

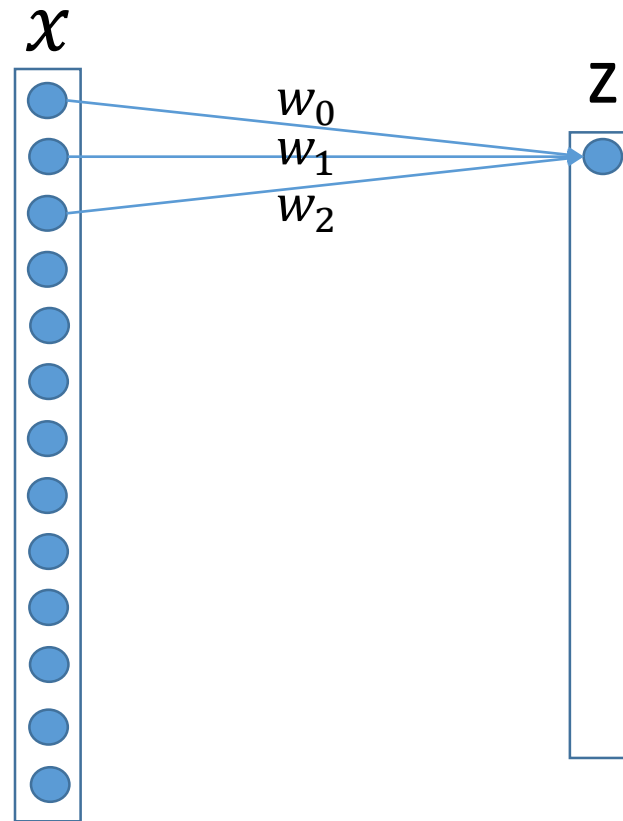
# Topic Text Highlighting by Deconvnet

李朋軒

2016/06/09

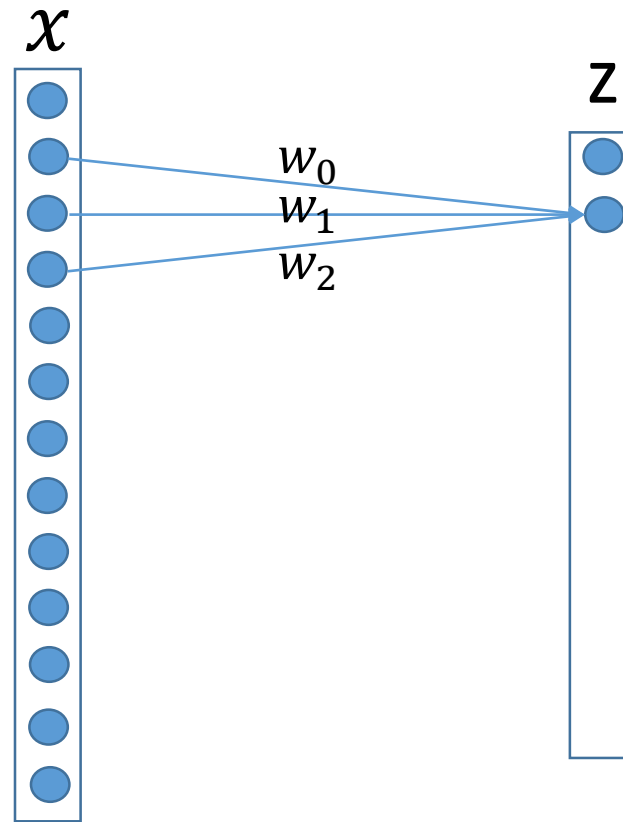
# Convolution

- Vector  $x$
- Vector  $w := \text{kernel/filter}$
- Vector  $z := \text{conv}(x, w)$



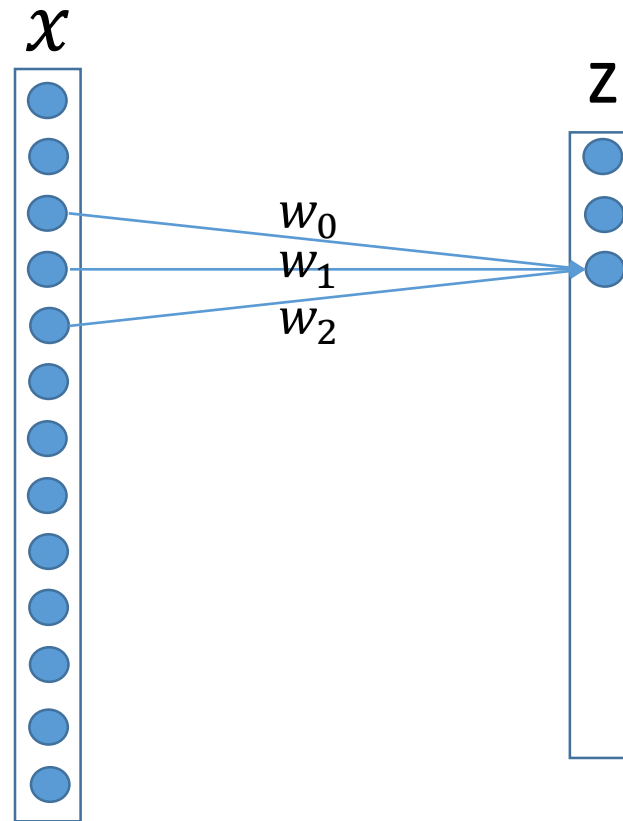
# Convolution

- Vector  $x$
- Vector  $w := \text{kernel/filter}$
- Vector  $z := \text{conv}(x, w)$



