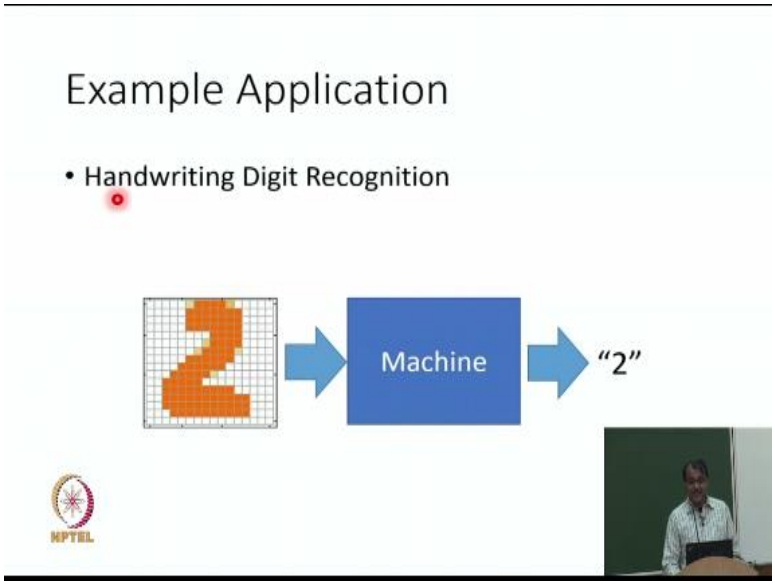
Artificial Intelligence
Prof. Mausam
Department of Computer Science and Engineering
Indian Institute of Technology-Delhi

Lecture - 85
Deep Learning: Example of Handwritten Digit Recognition

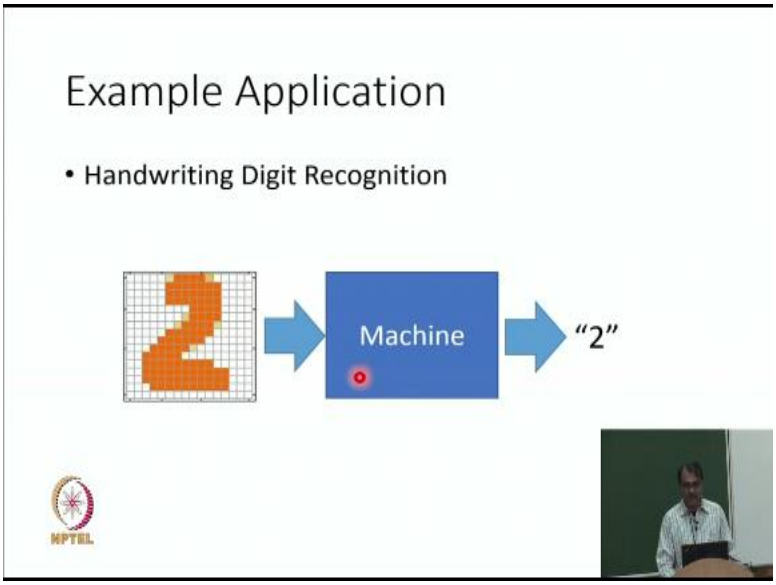
Now that we have understood so much, let us start with our first or our learning example. Our learning example would be of handwriting digit recognition.

(Refer Slide Time: 00:27)



The question is I am giving you a set of pixels and you as in the machine and you have to tell me, what does it say. And it will only say 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 one of the 10 digits. And here you want the output to be 2. But your input is an image and your output is a number between 0 to 9. Everybody with me on the question on the application? Now let us first think about what is my input.

(Refer Slide Time: 00:56)



Let us say I am giving you a 16 plus 16 bit vector, just to simplify things. So orange here means black. So either my pixel is white or it is black and my output right so we have 256 inputs. Each input is 01. So we can think of it as a vector of inputs, okay. So my input is x 1 and x 2 up to x 256. And each x i corresponds to a particular pixel. And right if there is ink 1 one if there is no ink zero and my output and now this is going to be interesting, we want a single number out.

But that might be hard for a neural network. So what we will do is we will say give me 10 outputs. Independently tell me not independently, but tell me whether it is a number or digit 1 or not. Whether it is a digit 2 or not, whether it is a digit 9 or not. And because the neural network will make somewhat like an approximate prediction, it will not be sure that it is number 2 or it is digit 3 or is digit 5.

It will give me some probability distribution over all the possible outputs. So basically think of y 1 to y 10 as the confidence that the input is the digit whatever i. Everybody with me on the modeling so far. So the input was an image 0, 1 image 256 dimensional because it is a 16 cross 16 image. The output we want a single digital out. But the neural network will find it hard. So we say give me 10 digits out.

