

Deep Learning
Prof. Mitesh M. Khapra
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module – 4.6
Lecture - 04
Back Propagation: Computing Gradients W. R. T. Hidden Units

Now, we will go to the Gradient with respect to the Hidden Units.



(Refer Slide Time: 00:16)

Quantities of interest (roadmap for the remaining part):

- Gradient w.r.t. output units
- **Gradient w.r.t. hidden units**
- Gradient w.r.t. weights and biases

$$\underbrace{\frac{\partial \mathcal{L}(\theta)}{\partial W_{111}}}_{\text{Talk to the weight directly}} = \underbrace{\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial a_3}}_{\text{Talk to the output layer}} \underbrace{\frac{\partial a_3}{\partial h_2} \frac{\partial h_2}{\partial a_2}}_{\text{Talk to the previous hidden layer}} \underbrace{\frac{\partial a_2}{\partial h_1} \frac{\partial h_1}{\partial a_1}}_{\text{Talk to the previous hidden layer}} \underbrace{\frac{\partial a_1}{\partial W_{111}}}_{\text{and now talk to the weights}}$$

- Our focus is on *Cross entropy loss* and *Softmax* output.

Mitesh M. Khapra CS7015 (Deep Learning) Lecture 4

So, this portion. So, you already see there is a repetition, here and I do not need to treat each hidden unit separately I can just have a formula for the hidden unit and then I could compute it for all the hidden units ok. So, that is what our aim is. So, let us do some simple stuff first and then you will come back to it.

(Refer Slide Time: 00:33)

$$\frac{dz}{dx} = \sum_{i=1}^2 \frac{dz}{dy_i} \frac{dy_i}{dx}$$

So, suppose you have a variable x you compute 2 functions from that one is x square, the other is x cube ok. I will call this as y_1 and I will call this as y_2 and I take y_1 and y_2 . And compute a z , which is say \log of y_1 by y_2 ok. Now what I am interested in is this, what is the answer for this? How do you get this? This is a fair question to ask y_1 y_2 are functions of x , z is a function of y_1 y_2 hence z is a function of x . So, I can compute this derivative and I can ask for this derivative, how would you compute it? If I cannot really do this right.

So, if this path did not exist, then it is trivial it is just the chain rule along one path, but now you have 2 paths. So, what will happen add them right. So, can you tell me a formula for that? So, let me know if this makes sense to you does this make sense now let me complicate this a bit just let me just do it as y_3 now.

Student: (Refer Time: 02:15).

What will happen?

Student: (Refer Time: 02:16).

That is all right? So, you see that if there are multiple paths you can just add up the chain rule across all these paths right? That is what chain will across multiple paths does ok.

(Refer Slide Time: 02:28)

Chain rule along multiple paths: If a function $p(z)$ can be written as a function of intermediate results $q_i(z)$ then we have :

$\frac{\partial \mathcal{L}(\theta)}{\partial h_{22}}$

40/57

Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

So, with this we will go back to this figure. So now, I am interested in going to the hidden layers, again I will do this to bit calculation where I first asked for this guy and then I will ask for the light blue guy right and am going to look at 1 unit at a time. Now what is the, what am I interested in the derivative of the loss function with respect to say $\frac{d}{dh_{22}}$, right? The second unit of the second hidden layer.

(Refer Slide Time: 02:58)

Chain rule along multiple paths: If a function $p(z)$ can be written as a function of intermediate results $q_i(z)$ then we have :

$$\frac{\partial p(z)}{\partial z} = \sum_m \frac{\partial p(z)}{\partial q_m(z)} \frac{\partial q_m(z)}{\partial z}$$

In our case:

- $p(z)$ is the loss function $\mathcal{L}(\theta)$
- $z = h_{ij}$
- $q_m(z) = a_{Lm}$

40/57

Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

Now, what I am going to say here is exactly what I had written on the previous slide this was our final function, right which was z . So, z was sorry again I have not chosen my

variables well ok, but if. So, we had exactly the same situation, right? Which is which you see here ok. So, we will just have to sum up the derivatives partial derivatives across all the paths, which lead from this guy to this guy and there could be as many paths as there can be, but I do not care I will just sum across all those paths. In fact, actually here there are not just 2 paths because we have always assumed there are k classes. So, there are actually k of these paths right.