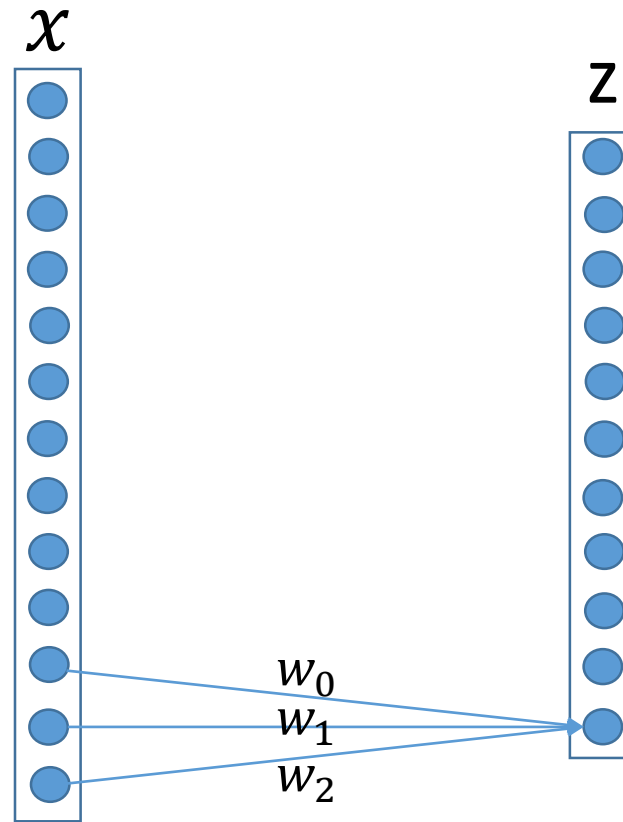
# Convolution

- Vector  $x$
- Vector  $w := \text{kernel/filter}$
- Vector  $z := \text{conv}(x, w)$



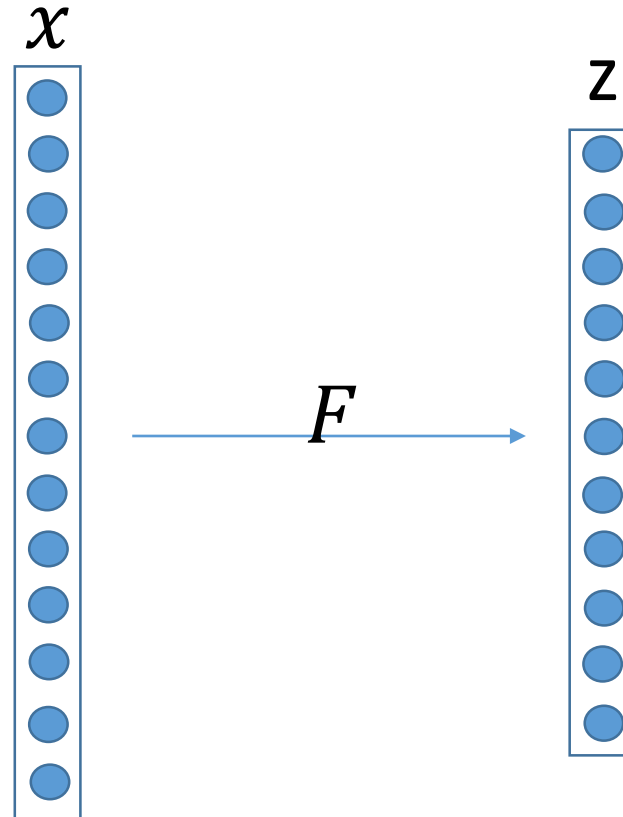
# Convolution

- Vector  $x$
- Vector  $w := \text{kernel/filter}$
- Vector  $z := \text{conv}(x, w)$



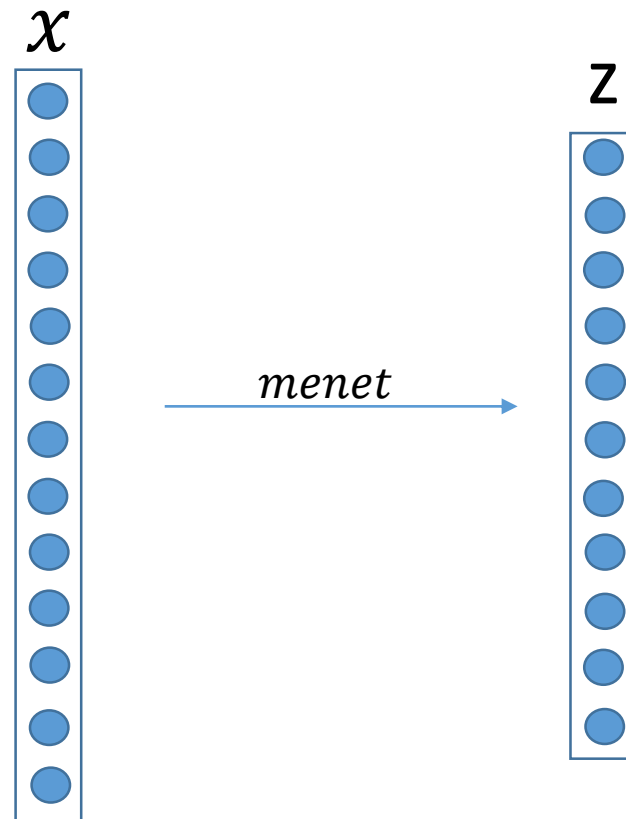
# Convolve by Matrix

- Vector  $x$
- Vector  $w := \text{kernel/filter}$
- Vector  $z := \text{conv}(x, w)$
- Exist matrix  $F$  s.t.  $Fx = \text{conv}(x, w)$