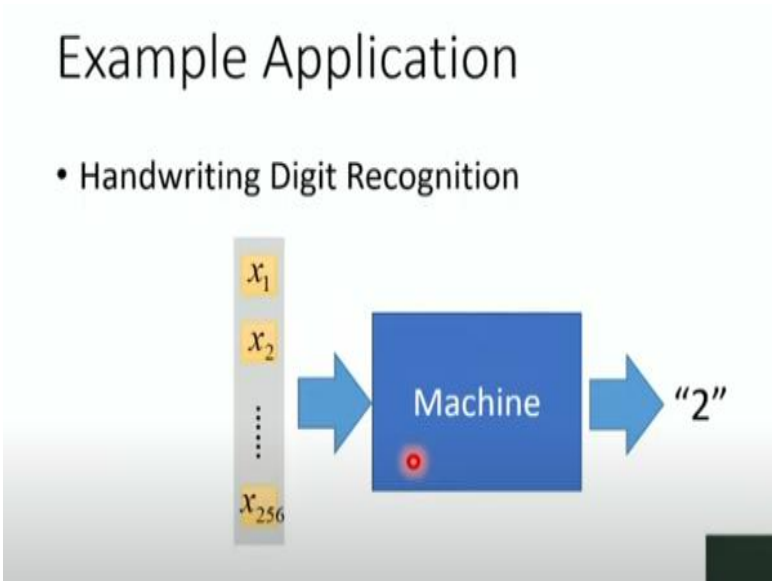
Tell me whether it is digit 0 or digit 1 or digit 2 or not. But it cannot give me a 0 1 answer. It will give me a number between 0 and 1. It will give me a probabilistic answer. So we say okay, fine. Give me just the probability distribution. I am okay

with that. I will manage with that. Okay, so for example, one probability distribution could be the output would be okay, I think it is 1 with probability 0.1.

I think it is 0 with probability 0.2. And I think it is 2 with probability 0.7. And we can deal with this, right? Because we know that it is the most likely prediction is 2. So we will say this is 2. This is what my neural network is saying okay. So this would be my final answer to the problem okay. And so now the most interesting part for us is how do we build this middle blue box?

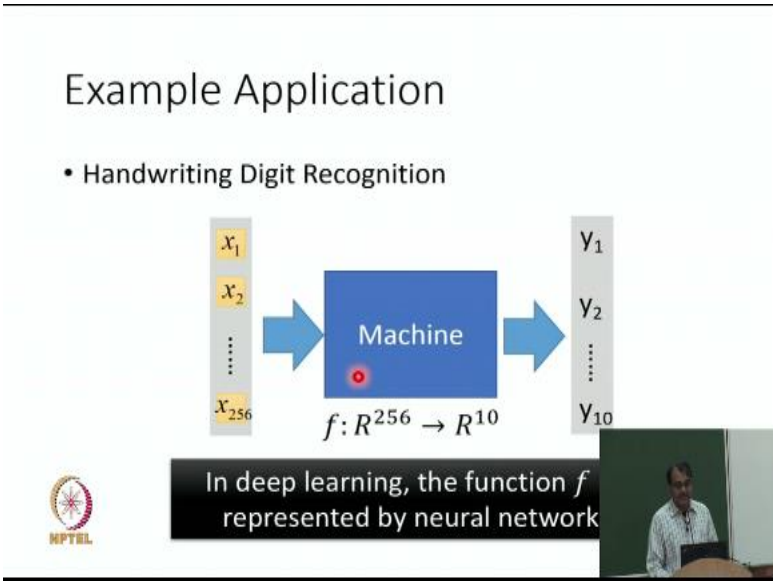
How do we build the machine box which can take this input and give me the output. And of course using neural networks because that is what we have been working on.

(Refer Slide Time: 04:04)



And this is a famous problem of handwriting digit recognition.