So, this form this is exactly the formula which I wrote on the next slide right this one, but just written in terms of the network that we are dealing with ok. So, you can just go back and look at this, but as long as you understand this figure you from my point of view we can go ahead ok. So, everyone understands this figure that we just need to compute the derivatives across all the paths and add them up ok.

(Refer Slide Time: 04:03)

The slide contains the following elements:

- Handwritten Formula:**

$$\frac{\partial \mathcal{L}(\theta)}{\partial h_{ij}} = \sum_{m=1}^k \frac{\partial \mathcal{L}(\theta)}{\partial a_{i+1,m}} \frac{\partial a_{i+1,m}}{\partial h_{ij}}$$

$i+1 = L$

$h_{ij} \rightarrow h_{2,2}$

$$\begin{bmatrix} a_{3,1} \\ a_{3,2} \end{bmatrix} = \begin{bmatrix} w_{3,1} & w_{3,2} & w_{3,3} \\ w_{3,2} & w_{3,2} & w_{3,2} \end{bmatrix} \begin{bmatrix} h_{2,1} \\ h_{2,2} \\ h_{2,3} \end{bmatrix} + \begin{bmatrix} b_{3,1} \\ b_{3,2} \end{bmatrix}$$

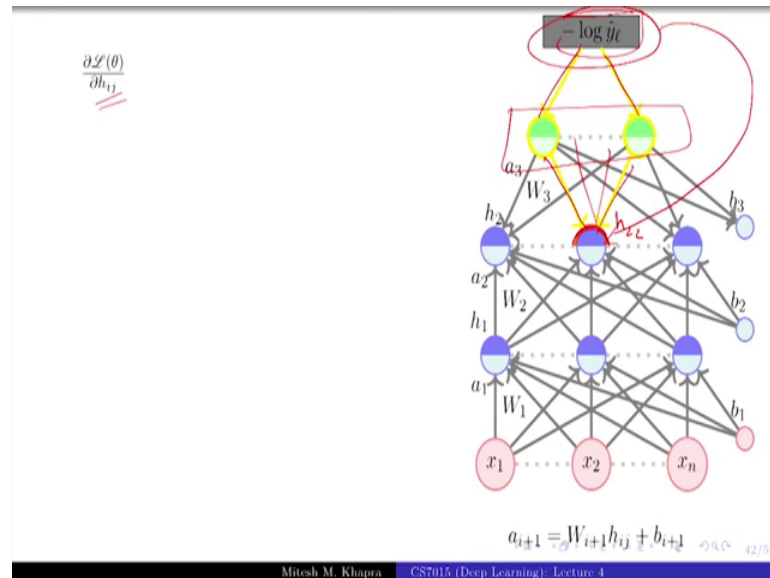
$$a_{3,1}^m = w_{3,1}h_{2,1} + w_{3,2}h_{2,2} + w_{3,3}h_{2,3} + b_{3,1}$$

$$\frac{\partial h_{ij}}{\partial a_{i+1,m}} = w_{3,1,2}$$
- Neural Network Diagram:** A diagram of a neural network with three layers. The input layer has nodes x_1, \dots, x_n . The first hidden layer has nodes h_1, h_2, h_3 with weights W_1 and biases a_1, a_2, a_3 . The second hidden layer has nodes h_1, h_2, h_3 with weights W_2 and biases b_1, b_2, b_3 . The output layer has nodes h_1, h_2, h_3 with weights W_3 and biases b_1, b_2, b_3 . A loss function $-\log \hat{y}_i$ is shown at the top. Red and yellow circles highlight specific nodes and connections.
- Video Inset:** A small video inset of the speaker, Mitesh M. Khapra, in the bottom right corner.
- Page-Footer:** Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

So now let us start we again the same recipe we will compute it with respect to one guy and then go towards the gradient ok. So, what is this now? Ok let me explain right. So, dl theta there are k of these guys between right. So, there are k paths. So, this summation has to happen over k paths just as you told me when there were 2 paths the summation was 2 3 paths to 3 that is k paths of the summation over k guys. The derivative with respect to each of these guys and the kth the mth unit rate that is the index that I am iterating over ok and then the derivative of this guy with respect to whatever you are interested ok.

