if we have assumed that we have represented or we have named all the colors not just red, green and blue, but a million colors we have names for all of those, and we have to make them available as part of the vocabulary. And if you are considering them as an independent entity, then we could not have a million access for each of those colors. if you want to reduce the number of access, what I was talking about earlier was can it be created in an ease vector or an ease a structure. For example, if color is considered as one element where we can say red is a color or color is your facet there ok. Green is a color, yellow is a color that means that color facet could be put in one single access.

Again if you consider the composition aspects of a classroom that contains blackboard, chalk piece, students, tables, chairs, maybe your projector or some TVs and so on right. we should be able to write or create access related to those compositions. For example, the kitchen could be one access; the classroom could be one access and so on, so there why you can reduce the dimensionality in some form.

So, is there any other way of doing that in a different domain or in a transformed domain? Yes, it is possible to really create a lower-dimensional structure in a different domain. We need to find out whether those kinds of models exist or how we can really exploit those patterns in the corpus so that we are able to reduce the dimension as well as find the meaning of the word.

So, and then we also want to find out whether the word vector can be distributed in nature or we can create a vector that is dense in nature. Remember in the one-hot vector a word is represented by the index or in that particular column. For example, if all the elements of \vec{v} except one element which is 1 that means that particular element in that column represents that word. it is an index of the word in the entire vocabulary, the rest of them are 0. we have no use of those.

So, can we create and it also one-half vector as you remember it does not really give you any similarity of the words? It does not really talk about other relationships with similar words. we want to find out whether it is possible to create a vector that is dense in nature, smaller in size, but captures the similarity of the words or synonyms or the context in which the words appear and so on. this is going to be a very interesting exercise that we are going to see in a short while ok.

So, as I mentioned these kinds of operations need not happen in the same domain as a term-document matrix or in the coincident matrix or binary incident matrix space. it can happen in a different space. Again it is a real space it is not very different, but it will be in a transformed domain. And each and every element will not represent exactly what we saw in the previous or in the earlier domain space ok.

(Refer Slide Time: 08:03)

METHODS TO CREATE DENSE VECTORS

- ▶ Latent Semantic Analysis or Latent Semantic Indexing
- ▶ Neural networks using skip grams and **BOW** (Center Word)
- ▶ CBOW - uses surrounding words to predict the center of words
- ▶ Skip grams use center of words to predict the surrounding words
- ▶ Brown clustering - statistical algorithms for assigning words to classes based on the frequency of their co-occurrence with other words

16 / 25
NPTEL

So, as I mentioned in the previous slide that our aim is to reduce the dimension, and at the same time create a dense vector. What are the different ways that we can do it? one of the ways that we are going to be looking at this is called a latent semantic analysis or latent semantic indexing ok. And then we going to be in the future to see some neural networks which would use the skip grams and continuous bag of words, I will mention a few lines about that in this slide. We going to be using brown clustering, we will talk about that later which is a statistical algorithm for assigning words to classes based on

the frequency of their co-occurrences of other words ok. there these are the different mechanism by which you can create dense vectors.

In the second point where I mentioned about skip-gram and CBOW, CBOW is nothing but a continuous bag of words. Remember what a bag of the word is, supposing if you have a long sentence, and then you a long sentence written in a strip of paper, and then you cut every word and drop it in a bag right. And then inside the bag, you have the words that belong to the entire sentence, but they are not in any order correct. if you pick one, it will be any random word that you actually cut, and then put it inside the back. In the case of a continuous bag of words it is not anything at random that you pick up from the bag, it is actually three words or four words co-occur together.

So, let us take one small example. In this case, I am going to use the same example that I used to define CBOW. let us say that we have uses surrounding words to predict the center of the words. It is nothing but fill in the blanks. For example, if I do not have this and if this is the sentence that was used in a book and in the question paper, you are asked to fill in the blanks. If this sentence is given, uses surrounding words to dash the center of the word; that means, here you will go on and write predict and if you are familiar with this particular sentence.

In this case, the context is used surrounding words to the center of words that is the context for the center word predict. this is your center word ok. In the case of skip-gram, in the case of skip-gram, you think of exactly the opposite of CBOW. This central word is given. you need to find out you need to find the surrounding words ok. the input for the neural network or in any other natural language processing application would be predicted.

And the idea of that particular model used to really find out what are the words that surround the predict should be able to give you with some probability value that skips grams use the center of words, the surrounding words should have some probability which is higher than any other random word in the vocabulary that you have. this is exactly the opposite of the CBOW. we will talk about those when we go into the details of embedding using neural networks and so on ok.

(Refer Slide Time: 12:25)

The slide is titled "WHY DENSE VECTORS?". It features a speaker on the left and a list of bullet points on the right. The bullet points are:

- ▶ Sparse vectors are too long and not very convenient as features machine learning
- ▶ Abstracts more than just frequency counts
- ▶ It captures neighborhood words that are connected by synonyms
 - ▶ Consider these two documents (1) Automobile association (2) car driver
 - ▶ Connects the neighbor of Automobile and the neighbor of car
 - ▶ "Automobile association" with "car driver" - driver and association could be connected using the similar words *Automobile and car*

There are handwritten annotations in yellow on the slide: a bracket under "Automobile association" and "car driver", and two overlapping circles at the bottom right. The NPTEL logo is in the bottom right corner.

So, why dense vectors as I mentioned earlier sparse vectors you know contain only one element in the entire 1 million words a word vocabulary right. there is only one element in the entire, 1 million elements of a one art vector which is not very convenient with respect to providing it as an input to any machine learning application. We also want to have more information related to the word and its similarities, and how the context words are attached to that particular center word and so on.

So, if the dense vector should be able to provide the entire information related to the similarity of the context of the similarity of the words and things like that. we also want to find out additional information from the dense vector to give you one example. Let us assume two documents automobile association and car driver right. These two are two different documents. And we want to associate the driver with the association.

How is it possible? So you have an automobile association; you have a car driver, and I want to find out the relationship between the driver and the automobiles. Assuming that you know we know that automobile and car are common words then it is possible for us to associate the association and the car and the driver automatically. these kinds of operations are possible when you create a dense matrix. This just a simple example I thought I would provide at this point in time.