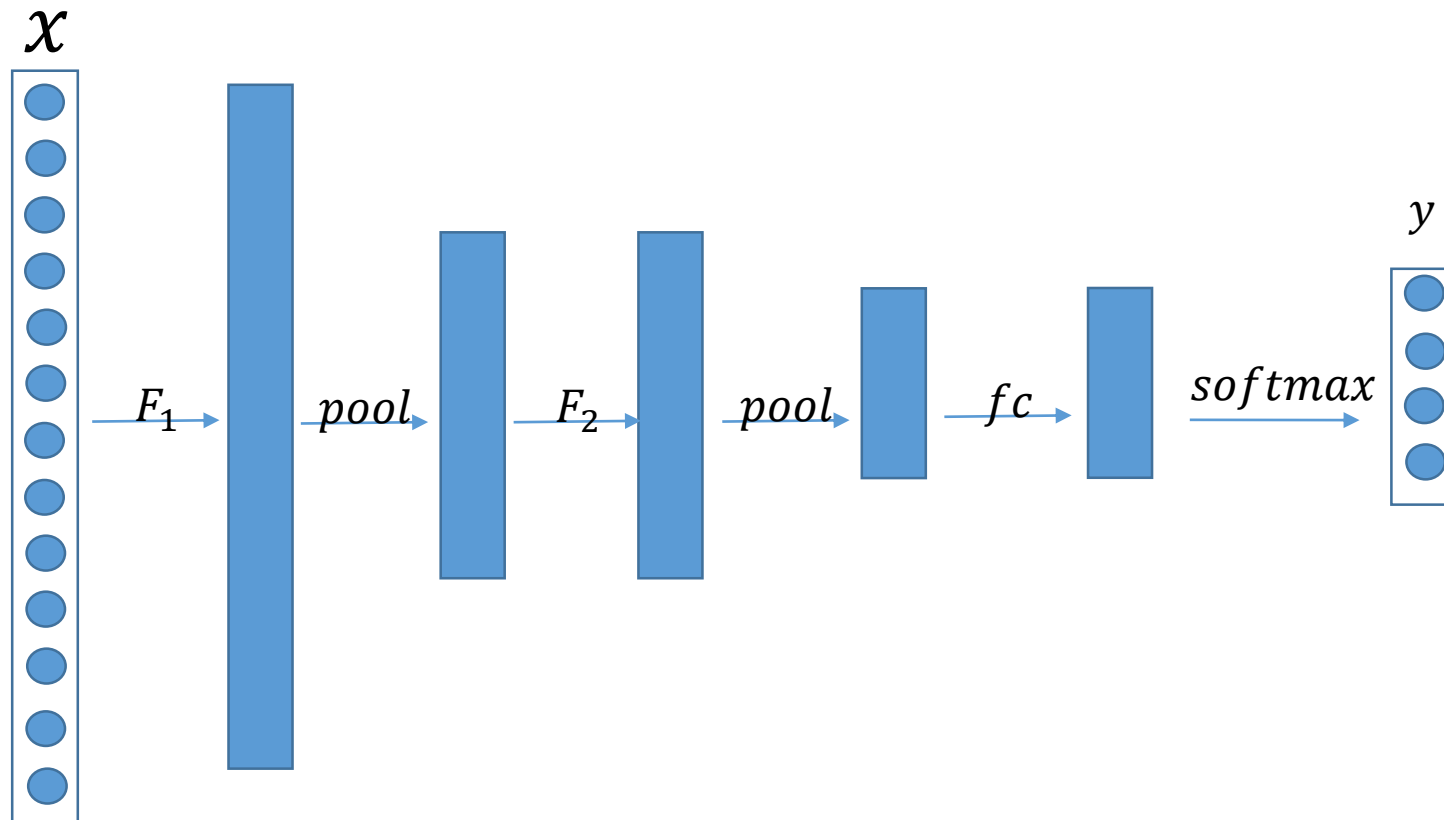
# Convolutional Neural Network (CNN)

- Let  $menet$  be  $(conv1, pool, conv2, pool, fc3, softmax)$
- $y :=$  probability distribution of 4 classes



# Convolutional Neural Network (CNN)

- Let  $menet$  be  $(conv1, pool, conv2, pool, fc3, softmax)$
- $y :=$  probability distribution of 4 classes

