Administrative

- The midterm has been graded!
- Office hours: Andrej instead of Fei-Fei, 4pm, Fei-Fei office
Midterm statistics

Mean: 64
Median: 65
Max grade: 94

top 10%: 84+
top 20%: 78+
top 30%: 73+
top 40%: 69+
top 50%: 65+

top scores descending:
94, 93, 92, 90, 90, 90, 89, 89, 86, 86, 86, 86, 85, 85, 84, 83, 83 ...
Midterm statistics

10 Multiple Choice questions: average grade per question

20 True False questions: average grade per question
Midterm statistics

Correlated T/F questions:

pair, correlation:
(7,8), 0.37
(2,15), 0.37
(2,7), 0.34
(2,11), 0.3
(7,17), 0.24
-----
(17,18), -0.47
(4,19), -0.21
Midterm statistics

Can also do the same thing but swap axes:

**Student Correlation Matrix**

- lowest correlation: -0.7
- highest correlation: 0.95

fun notes:
- noone gave exact same TF answers
- noone got all TF questions correct
Midterm statistics

**BONUS:**
6 people got the Bonus Question
9 people gave admirable effort (>0 & < 3 points awarded)
Lecture 11:

Beyond Image Classification
Where we are...
We’ve introduced (Convolutional) Neural Nets
We’ve seen how to train them
We’ve looked at how they work
And how they are applied in practice
But one thing has remained the same...

(assume given set of discrete labels)
{dog, cat, truck, plane, ...}
Lecture 11:

Beyond Image Classification
Localization

Model must output:

- **class** (integer)
- \(x_1, y_1, x_2, y_2\) **bounding box** coordinates
Idea: train a Localization net
Take out Softmax loss, swap in L2 (regression) loss, **fine-tune** the classification network.
Idea: train a Localization net
Take out Softmax loss, swap in L2 (regression) loss, **fine-tune** the classification network.

Predictions: instead of class scores, now interpreted as the 4 bounding box coords (also 4D vector from net)

Targets: true bounding box 4D vector of \([x_1, y_1, x_2, y_2]\)

\[ L_i = \| f - y_i \|_2^2 \]
In practice:

- It works better to predict a **4D vector for every class** (e.g. 4000D vector for 1000 ImageNet classes). During training only backprop the loss for the correct class.
- apply at **multiple locations and scales**

swap the Softmax layer at the end with L2 loss
OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks, Sermanet et al., 2014

greedy merging procedure
Tip: Note that we can apply ConvNets over all positions in an image very efficiently.

**Normal ConvNet:**

\[
[224 \times 224 \times 3] \rightarrow \ldots \rightarrow [7 \times 7 \times 512] \rightarrow [1 \times 1 \times 4096] \rightarrow \ldots \rightarrow [1 \times 1 \times 1000]
\]

last volume  first FC layer  class scores
Tip: Note that we can apply ConvNets over all positions in an image very efficiently.

Normal ConvNet:

$[224 \times 224 \times 3] \rightarrow \cdots \rightarrow [7 \times 7 \times 512] \rightarrow [1 \times 1 \times 4096] \rightarrow \cdots \rightarrow [1 \times 1 \times 1000]$

- last volume
- first FC layer
- class scores

Convert the first FC layer into a CONV layer:
Note: This FC layer is equivalent to CONV layer with:
- receptive field size of $7 \times 7$, pad 0, stride 1, and 4096 neurons

Convert later FC layers: receptive field sizes $1 \times 1$, pad 0, stride 1
Tip: Note that we can apply ConvNets over all positions in an image very efficiently.

**Normal ConvNet:**

\[
[224x224x3] \rightarrow \ldots \rightarrow [7x7x512] \rightarrow [1x1x4096] \rightarrow \ldots \rightarrow [1x1x1000]
\]

- **last volume**
- **first FC layer**
- **class scores**

**Convert the first FC layer into a CONV layer:**

Note: This FC layer is **equivalent** to CONV layer with:
- receptive field size of 7x7, pad 0, stride 1, and 4096 neurons

**Convert later FC layers:** receptive field sizes 1x1, pad 0, stride 1

**Modified ConvNet:**

\[
[384x384x3] \rightarrow \ldots \rightarrow [12x12x512] \rightarrow [6x6x4096] \rightarrow \ldots \rightarrow [6x6x1000]
\]

- **last volume**
- **7x7 CONV**
- **1x1 CONV**
- **class scores volume!**

This is very efficient, can be seen as evaluating the FC layer in parallel on many locations in the image. This is a common trick. Same trick used in localization.
Detection

Model must output:

A set of detections

Each detection has:
- confidence
- class (integer)
- \(x1, y1, x2, y2\) bounding box coordinates
Idea: Turn a Detection Problem into an Image Classification problem (but over image regions).

