



deeplearning.ai

Learning from multiple tasks

Multi-task learning

So whereas in transfer learning, you have a sequential process where you learning from task A and then transfer that to task B.

In multi-task learning, you start off simultaneously trying to have one neural network do several things at the same time. And then each of these task helps hopefully all of the other task.

Simplified autonomous driving example



$x^{(i)}$

Pedestrians

Cars

Stop signs

Traffic lights

⋮

$y^{(i)}$

0

1

1

0

⋮

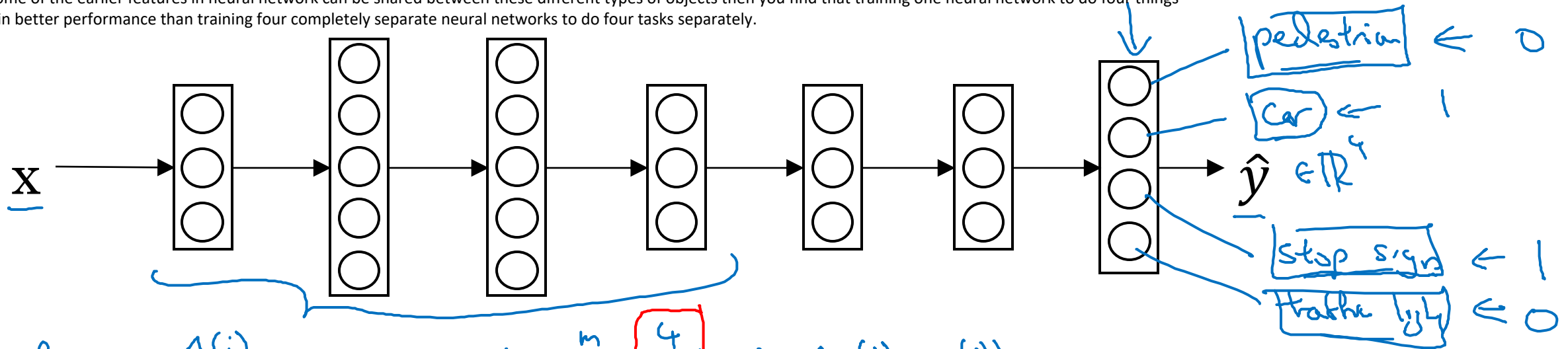
$(4, 1)$

$$Y = \begin{bmatrix} y^{(1)} & y^{(2)} & y^{(3)} & \dots & y^{(m)} \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix}$$

$(4, m)$

Neural network architecture

But if some of the earlier features in neural network can be shared between these different types of objects then you find that training one neural network to do four things results in better performance than training four completely separate neural networks to do four tasks separately.



Loss: $\frac{1}{m} \sum_{i=1}^m \sum_{j=1}^4 \mathcal{L}(\hat{y}_j^{(i)}, y_j^{(i)})$

Sum only over value of j with 0/1 label.

Usual logistic loss: $-y_j^{(i)} \log \hat{y}_j^{(i)} - (1 - y_j^{(i)}) \log (1 - \hat{y}_j^{(i)})$

Unlike softmax regression:
One image can have multiple labels

Multi-task learning \leftarrow

$$Y = \begin{bmatrix} 1 & 1 & \dots & 1 & ? & \dots \\ 0 & 1 & \dots & 1 & 1 & \dots \\ ? & ? & \dots & ? & ? & \dots \\ \vdots & \vdots & \dots & \vdots & \vdots & \dots \end{bmatrix}$$

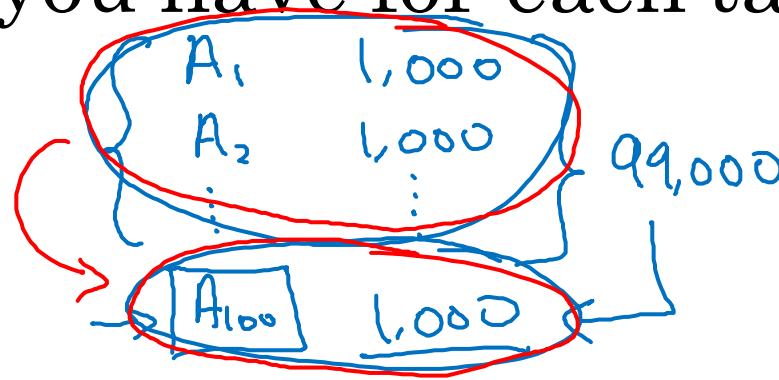
You can still train your learning algorithm to do four tasks at the same time, even when some images have only a subset of the labels and other are sort of question marks or don't care.

When multi-task learning makes sense

- Training on a set of tasks that could benefit from having shared lower-level features.
- Usually: Amount of data you have for each task is quite similar.

Second, this is less of a hard and fast rule, so this isn't always true.

A 1,000,000
↓
B 1,000



These in aggregate have 99,000 training examples which could be a big boost, could give a lot of knowledge to augment this otherwise, relatively small 1,000 example training set that you have for task A100.

- Can train a big enough neural network to do well on all the tasks.

If you can train a big enough neural network, then multi-task learning certainly should not or should rarely hurt performance. And hopefully it will actually help performance compared to if you were training neural networks to do different tasks in isolation.

In practice, multi-task learning is used much less often than transfer learning.