(Refer Slide Time: 04:10)



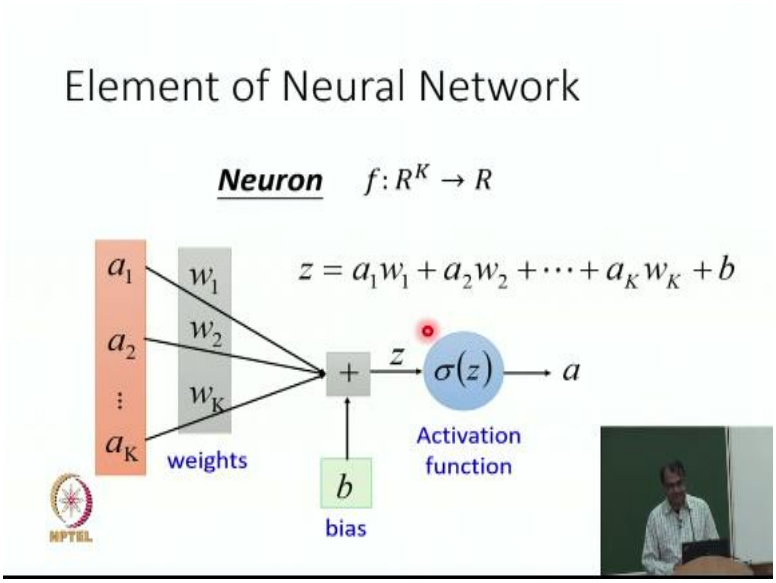
Now if you think about it, can we cast it as a function? That maybe the machine wants to represent a function. If it has to represent a function it would represent a function from what input to what input. What is the input dimensionality? 256. What is the output dimensionality? 10. So we want to learn a function that goes from a 256 dimensional input to a 10 dimensional input okay.

And technically the 256 dimensional input is a Boolean input and the output is a probability distribution. So not all outputs are valid like 11111 is not allowed, but we will add that as constraints to the network and not worry about it. So for now, so neural networks typically work in the real space, right even in complex space sometimes. They can be complex neural networks.

Not about complexity of the network, but actually each number is a complex number. But again, we will not talk about it. Most neural networks take input in the real space and take output in the real space. So we always think about how do we convert our input to a real space and how do we convert the output to the actual space that is needed, okay.

And what is going to happen is that this function f is the function that will be represented by a neural network, okay.

(Refer Slide Time: 05:38)

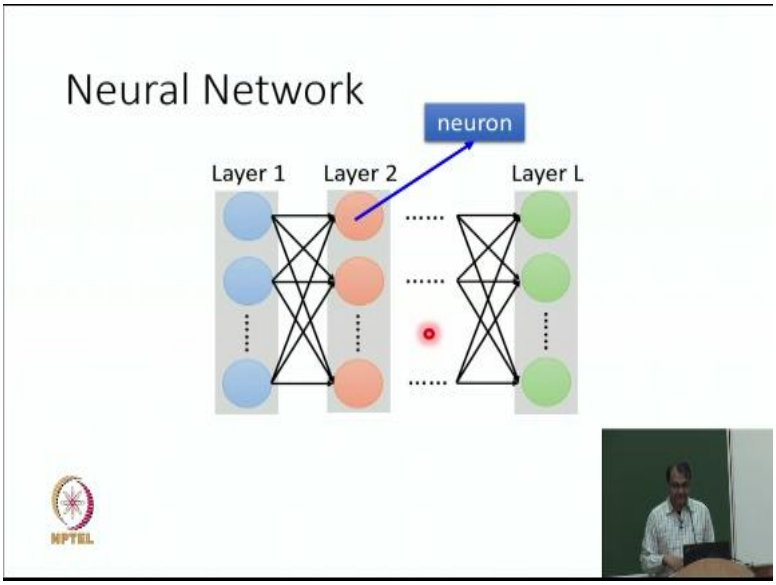


And how would it be represented? Well, the same idea, we will put all the inputs into a neuron. The neuron will have weights. The weight it will have some non-linearity and then it will send signal out and then there will be another neuron and so on so forth. It will be exactly that. So these are my inputs. Let us say they go through a weight, they go through a sum and we will also add a bias term.

This bias term, think of it as a mu i term. But now we will call it the bias term. So this is my final linear sum, it is summation w i a i + b. And then that will give me the output z. And from the z, I will take it through any kind of non-linearity. Sigmoid ReLU, tanh, whatever. And then that will give me the output of this neuron.

And this output of the neuron may go through go through another input of the next neuron or next set of neurons. And these w i's are the weights. And this B is also called the bias term. And this is the activation function. So we all understand these concepts.

