Lecture 3: Convolutional Neural Networks

April 12th, 2021

IT IS FRIDAY, ALL MY FRIENDS ARE AT PARTY

BUT, I'M HERE LEARNING BACK PROPAGATION
Expectations of visual recognition network

1. Maintain 2D structure logic
2. Shift invariant (actually, equivariant)
3. Consider only local correlations
4. Hierarchically growing field of view
5. Hierarchically progressing complexity
6. Reasonable amount of params
1. Maintain 2D structure logic
2. Shift invariant (actually, equivariant)
3. Consider only local correlations
4. Hierarchically growing field of view
5. Hierarchically progressing complexity
6. Reasonable amount of params
Convolution layer

Toeplitz matrix
Q: Given input size and filter size, find output size.

Q: is this a convolution?
A: Yes, but with the flipped filter. This is cross-correlation.
Stride

\[ x^l_0 \rightarrow \sum \rightarrow x^{l+1}_1 \]

\[
\begin{bmatrix}
    a & b & c & 0 & 0 & 0 & 0 \\
    0 & a & b & b & c & 0 & 0 \\
    0 & 0 & a & b & a & b & 0 \\
    0 & 0 & 0 & a & b & c & 0 \\
    0 & 0 & 0 & 0 & a & b & c \\
    0 & 0 & 0 & 0 & 0 & a & b \\
\end{bmatrix}
\]
Padding

```
x_{0}^{l}  \sum \sum x_{1}^{l}  \sum x_{2}^{l}  \sum x_{3}^{l}  \sum x_{4}^{l}  \sum x_{5}^{l}  \sum 0
x_{0}^{l+1}  \sum \sum x_{1}^{l+1}  \sum x_{2}^{l+1}  \sum x_{3}^{l+1}  \sum x_{4}^{l+1}  \sum x_{5}^{l+1}  \sum 0
```

“Same”
BTW: sometimes, you have to pad
Conv2D

$N, C, H, W$

$C, h, w$
Conv2D

\[ N, C, H, W \]

\[ [N, 1, \tilde{H}, \tilde{W}] \]
Conv2D

$[N, C_{in}, H, W]$  \rightarrow  $[[C_{out}, C_{in}, h, w]]$  \rightarrow  $[N, C_{out}, \tilde{H}, \tilde{W}]$
Receptive Field
Receptive Field
Convs rock!

1. Maintain 2D structure logic
2. Shift invariant (actually, equivariant)
3. Consider only local correlations
4. Hierarchically growing field of view
5. Hierarchically progressing complexity
6. Reasonable amount of params
Transposed Convolution

\[
\begin{bmatrix}
    b & a & 0 & 0 & 0 & 0 & 0 & 0 \\
    c & b & a & 0 & 0 & 0 & 0 & 0 \\
    0 & c & b & a & 0 & 0 & 0 & 0 \\
    0 & a & c & b & a & 0 & 0 & 0 \\
    0 & 0 & a & c & b & a & 0 & 0 \\
    0 & 0 & 0 & 0 & c & b & a & 0 \\
    0 & 0 & 0 & 0 & 0 & c & b & 0
\end{bmatrix}
\]
Transposed Convolution with stride

Recall- stride 2 conv:

\[
\begin{bmatrix}
a & b & c & 0 & 0 & 0 & 0 \\
0 & 0 & a & b & c & 0 & 0 \\
0 & 0 & 0 & 0 & a & b & c
\end{bmatrix}
\]
Transposed Convolution by dilation & flip

\[ x_0^{l+1}, x_1^{l+1}, x_2^{l+1}, x_3^{l+1}, x_4^{l+1}, x_5^{l+1}, x_6^{l+1} \]

\[ \sum \]

\[ a, b, c \]
Q: How do you backprop a Conv?

Transposed Conv with the same filter! (but be careful with padding)

Q: How about Conv2D?

1. Transpose filter $c_{in}, c_{out}$
2. Transposed Conv2D with the modified filter
   2.1. Dilate input (in both spatial dims)
   2.2. Flip filter in both spatial dims
   2.3. Conv2D with the modified filter (careful with padding)
Two important intuitions about feature maps
Two important intuitions about feature maps
Max Pooling

- Usually stride=win-size, but not always.
- Each channel separately.

Input:

```
  7  3  5  2
  8  7  1  6
  4  9  3  9
  0  8  4  5
```

Output:

```
  8  6
  9  9
```
ConvNet Example

Fukushima 1980
LeCunn et al. 1989
LeCunn et al. 1998
Krihzevsky et al. 2012
More special Convs!

Depthwise Conv

Group Conv (Krizhevsky 2012)

Dilated Conv (Yu & Koltun 2016)

Group Conv (Krizhevsky 2012)

Input Features

Output Features

Input Features

Output Features

Convolution

Convolution

Convolution

Group 1

Group 2

Group 3

Depthwise Conv

Convolution

Convolution

Convolution

dilation=1
dilation=2
dilation=3
This week no tutorial (Memorial Day)

Next week’s lecture:

Practical Training

Shai Bagon

My Neural Net

Initialization
Augmentation
Regularization
DropOut
BatchNorm
Hyperparam tuning