EE5934/EE6934 Q&A Assignment

Jin Lexuan A0232696W

April 2022

1 Q1

For self-attention layer, the order of output will change according to the order of input. As a result, after being permuted order, the order of input images will be modified, after sequently, experiencing the self-attention layer, a different ordered outputs will be obtained. However, there is a permutation back to recover the original order. So the result will be the same feature X_1 equals to X_2.

2 Q2

RNN is Recurrent Neural Network. It is not necessary to to use different parameter sets for different input vectors. The advantage of RNN is that it can process any length input and the model size does not increase for longer input. Also, same weights are applied on every timestep, so there is symmetry in how inputs are processed.

3 Q3

3.1 R-CNN

Step of R-CNN: Firstly, use the selective search algorithm to extract about 2000 Regions of Interest Proposals from top to bottom in the image. Then each Region Proposal is warped to a size of $224 * 224$ and input it to the CNN, and utilize the output of the fc7 layer of the CNN as a feature. The CNN features extracted by each region proposal is input of SVM for classification; For the last step, perform bounding regression for the Region Proposal classified by SVM, correct the original suggested window with the bounding box regression value, and generate the predicted window coordinates.

Defect of R-CNN: The training is divided into multiple stages, and the steps are troublesome, including fine-tuning the network, training the SVM training the bounding box regressor; As a result, training is time-consuming and takes up a lot of memory space. The speed of test process is also slow. Each candidate region needs to run the entire forward CNN calculation. SVM and regression are post-hoc operations, and CNN features are not learned and updated during the SVM and regression process.

3.2 Fast R-CNN

Improvement of Fast R-CNN: A ROI pooling layer is added after the last convolutional layer. The loss function uses a multi-task loss function, and the border regression is directly added to the CNN network for training.

The reason of why Fast R-CNN is Fast: Fast R-CNN normalizes the entire image and sends it directly to CNN. On the feature map output by the last convolutional layer, the proposed frame information is added, so that the previous CNN operations can be shared. During training, only one image is sent to the network, CNN features and region proposals are extracted from each image at one time, and the training data enters directly into the loss layer in the GPU memory, consequently the features of the first few layers of the candidate region do not need to be recalculated and a large amount of data is no longer necessary to be stored on the hard disk. Fast R-CNN integrates category judgment and position regression with a deep network, and no additional storage is required.

3.3 Faster R-CNN

Improvement of Faster R-CNN: Faster R-CNN uses RPN, which is shorted for Region Proposal Network, instead of the original selective search method to generate a proposal window. The CNN that generates the proposal window is shared with the CNN for object detection.

The reason of why Faster R-CNN is Faster: Faster R-CNN creatively applies the convolutional network to generate the proposal frame by itself, and shares the convolutional network with the target detection network, so that the number of proposal frames is reduced, and the quality of the proposal frame is also substantially improved. Use RPN to first generate a bunch of Anchor boxes, cut and filter them, and then use softmax to judge whether the anchors belong to the foreground or the background, in another word, objects or not objects, so this is a two-category classification problem; At the same time, another branch is bounding Box regression corrects the anchor box to form a more accurate proposal.

4 Q4

For overlapping detection problems, the solution is the post-process raw detections using Non-Max Suppression(NMS):

1. Select next highest-scoring box.
2. Eliminate lower-scoring boxes with IoU greater than threshold.
3. If any boxes remain, GOTO 1.

In the step, IoU means *intersection oover Union*, the formula is:

$$\frac{Area\ of\ Intersection}{Area\ of\ Union}$$

5 Q5

The network for semantic segmentation is a bunch of convolutional layers, with downsampling and upsampling inside the network. For a specific dimension, $H \times W \times C$, the downsampling process transfer it to high-res of $D_1 \times H/2 \times W/2$ and then med-res $D_2 \times H/4 \times W/4$ and final low-res $D_3 \times H/4 \times W/4$. The upsampling is the inverse process of downsampling to transfer it back to $H \times W \times C$.

There are several methods to realize upsampling. The first one is unpooling, includes Nearest Neighbor method, "Bed of Nails" method and max unpooling. When in the max pooling period, it is required to remember which element was max, then pass through the layers. When in the max unpooling stage, use positions from pooling layer to recover them.

Another approach is Learnable Upsampling: Filter moves 2 pixels in the input for every one pixel in the output. Stride gives ratio between movement in input and output. This strided convolution is interpreted as "learnable downsampling".

The convolution in terms if a matrix multipulation can be expressed as:

$$\vec{x} * \vec{a} = X\vec{a}$$

Transposed convolution multiplies by the transpose of the same matrix:

$$\vec{x} *^T \vec{a} = X^T\vec{a}$$

A 3×3 convolution transpose with stride of 2 gives 5×5 output which is necessary to trim one pixel from top and left to give 4×4 output. For the input, weight filter by input value and copy to output. For the output, sum where output overlaps.

6 Q6

The structure of 3D convolution and R(2+1)D convolution is presented in Figure 1. Full 3D convolution is carried out using a filter of size $t \times d \times d$ where t denotes the temporal extent and d is the spatial width and height. A R(2+1)D convolutional block splits the computation into a spatial 2D convolution followed by a temporal 1D convolution[1].

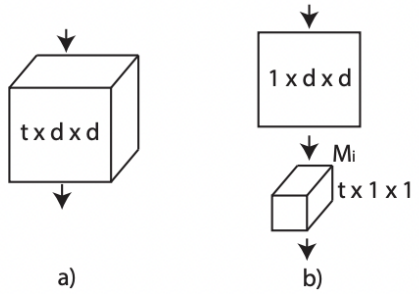

Figure 1: (2+1)D vs 3D convolution[1]

The first layer of P3D is 2D convolution, followed by the P3D module, and the R(2+1)D network is this module from the beginning;The R(2+1)D module calculates the hyperparameters, and by increasing the number of channels, the decomposed R(2+1)D model has the same amount of parameters as the previous 3D model.

Compared with 3D convolution, R(2+1)D has two advantages. Firstly, although the number of parameters is not changed, it doubles the number of non-linearity in the network due to the extra Relu between the 2D and 1D convolutions in each block. Growth the number of non-linearity increases the complexity of the representative function. The effect of large filters is approximated by applying multiple smaller filters. The second benefit is that 3D convolution is forced to separate the spatial and temporal components, making optimization easier. This shows that the training error is lower compared to a 3D convolutional network of the same capacity.

7 Q7

The diagram of the basic inception block with specific feature sizes is shown in Figure 2.

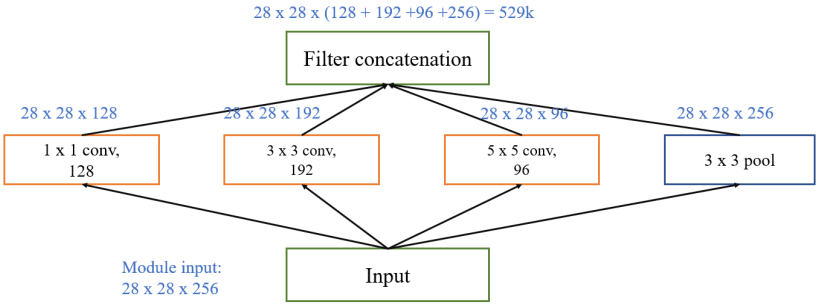

Figure 2: basic inception block

References

[1] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.