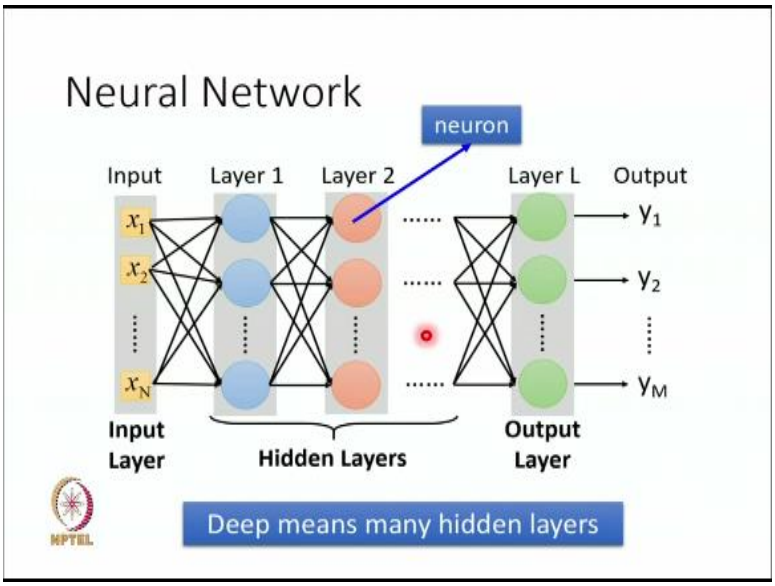
(Refer Slide Time: 06:40)



And one kind of a network could be a network where these neurons are arranged in layers. And pretty much all inputs go to each neuron and all the outputs of the ith layer go to the input of the i plus 1th neuron and that is called a fully connected network, fully connected feed forward network. Feed forward means that I take the input and I feed it forward, and some output comes out. These are just terms that people have coined.

But basically, it just means a multilayer perceptron, where every output from the previous layer goes to every input of the next layer.

(Refer Slide Time: 07:18)



But there will always be an input layer, and there will always be an output layer. So do not forget these. These are very important, right. And in the middle all these layers,

there is the input layer, output layer. But in the middle, all the layers are called the hidden layers. And the reason they are called hidden layers is because not because they are hidden from the human, but because human has no understanding of what is going on internally, right.

Human knows the input layer, you can observe the image. Human knows the output layer, you can figure out what the neural network said. But in the middle, what does each neuron do we do not know. And in fact, if you had to figure out what is a particular image, which digit is it, what intuitions will you use? What intuitions will you use? You will identify the basic, the basic shapes structures. What is the basic shape structure?

Straight horizontal line, straight vertical line, straight line like this, straight line like this, a curved line in this direction, a curved line in this direction, this direction, this direction etc., etc. And then you will say how are these arranged? Okay I have a curved line and a curved line and a curved line and another curved line. I have four curved lines and they are arranged in sort of this kind of a fashion.

Then we will say okay, this is zero. Or I have no curved lines. I only have a vertical line and an angle line like this, and it is arranged so that the angle line is at the top and the right. So then you will say it is a 1 or whatnot. So you will first define some basic shapes, then you will start to combine those basic shapes, then you will combine those basic shapes. And finally will say okay these basic shapes are in this configuration.

So I am going to output 1 or 0 or whatever. So internally what am I saying? I am saying that there is the first step of identifying this smallest shapes and then identifying the slightly bigger shapes and the bigger shapes and the bigger shapes. And that can be arranged in a sort of a sequential fashion. And you can like and you can imagine that the neural network layer 1 is sort of doing the very basic shape and layer 2 is doing slightly stronger shapes and layer l - 1 is doing sort of the full picture, right.

Again, what is the neural network doing we do not know. But intuitively, this is what it might be doing. Right? Yes. **“Professor - student conversation starts”** Sir once

we know the weight of these link then we can give the function explicitly on what a neuron will be internally. **“Professor - student conversation ends”.** So the question is that Jai asked if I know all the weights, can I go inside and figure out what is it that each neuron is doing?

And technically, the answer is yes. We can figure out what function that neuron is representing. But that function in the middle will be a function of all the neurons in the previous layer, which will be a function of all the neurons in the previous layer. Because of all this complication, it will be very hard for us to understand the middle neuron independently. Technically, you can compute that function, right.

And what is a deep network, deep learning, nothing. It is just too many layers. That is it. That is the definition of deep. If I give you one layer, I will not call it anything interesting. If I give you two layers, it is universally approximated but it is not deep network. If I give you 100 layers, absolutely it is a deep network. It is said that the number of zeros in your salary is a function of the number of layers of neural networks that you have been able to train.

And this is it is also a joke, but there is half-truth in this. So if you are able to train a 100 layer neural network, then a million dollars have not gone anywhere. I am talking dollars by the way. If you can say that I trained a 100 layer neural network from scratch without using other codes and just taking it in and downloading and solving. No. You put in the effort to train it. A million dollars have not gone anywhere.

You will easily get a million dollar salary today. Today, maybe not 5 years from now, but at least today, okay.