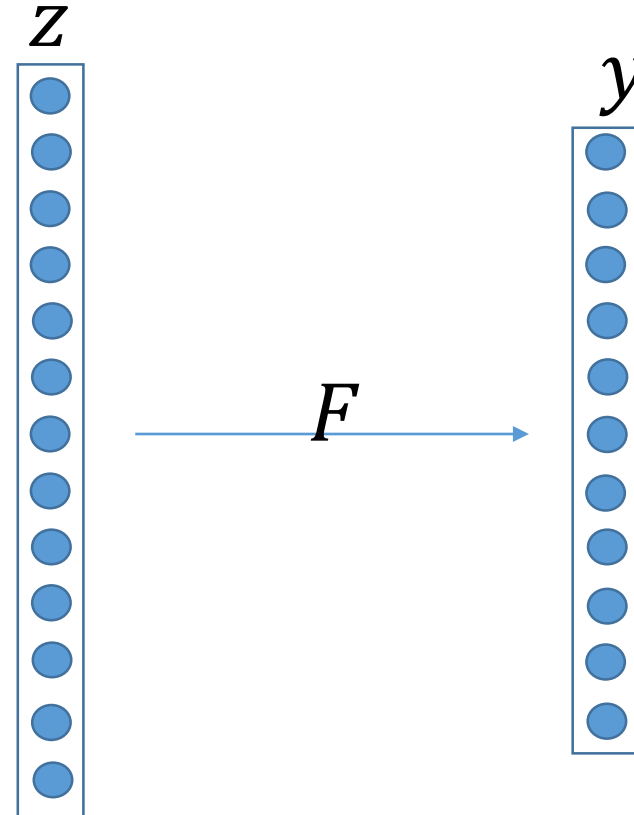




# Deconvnet (Zeiler et al., 2010)

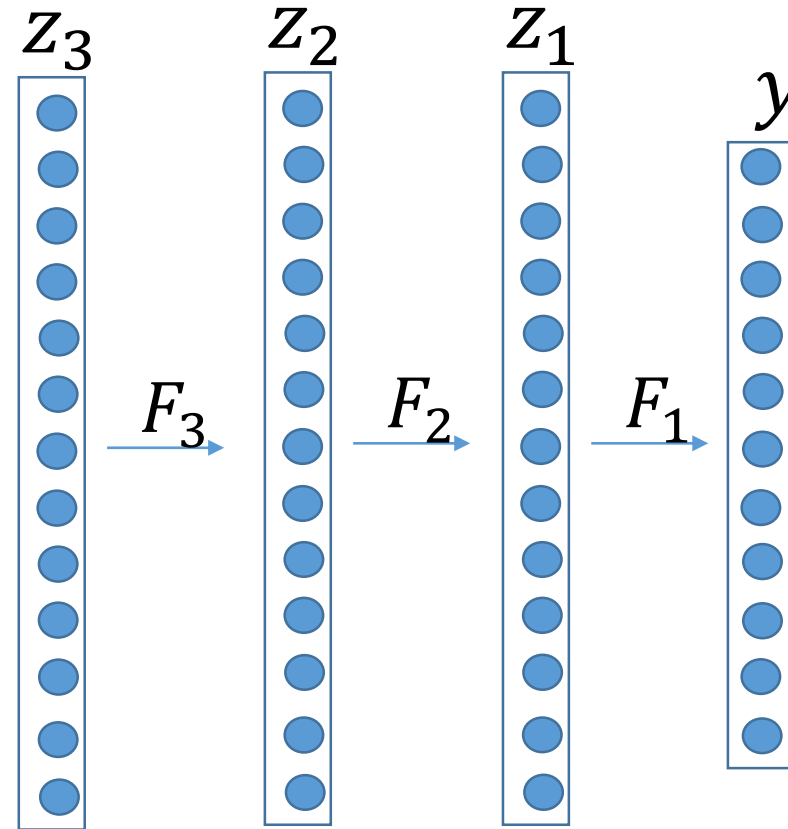
- Inference
  - Given model and  $y$ , find  $z$
- Training
  - Given many  $y$ 's, find best  $F$

$$C_1(y^i) = \frac{\lambda}{2} \sum_{c=1}^{K_0} \left\| \sum_{k=1}^{K_1} z_k^i \oplus f_{k,c} - y_c^i \right\|_2^2 + \sum_{k=1}^{K_1} |z_k^i|^p$$



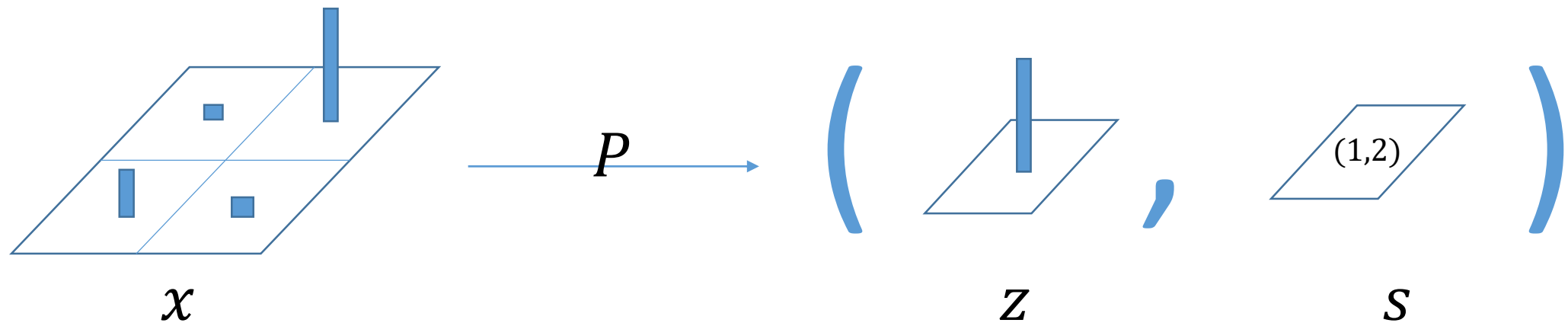
# Deconvnet (Zeiler et al., 2010)

- $y$ 
  - an image
- Inference
  - Given model and  $y$ , find  $z_3$
  - Feature extraction