Content of every labeled bounding box for is a positive example for a class.

Every other bounding box in the image is a special negative class.
Idea: Turn a Detection Problem into an Image Classification problem (but over image regions).
Selective Search for Object Recognition
[J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, A. W. M. Smeulders]

Gives on average ~2,000 candidate region proposals per image.
(This paradigm currently outperform the “sliding window” approach)
Rich feature hierarchies for accurate object detection and semantic segmentation

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik
Segmentation

Fully Convolutional Networks for Semantic Segmentation
Long, Shelhamer, Darrell
Depth Map Prediction from a Single Image using a Multi-Scale Deep Network

[Eigen et al.], 2014

---

**Layer** | **input** | **1** | **2,3,4** | **5** | **6** | **7** | **1,2,3,4**
---|---|---|---|---|---|---|---
Size (NYUDepth) | 304x228 | 71x20 | 18x13 | 8x6 | 1x1 | 142x27 | 142x27
Size (KITTI) | 576x172 | 71x20 | 35x9 | 17x4 | 1x1 | 142x27 | 142x27
Ratio to input | /1 | /8 | /16 | /32 | – | /4 | /4
Video Classification

Two-Stream Convolutional Networks for Action Recognition in Videos [Simonyan et al., 2014]

Long-term Recurrent Convolutional Networks for Visual Recognition and Description [Donahue et al., 2014]

Large-scale Video Classification with Convolutional Neural Networks [Karpathy et al., 2014]
Image Captioning

- man in black shirt is playing guitar.
- construction worker in orange safety vest is working on road.
- two young girls are playing with lego toy.
- boy is doing backflip on wakeboard.
- girl in pink dress is jumping in air.
- black and white dog jumps over bar.
- young girl in pink shirt is swinging on swing.
- man in blue wetsuit is surfing on wave.
Neural Networks practitioner
Recurrent Neural Network

Convolutional Neural Network
Recurrent Networks are good at modeling sequences...

Generating Sequences With Recurrent Neural Networks
[Alex Graves, 2014]
Recurrent Networks are good at modeling sequences...

Word-level language model. Similar to:

Recurrent Neural Network Based Language Model
[Tomas Mikolov, 2010]
Recurrent Networks are good at modeling sequences...

Machine Translation model

French words — English words

Sequence to Sequence Learning with Neural Networks
[Ilya Sutskever, Oriol Vinyals, Quoc V. Le, 2014]
Suppose we had the training sentence “cat sat on mat”

We want to train a **language model**:

\[
P(\text{next word} \mid \text{previous words})
\]

i.e. want these to be high:

\[
\begin{align*}
P(\text{cat} \mid [\langle S \rangle]) \\
P(\text{sat} \mid [\langle S \rangle, \text{cat}]) \\
P(\text{on} \mid [\langle S \rangle, \text{cat}, \text{sat}]) \\
P(\text{mat} \mid [\langle S \rangle, \text{cat}, \text{sat}, \text{on}]) \\
P(\langle E \rangle \mid [\langle S \rangle, \text{cat}, \text{sat}, \text{on}, \text{mat}])
\end{align*}
\]
Suppose we had the training sentence “cat sat on mat”

We want to train a **language model**: $P(\text{next word} \mid \text{previous words})$