That is just that there are only 2 nodes in the path in the chain, but there are k such chains, how many of you exactly get this? Ok how many of you have a problem want me to repeat this? You have problem oh many of you ok, good please do this. So, I am interested in this quantity; that means, I am interested in the partial derivative of this loss function with respect to this guy.

(Refer Slide Time: 05:06)



And this guy is nothing but h_{ij} that much is clear is the j th unit of the i th hidden layer. In fact, this is actually h_{22} . So, my i is equal to 2 and j is equal to 2 ok. Now I just made a case on the previous slide that, if you have such a function which first computes some intermediate values, and then your final function is computed based on all these intermediate values right. And now you are trying to find the gradient the partial derivative of this with respect to the original input that you had ok.

So, then what you will do is you will sum across all the paths that lead from this guy to the output ok, how many such paths are there? You already see 2 such paths here right, but I am saying there are k such paths, because there are some other nodes here which I have not drawn we have already said that in the output layer we have k nodes right? So, there are k paths. So, that takes care of the first bit that the summation is going to be over the k paths ok.

Now what is each of these paths composed of? This intermediate value and this quantity that we are interested in ok, first we will take the derivative of the out of the loss with

respect to this intermediate value, what is that? That is the unit in the, that is the unit in the previous layer or the next layer rather. So, I am interested in i . So, I am looking at the unit in the next layer hence $i + 1$ right, because that is what comes in my path the next layer is what comes in my path. We have always the special case right, that this guy feeds into k guys, but all the other hidden units before that feed into n guys right.

So, that is let us just keep that complication aside for the minute and we just look at this case ok, is that fine ok. So, we have agreed there are k paths and each path is composed of these 2 nodes, from the last loss function to this intermediate value and then from this intermediate value to the quantity of interest ok. And why is this $i + 1$ because the next node in the path when I am at the i th layer.

So, I will be feeding to the $i + 1$ th layer right? And in fact, I will be feeding to all the nodes in the $i + 1$ at layer, that is why I am taking or all the k paths right; and then that node which is this node with respect to the quantity, that I am interested, is this clear? Now right this is very similar to the toy example which I did I just have k paths now instead of 2 paths there is it clear now sure fine.

So, let us move ahead, now what is which of these quantities do we already know, is there any quantity that we know? This one, why? Because in this special case $i + 1$ is actually equal to L right. Because we are feeding into the last layer and they have already seen how to compute the partial derivatives with respect to the last layer.

So, this quantity is known we do not know this for the generic case yet, but we will get that, but for this special case when we are feeding into the last layer we know this does everyone get this? Ok now do we know this quantity. So, what you have told me is that we know this quantity because that is what we have computed in the previous module. Do we know this quantity? We have to compute it; can you compute it? Ok let us just do it right. So, let us assume that this h_{ij} that am dealing with is actually h_{22} fine. Now what is a $i + 1$ m ? Actually which are the elements there a 3_1 and a 3_2 , I am assuming that I only have 2 units in the output layer ok.

So, my m is equal to 2. Now is this fine, this is how the next layer is related to the current hidden layer plus biases ok. Now what am I interested in one of these guys ok. Let me take one of these guys. So, can you tell me what is a 3_1 , first row multiplied by the first column there is only one right plus b_{21} ok.

Student: (Refer Time: 10:11).

Sorry.

Student: (Refer Time: 10:12).