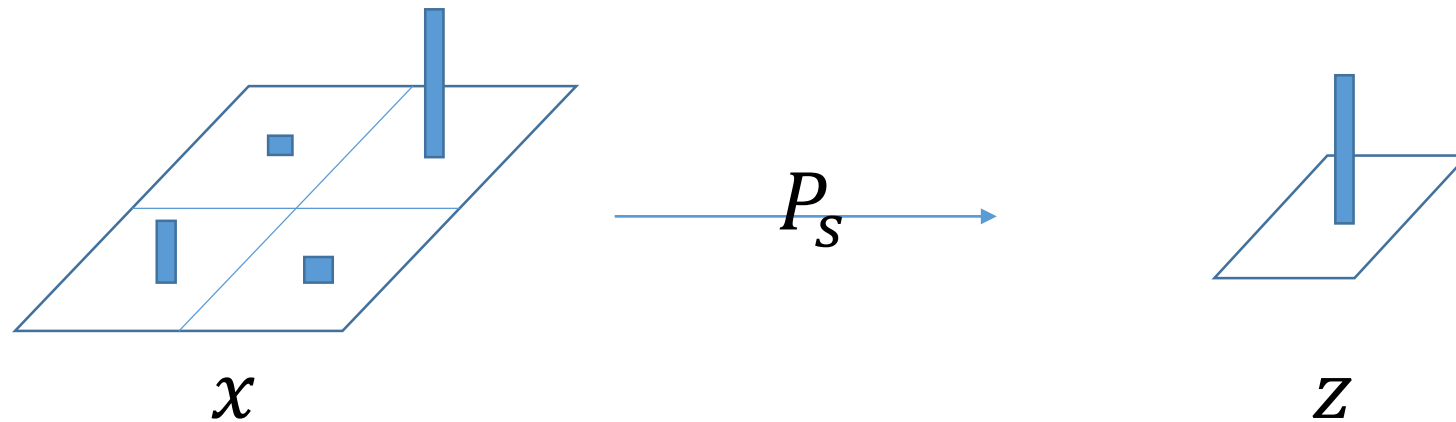
# Adaptive Deconvnet (Zeiler et al., 2011)

- Max pooling
  - $n \times n$  matrix  $x$
  - $n/2 \times n/2$  matrix  $z$
  - $n/2 \times n/2$  matrix  $s$  – argmax/switch



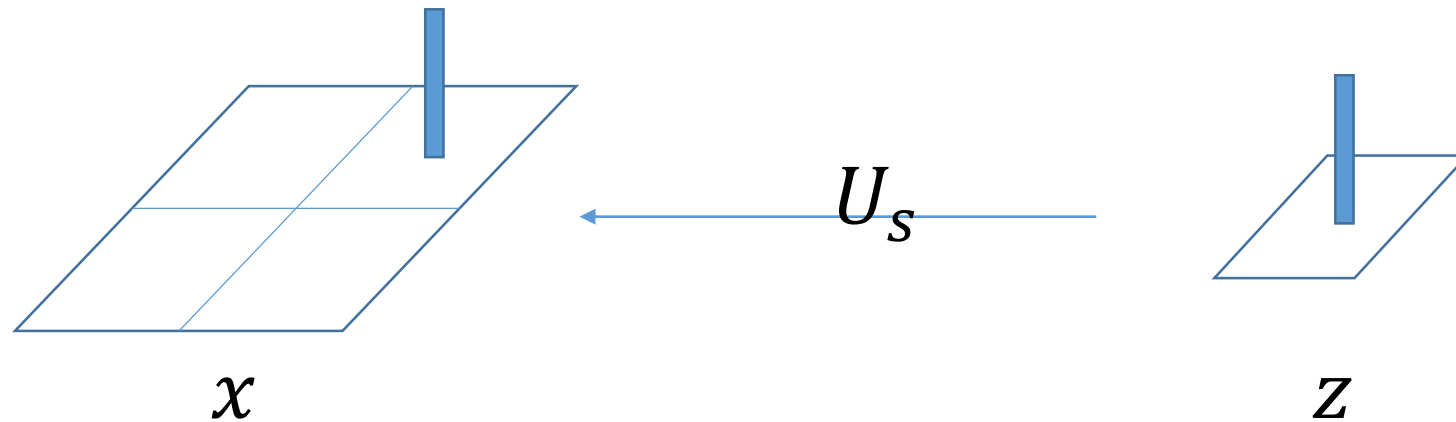
# Adaptive Deconvnet (Zeiler et al., 2011)

