Lecture Notes

Transposed Convolution

Debapriya Roy
Senior Research Scholar
Indian Statistical Institute, Kolkata

I. Convolution arithmetic

For any input size i, kernel size k, stride s and padding p The formula for convolution is,

$$o = \lfloor \frac{i + 2p - k}{s} \rfloor + 1 \tag{1}$$

II. Transposed Convolution arithmetic

From Eq. 1 we have,

$$\begin{aligned} o &= \frac{i + 2p - k}{s} + 1 \text{ (disregarding floor for notational simplicity)} \\ o - 1 &= \frac{i + 2p - k}{s} \\ s(o-1) &= i + 2p - k \\ i &= s(o-1) + k - 2p \end{aligned}$$

Now for the given i and o we consider the equivalent transposed convolution that has input o and output size i. However instead of o and i we use the notation o', i' to denote the case of transposed convolution. Therefore the formula of output size for transposed convolution becomes,

$$o' = s(i' - 1) + k - 2p \text{ (using } o' \text{ in place of i and } i' \text{ in place of o)} \tag{2}$$

A. *How to get padding and internal zero padding*

Internally a transposed convolution layer computes two things,

- Number of zero insertions within the input (say z') and
- padding (say, p').

Now how to find these two numbers.

- We know that with the computed padding and other factors the transposed convolution works same as convolution. Therefore, considering T' as the total padding we get,

$$\begin{aligned} o' &= \frac{i' + T' - k}{s'} + 1 \text{ (considering } T' \text{ as the total padding)} \\ i &= \frac{o + T' - k}{s'} + 1 \text{(since we know } i' = o \text{ and } o' = i) \\ i &= o + T' - k + 1 \text{(assuming } s' \text{ =1)} \end{aligned} \tag{3}$$

From the formula of convolution we know,

$$
\begin{aligned}
o &= \lfloor\frac{i+2p-k}{s}\rfloor + 1 \\
o &= \frac{i+2p-k}{s} + 1 \text{(disregarding floor for simplicity)} \\
s(o-1) &= i+2p-k \\
i &= s(o-1) - 2p + k
\end{aligned} \tag{4}
$$

Using Eq. 4 in Eq. 3 we get,

$$
\begin{aligned}
i &= o + T' - k + 1 \\
s(o-1) - 2p + k &= o + T' - k + 1 \\
T' &= s(o-1) - 2p + k - o + k - 1 \\
T' &= s(o-1) - 2p + k - o + k - 1 - 1 + 1 \text{(introducing -1 and +1)} \\
T' &= s(o-1) - 2p + k - o + 1 + k - 2 \\
T' &= s(o-1) - 2p + k - (o-1) + k - 2 \\
T' &= (o-1)(s-1) - 2p + 2k - 2 \\
T' &= (o-1)(s-1) + 2(k-p-1)
\end{aligned}
$$

So the total padding that we require is,

$$T' = (o-1)(s-1) + 2(k-p-1)$$

Using the notation of the transposed convolution,

$$T' = (i'-1)(s-1) + 2(k-p-1) \tag{5}$$

So this padding has two parts -

- one part dependent on the input size
- and the other part involving the kernel size and the padding of the equivalent convolution

Now the question comes how to apply this padding to the input.
One solution is we may apply the whole padding as the traditional way padding is applied.
But there is a problem if we apply this padding only on the sides of the input. Then the top left input may not be associated to the top left pixel of the output. Thus a meaningless computation of feature value.
Hence we split this into two halves. A part is applied as padding and another part is applied as zero insertions within the input.