B 3 1 ok yeah fine. Now let me just clarify something what is this in terms of variables ijk m , what is this? This is i this is j this is k this is m . This is i plus 1 right ok. This is one of the m s that I am dealing with. Now I want the derivative of this with respect to h_{ij} ok. In fact, I want it with respect to h_{22} where this is i and this is j is this clear? What is this derivative, w_{312} everything everyone fine with this; ok? Now help me find this, what is this ijk and i plus 1, what is this? This is coming from the m , how many of you see this? Because that is the unit that you are connecting to and this is j . So, what is the formula; how many? As many as the number of neurons in the next layer a bias will be connected to all the neurons in that layer right? Everyone gets that right there are only 2 units.

So, there will be only 2 guys ok. So, what is the formula for this W_{i+1mj} , everyone comfortable with that fine? You can just go back and look at this and it should be cleared right. So, whenever you are dealing with vectors and matrices right if you are really good at it you can imagine the entries and figure out what is happening. If you are not good at it do not be lazy just work it out, right? You just need to write down this product and at the end remember everything is always element wise and you are never dealing with a vector or matrix now just dealing with the individual components of them.

So, you should always be able to compute these derivatives or partial derivatives with respect to the individual components, and that is exactly what I did here, right? Just work it out if you just write it out then you will always get it if you cannot, but eventually try to get to a point where you can just visualize it, but if you cannot at least try to work it out ok.

(Refer Slide Time: 12:25)

$$\frac{\partial \mathcal{L}(\theta)}{\partial h_{ij}} = \sum_{m=1}^k \frac{\partial \mathcal{L}(\theta)}{\partial a_{i+1,m}} \frac{\partial a_{i+1,m}}{\partial h_{ij}}$$

$$= \sum_{m=1}^k \frac{\partial \mathcal{L}(\theta)}{\partial a_{i+1,m}} W_{i+1,m,j}$$

Now consider these two vectors,

$$\nabla_{a_{i+1,\cdot}} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial a_{i+1,1}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{i+1,k}} \end{bmatrix}; W_{i+1,\cdot,j} = \begin{bmatrix} W_{i+1,1,j} \\ \vdots \\ W_{i+1,k,j} \end{bmatrix}$$

$W_{i+1,\cdot,j}$ is the j -th column of W_{i+1} ; see that,

$$(W_{i+1,\cdot,j})^T \nabla_{a_{i+1,\cdot}} \mathcal{L}(\theta) = \sum_{m=1}^k \frac{\partial \mathcal{L}(\theta)}{\partial a_{i+1,m}} W_{i+1,m,j}$$

Mitesh M. Khapra CS7015 (Deep Learning) Lecture 4

So, this is what it will look like ok. Now consider these 2 vectors one is this vector what does this vector look like, this is a collection of all the partial derivatives. So, this is just a collection of all the partial derivatives nothing new we have already seen this. Now what is this vector actually? In fact, I have started with the matrix and am saying look at this vector, what does this mean? This $i+1$ is just the layer in which the matrix is right. So, that index we do not really care about, for a matrix what we care about is the i comma j index ok. Now what does this dot comma j mean? All the is belonging to j ; that means, the dash column j th column everyone gets this, this is all the is or all the entries belonging to the j th column.

So, it is effectively just the j th column. So, it is 1 comma j 2 comma j up to k comma j right. So, these are 2 valid vectors, now tell me what is this quantity going to be? This is the dash between 2 vectors dot product dot product between 2 vectors is a.

Student: (Refer Time: 13:43).

Is a summation over element wise thing ok. I have said enough now try to connect this is a very simple match the column that you will ever get in your life, try to connect this to something which is already there in the slide. How many of you think the answer is this? This into this plus this into this plus this into this and just write it as a formula you will get this everyone sees that ok. So now, I have a compact way of writing one of these entries ok.