- Switch pooling
  - Vector  $x$
  - Matrix  $P_S$
  - Vector  $z := P_S x$



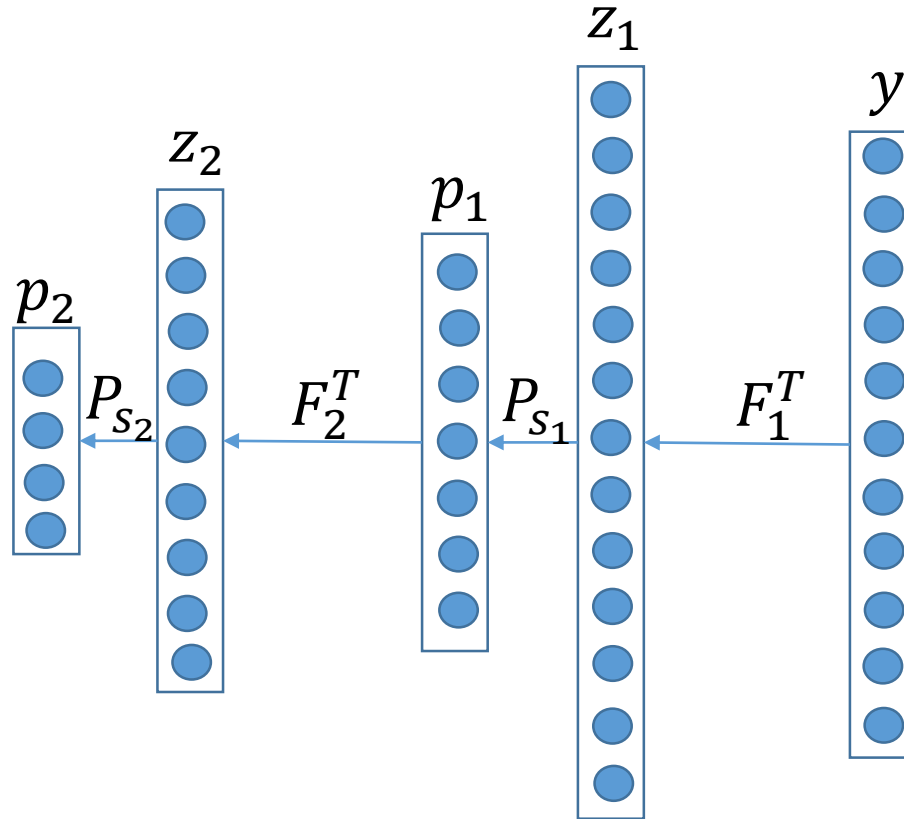
# Adaptive Deconvnet (Zeiler et al., 2011)

- Unpooling
  - Vector  $z$
  - Matrix  $U_S = P_S^T$
  - Vector  $x := U_S z$



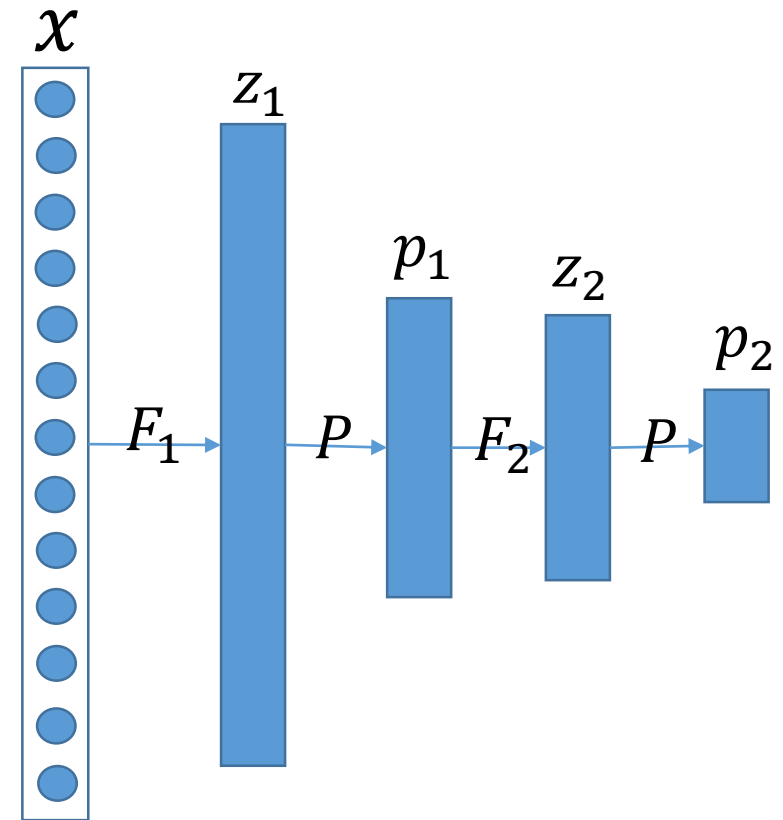
# Adaptive Deconvnet (Zeiler et al., 2011)

- $y$ 
  - an image
- Inference
  - Given model and  $y$ , find  $z_2$
  - Feature extraction



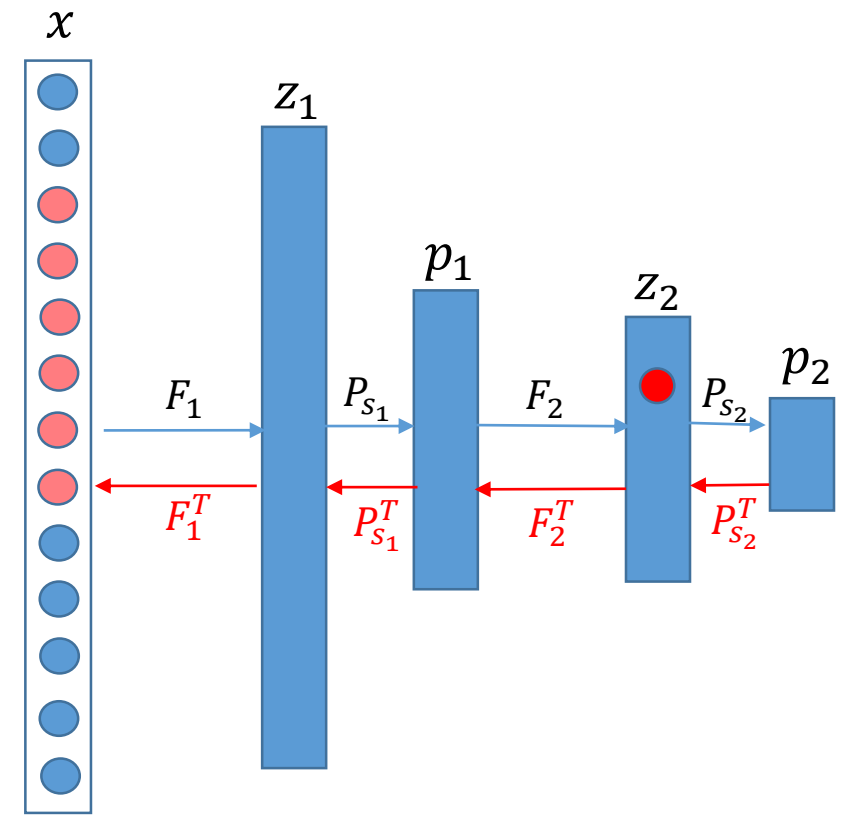
# Back-Project CNN with Deconvnet (Zeiler & Fergus, 2014)