First, suppose we had only a finite, 1-word history: i.e. want these to be high:
- $P(\text{cat} \mid <S>)$
- $P(\text{sat} \mid \text{cat})$
- $P(\text{on} \mid \text{sat})$
- $P(\text{mat} \mid \text{on})$
- $P(<E> \mid \text{mat})$
Vanilla 2-layer classification net for each word given previous word:

```
y0 ----> h0 ----> x0
    ^         ^
     |         |
  y1 ----> h1 ----> x1
     |         |
  y2 ----> h2 ----> x2
     |         |
  y3 ----> h3 ----> x3
     |         |
  y4 ----> h4 ----> x4
```

```
300 (learnable) numbers associated with each word in vocabulary
```

```
10,001-D class scores:
Softmax over 10,000 words and a special <END> token.
y4 = Why * h4
```

```
h4 = max(0, Wxh * x4)
```

```
“cat sat on mat”
```

```
"cat sat on mat"

Turn it into RNN: (#anticlimatic)

10,001-D class scores: Softmax over 10,000 words and a special <END> token.

\[ y_4 = \text{Why} \times h_4 \]

hidden layer (e.g. 500-D vectors)

\[ h_4 = \max(0, W_{xh} \times x_4 + W_{hh} \times h_3) \]

300 (learnable) numbers associated with each word in vocabulary
Training this on a lot of sentences would give us a language model. A way to predict

\[ P(\text{next word} \mid \text{previous words}) \]
Training this on a lot of sentences would give us a language model. A way to predict

\[ P(\text{next word} \mid \text{previous words}) \]
Training this on a lot of sentences would give us a language model. A way to predict

\[ P(\text{next word} \mid \text{previous words}) \]
Training this on a lot of sentences would give us a language model. A way to predict

\[ P(\text{next word} \mid \text{previous words}) \]
Training this on a lot of sentences would give us a language model. A way to predict

\[ P(\text{next word} \mid \text{previous words}) \]
Training this on a lot of sentences would give us a language model. A way to predict

\[ P(\text{next word} \mid \text{previous words}) \]
Training this on a lot of sentences would give us a language model. A way to predict

$$P(\text{next word} \mid \text{previous words})$$
Training this on a lot of sentences would give us a language model. A way to predict

\[ P(\text{next word} \mid \text{previous words}) \]
Training this on a lot of sentences would give us a language model. A way to predict

\[ P(\text{next word} \mid \text{previous words}) \]
Training this on a lot of sentences would give us a language model. A way to predict

\[ P(\text{next word} \mid \text{previous words}) \]
Training this on a lot of sentences would give us a language model. A way to predict

$$P(\text{next word} \mid \text{previous words})$$

samples $<$END$>$? done.
Recurrent Neural Network

Convolutional Neural Network
“straw hat”

training example
"straw hat"

training example
h0 = max(0, Wxh * x0)

now:

h0 = max(0, Wxh * x0 + Wih * v)
test image
test image

sample!
test image
test image

sample!

test image

sample!
Fei-Fei Li & Andrej Karpathy

Lecture 11 - 64
18 Feb 2015

test image
test image

sample!

<END> token

=> finish.
LSTM just changes the form of the equation for $h$ such that:

1. more expressive multiplicative interactions
2. gradients flow nicer
3. network can explicitly decide to reset the hidden state

\[ h_1 = \max(0, W_{xh} \ast x_1 + W_{hh} \ast h_0) \]
RNN vs. LSTM

- Use LSTM over RNN
- Do not be intimidated by pictures that try to draw an LSTM, e.g.:

(it’s just a particular funny form of forward function, backpropagation as usual)

\[
\text{LSTM: } h_{t}^{l-1}, h_{t-1}^{l}, c_{t-1}^{l} \rightarrow h_{t}^{l}, c_{t}^{l}
\]

\[
\begin{pmatrix}
    i \\
    f \\
    o \\
    g
\end{pmatrix} =
\begin{pmatrix}
    \text{sigm} \\
    \text{sigm} \\
    \text{sigm} \\
    \text{tanh}
\end{pmatrix} T_{2n,4n} \begin{pmatrix}
    h_{t}^{l-1} \\
    h_{t}^{l}
\end{pmatrix}
\]

\[
c_{t}^{l} = f \odot c_{t-1}^{l} + i \odot g
\]

\[
h_{t}^{l} = o \odot \text{tanh}(c_{t}^{l})
\]

Recurrent Neural Network Regularization
[Zaremba, Sutskever, Vinyals]
Image Sentence Datasets

Microsoft COCO  
[Tsung-Yi Lin et al. 2014]  
[mscoco.org]

currently:
~120K images  
~5 sentences each
Wow I can’t believe that worked

a group of people standing around a room with remotes
logprob: -9.17

a young boy is holding a baseball bat
logprob: -7.61

a cow is standing in the middle of a street
logprob: -8.84
Well, I can kind of see it
Not sure what happened there...

Ranking and Retrieval

Each example is a query test sentence, the most likely retrieved image & the grounding:

Ranking and Retrieval

Deep Visual-Semantic Alignments for Generating Image Descriptions
[Karpathy and Fei-Fei, 2015]
Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models
[Kiros, Salakhutdinov, Zemel, 2014]
Recurrent Attention Models

Multiple Object Recognition with Visual Attention
[Jimmy Lei Ba, Volodymyr Mnih, Koray Kavukcuoglu], 2014

web demo
http://www.psi.toronto.edu/~jimmy/dram/
also DRAW: https://www.youtube.com/watch?v=Z7MI9eKEo
Summary:

- We looked at many Computer Vision tasks beyond Image Classification and how they are addressed with Convolutional Neural Networks
Next: Guest Lecture:

Evan Shelhamer

Caffe