(Refer Slide Time: 14:10)

We have, $\frac{\partial \mathcal{L}(\theta)}{\partial h_{ij}} = (W_{i+1, \cdot j})^T \nabla_{a_{i+1}} \mathcal{L}(\theta)$

We can now write the gradient w.r.t. h_i

$$\nabla_{h_i} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial h_{i1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial h_{i2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial h_{in}} \end{bmatrix} = \begin{bmatrix} (W_{i+1, \cdot 1})^T \nabla_{a_{i+1}} \mathcal{L}(\theta) \\ (W_{i+1, \cdot 2})^T \nabla_{a_{i+1}} \mathcal{L}(\theta) \\ \vdots \\ (W_{i+1, \cdot n})^T \nabla_{a_{i+1}} \mathcal{L}(\theta) \end{bmatrix}$$

$$= (W_{i+1})^T \nabla_{a_{i+1}} \mathcal{L}(\theta)$$

- We are almost done except that we do not know how to calculate $\nabla_{a_{i+1}} \mathcal{L}(\theta)$ for $i < L-1$
- We will see how to compute that

43/57

Mitesh M. Khapra CS7015 (Deep Learning) Lecture 4

One of these guys I have a compact way of writing this, it happens to be the dot product between 2 vectors one of them is the gradient, but do I know this already; do I know this quantity already? In this special case yes, because I plus 1 is equal to L and that I have already computed this of course, I know right because these are the weights that I am dealing with, where do I go from here? This dot yeah it means anything from that column so; that means, the entire column.

Student: (Refer Time: 14:48).

Ah no these are weights right. So, this is a weight matrix it has columns and rows. So, am talking about the j th column. So, I fixed the value of j. So, I am talking about the j th column, but I am not telling your given ith entry there am just telling you all the entries there that just means the j th column you can take this offline ok. This is very simple I will take it offline ah. Now where do I go from here.

Student: (Refer Time: 15:16).

I plus 1.

Student: (Refer Time: 15:20)

No in this specific case are we done.

Student: (Refer Time: 15:27).

Where are we right now with respect to one unit, where do we want to go? The entire thing. So, what is the quantity that I am interested in gradient with respect to always say with respect to h_i right.

Student: (Refer Time: 15:44)

Where i is 2 in this case this special case ok. What is that going to be collection of all these guys that you have already computed ok. Now simplify this. What is this first column of the matrix? Multiplied by the same vector the second column of the matrix, multiplied by this vector, the n th column of the matrix multiplied by this vector this reminds you of something very, very difficult. This is a very, very complicated matrix multiplication right?

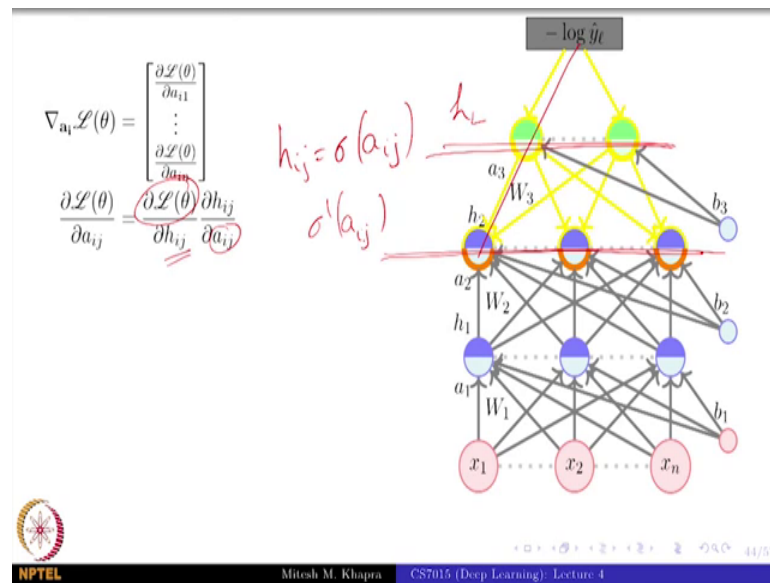
First row of the matrix multiplied by a column the second row of the matrix multiplied by column, how many if you get this; right? So, this is can you tell me what this is W_i plus 1 transpose.