- Let  $menet = (conv1, pool, conv2, pool)$  be part of a classifier
- Let  $x$  be an image
- The inference of  $menet$  on  $x$ 
  - Outputs  $z_i$ 's and  $p_i$ 's
  - Also outputs **switches**



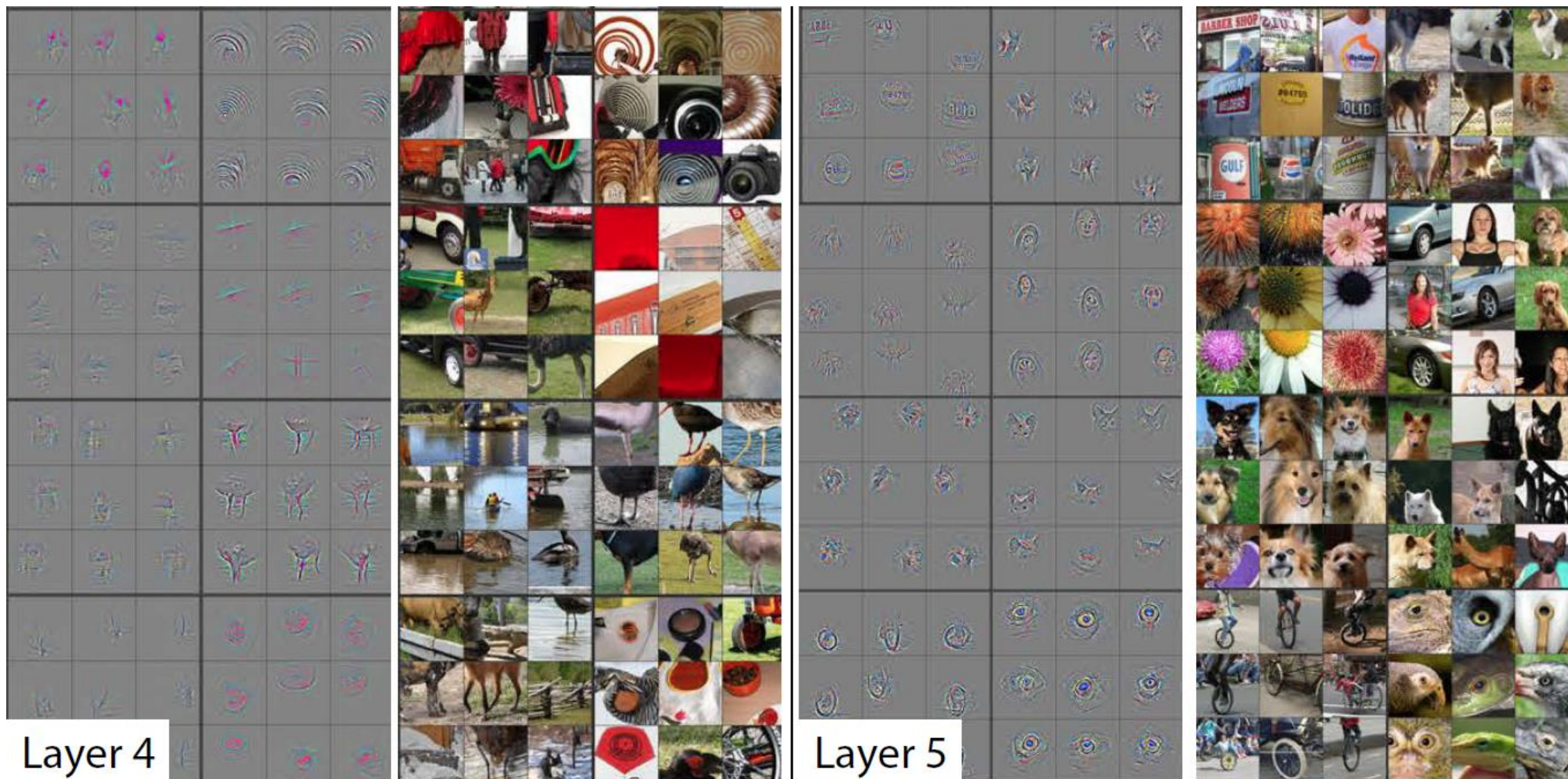
# Back-Project CNN with Deconvnet (Zeiler & Fergus, 2014)