Student: (Refer Time: 16:52).

Perfect right? So now, you see that this entire quantity we can compute in one go by using a matrix vector multiplication right. So, that is what I meant; when I was saying that we should not be doing these unusual computations, but we able to compute that at one row right. So now, we can just do this matrix vector multiplication and get this entire quantity ok. Now what is still missing in this module.

So, what is the special case that I have assumed, I told you that I already know these quantities, but only if $i + 1$ is equal to l . I need to tell you this in the generic case ok. So, we are almost there except that I do not know this when i is not equal to L or i is less than equal to $L - 1$ ok; that is the case that I am looking for.

(Refer Slide Time: 17:38)



So, that is again very simple, again what will I do? I will compute it with respect to a_i . What is this? This is the guy that I am interested in the generic i not the L th one right the generic i . This is what the vector looks like the gradient vector looks like. I want each of these guys a_i . Now I will take one of those and I will write it as this a_i . What am I doing? Am saying that, I already have the entries up to here a_i ok at a very general level even here I could have said the same thing, remember that I had said that the output layer you can always write as h_L , right?

So, even at the output layer I could say this chain rule always holds, how many of you agree with that? I want to go from the loss function to one of the lighter blue guys. So, am saying that I can go through the intermediary dark blue guys, that is all I am saying. I have just compressed this entire path into up to the dark blue guy. Remember I had said earlier that I will be compressing this chains, is this clear to everyone? Ok. Now how many of these quantities do you know? The first one is what we computed on the previous (Refer Time: 18:52) ok. The second one looks very difficult sorry.

So, h_{ij} is nothing but sigmoid of a_{ij} or any non-linearity of the a_{ij} . So, I can just write this derivative as I will just write it as σ' ok.

(Refer Slide Time: 19:10)

The slide contains the following mathematical derivations:

$$\nabla_{\mathbf{a}_i} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial a_{i1}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{in}} \end{bmatrix}$$

$$\frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} = \frac{\partial \mathcal{L}(\theta)}{\partial h_{ij}} \frac{\partial h_{ij}}{\partial a_{ij}}$$

$$= \frac{\partial \mathcal{L}(\theta)}{\partial h_{ij}} g'(a_{ij}) \quad [: h_{ij} = g(a_{ij})]$$

$$\nabla_{\mathbf{a}_i} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial h_{i1}} g'(a_{i1}) \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial h_{in}} g'(a_{in}) \end{bmatrix}$$

$$= \nabla_{h_i} \mathcal{L}(\theta) \odot [\dots, g'(a_{ik}), \dots]$$

The diagram shows a neural network with three layers of nodes. The input layer has nodes x_1, \dots, x_2 . The hidden layer has nodes h_1, h_2 . The output layer has nodes a_1, a_2, a_3 . Weights W_1, W_2, W_3 connect the layers. Biases b_1, b_2, b_3 are also shown. The loss function is $-\log \hat{y}_t$. The diagram highlights the backpropagation of gradients from the output layer back to the input layer.

NPTEL
Mitesh M. Khapra CS7015 (Deep Learning) Lecture 4

Or g prime is this fine ok. Now I have it with respect to 1 unit, what will I do? Go to the gradient fit it all these values. Now simplify this, what is this? A vector right, what is this? Another vector, there is a one to one correspondence between them. So, you have 2 vectors and you are doing a 1 to 1 multiplication, what is this?

Student: (Refer Time: 19:43).

How many of you say dot product? Dot product is always a, what is the output here?

Student: Vector.

Can it be a dot product; can it be a dot product? No please empathic no ok. So, what is it going to be? An element wise multiplication and this is how you denote that ok. So, what is this called? You had a mud product right. So, this is every element of one vector multiplied by the corresponding element of the other vector ok. So now, again the entire vector we can compute at 1 row right; I am not I am when I am teaching this I am telling you how to compute one element and then go to the gradient, but when you are going to implement this we are just going to compute the gradient at one go is that clear? Ok.