- Attach a *de\_menet* to *menet*
- **Reconstruct** the part of  $x$  which activates
  - $z_i$
  - $z_{ij}$
- Reconstruct the part of the image of all images which has Highest  $z_i$

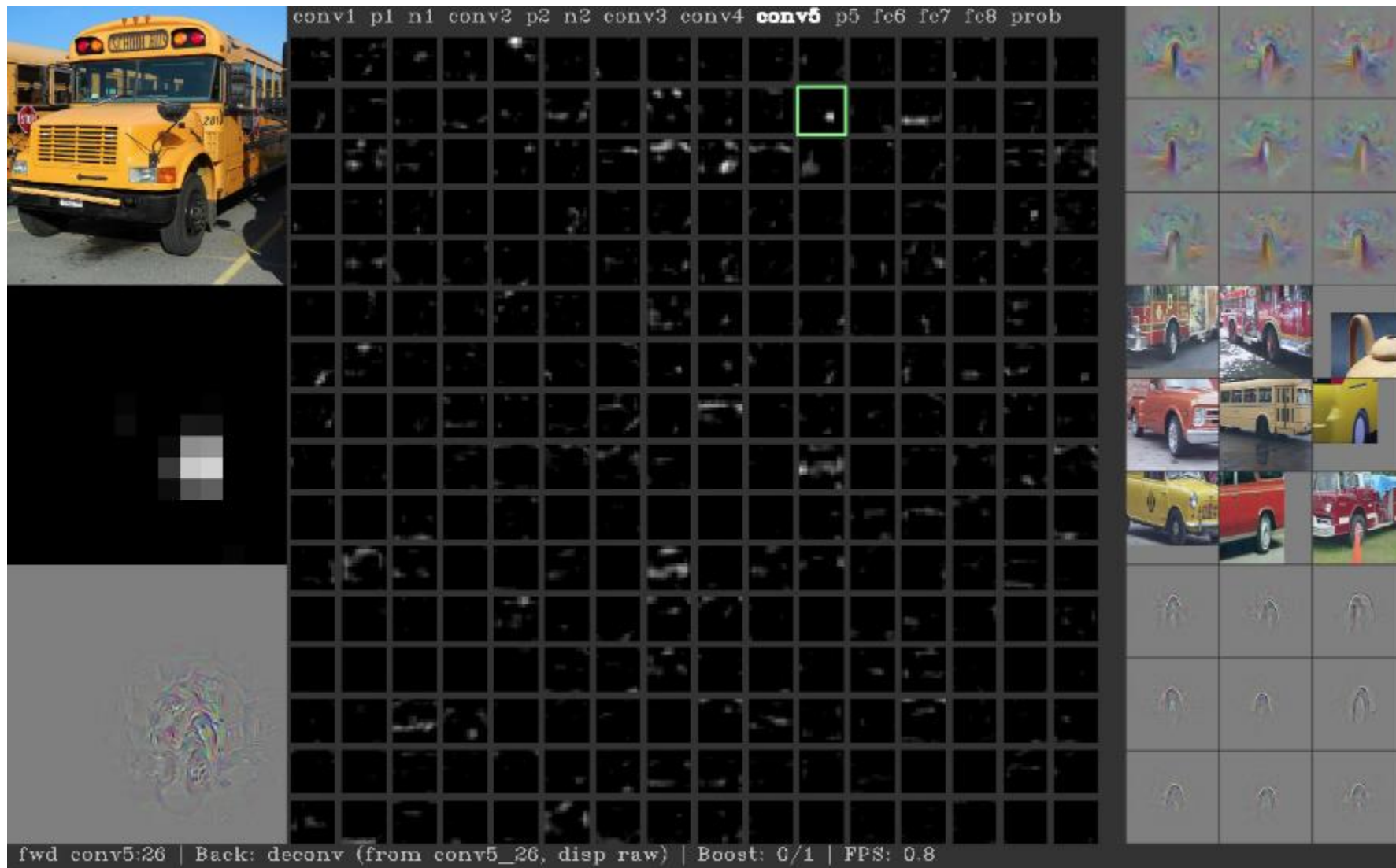




# Back-Project CNN with Deconvnet (Zeiler & Fergus, 2014)



# Deep Visualization Toolbox (Yosinski et al. 2015)



# Rotten Tomatoes

## Raw documents

- schepisi , aided by a cast that seems to include every top-notch british actor who did not appear in gosford park ( as well as one , ms . mirren , who did ) , has succeeded beyond all expectation .
- windtalkers is shapelessly gratifying , the kind of movie that invites you to pick apart its faults even as you have to admit that somehow it hit you where you live .

# Rotten Tomatoes

- A document to an image
  - A word to a 300-dimensional vector
  - Concatenate 60 words to form a  $60 \times 300$  image

# Rotten Tomatoes

- Let *rtnet* be  $(conv1, pool1, conv2, pool2, fc1, fc2, softmax)$
- Train on 8528 documents, validate on 2132
  - Half have *positive* as labels; half has *negative*
- Validation accuracy is 68%

# Rotten Tomatoes

## Highlighted documents

- schepisi , aided [\*\*\*by] a [\*\*cast] that seems to include every top-notch british actor who did not appear in gosford park ( as [\*well] as one , ms . mirren , who did ) , has succeeded beyond all expectation .
- windtalkers is shapelessly [\*\*gratifying] , [\*\*\*the] kind of movie that invites you to pick apart [\*its] faults even as you have to admit that somehow it hit you where you live .