VISION & LANGUAGE

From Captions to Visual Concepts and Back

Brady Fowler & Kerry Jones

Tuesday, February 28th 2017

CS 6501-004
VICENTE

Agenda

- Problem Domain
- Object Detection
- Language Generation
- Sentence Re-Ranking
- Results & Comparisons

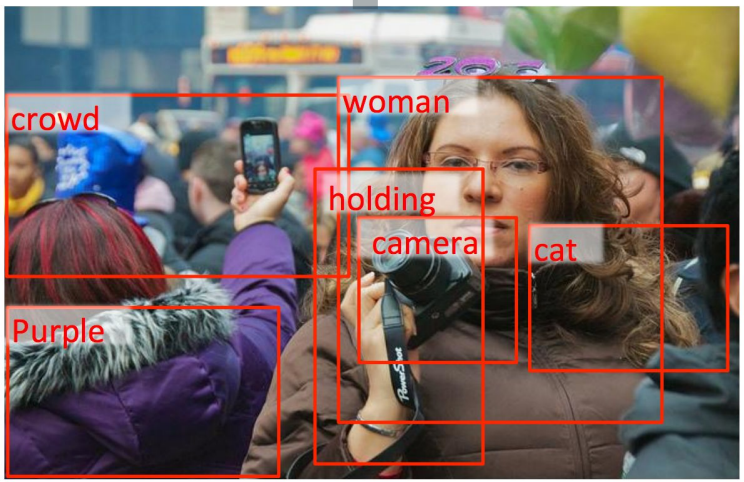
Problem & Goal

- **Goal**: Generate image captions that are on par with human descriptions
- Previous approaches to generating image captions relied on object, attribute, and relation detectors learned from separate hand-labeled training data
  - This implementation seeks to use only images and captions without any human generated features
- Benefit of using captions:
  1. Caption structure inherently reflects object importance
  2. Possible to infer broader concepts (beautiful, flying, open) not directly tied to objects tagged in image.
  3. Learning a joint multimodal representation allows global semantic similarities to be measured for re-ranking

Related Work

- 2 major approaches to automatic image captioning and a few examples:
  - Retrieval of human captions
    - R. Socher et al. used dependency trees to embed sentences into a vector space in order to retrieve images that are described by those sentences
    - Karpathy et al. embedded image fragments (objects) and sentence fragments into common vector space
  - Generation of new captions based on detected objects:
    - Mitchell et al. developed Midge system which integrates word co-occurrence statistics to filter out noise in generation.
    - BabyTalk system which inserts detected words into template slots.

Captioning Pipeline



Detect Words

Woman, Crowd, Cat,
Camera, Holding, Purple

Generate Sequences

A purple camera with a woman.
A woman holding a camera in a crowd.
…
A woman holding a cat.

Re-rank Sequences

A woman holding a camera in a crowd.

OBJECT **DETECTION**

Apply CNN to image regions with **Multiple Instance Learning**

Word Detection Approach

- Input is raw images without bounding boxes
- Output is probability distribution of word vocabulary
  - Vocab = 1,000 most frequent words; 92% of total words
- Instead of using entire image, they use dense scanning of the image:*
  - Each region of the image is converted into features w/ CNN
  - Features are mapped to output vocabulary words with highest probability of being in the caption
    - Using multiple instance learning setup this learns a visual signature for each word

*early version of the system used edge box recommendations

Word Detection Approach

- *"When this fully convolutional network is run over the image, we obtain a coarse spatial response map.*

- *Each location in this response map corresponds to the response obtained by applying the original CNN to overlapping shifted regions of the input image (thereby effectively scanning different locations in the image for possible objects).*

- *We up-sample the image to make the longer side to be 565 pixels which gives us a 12 × 12 response map at fc8 for both [21, 42] and corresponds to sliding a 224×224 bounding box in the up-sampled image with a stride of 32.*

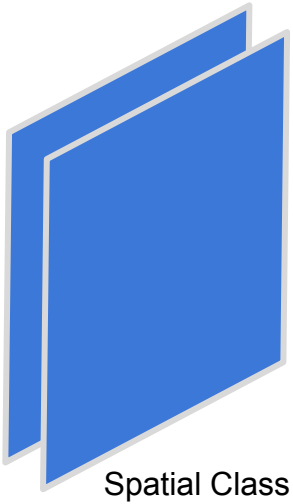
- *The noisy-OR version of MIL is then implemented on top of this response map to generate a single probability* p^w_i *for each word for each image. We use a cross entropy loss and optimize the CNN end-to-end for this task with stochastic gradient descent."*

Word Detection

Image

CNN

FC-8 as fully convolutional layers



Spatial Class Probability Maps

MIL

Multiple Instance Learning

Per Class Probability

$$p^{w}{}_{ij} = \frac{1}{1 + \exp\left(-\left(\mathbf{v}_{\mathbf{w}}^{\mathbf{t}}\phi(b_{ij}) + u_w\right)\right)}$$

Architecture Layout:
Saurabh Gupta

Word Detection

$$p^{w}{}_{ij} = \frac{1}{1 + \exp\left(-\left(\mathbf{v}_{\mathbf{w}}^{\mathbf{t}}\phi(b_{ij}) + u_w\right)\right)}$$

- For a given word:
  - Divide images into “positive” and “negative” bags of bounding boxes (each image = a bag)
  - Pass image through CNN and retrieve response map, $\boldsymbol{\phi}(\mathrm{b}_{\mathrm{ij}})$
    - There are as many $\boldsymbol{\phi}(b_{ij})$ as there are regions (j indicates region)
  - For every $\boldsymbol{\phi}(b_{ij})$ you compute $p^{w}{}_{ij}$ (probability for every word)
  - To calculate the probability of a word being in the image ($b^{w}{}_{i}$) you pass in the probability of that word across all regions into:

$$b^{w}{}_{i} = 1 - \prod_{j \in b_i} \left(1 - p_{ij}^{w}\right)$$

Loss

- After all this we will be left with a vector of word probabilities for the image which we can compare to the ground truth:

  Estimation: [.01, .03, .01, .9, .01, … 0.1, .8, .6, .01 **]**

  Truth: **[** 0, 0, 0, 1, 0, … 0, 1, 1, 0 **]**

  crowd *woman* *camera*

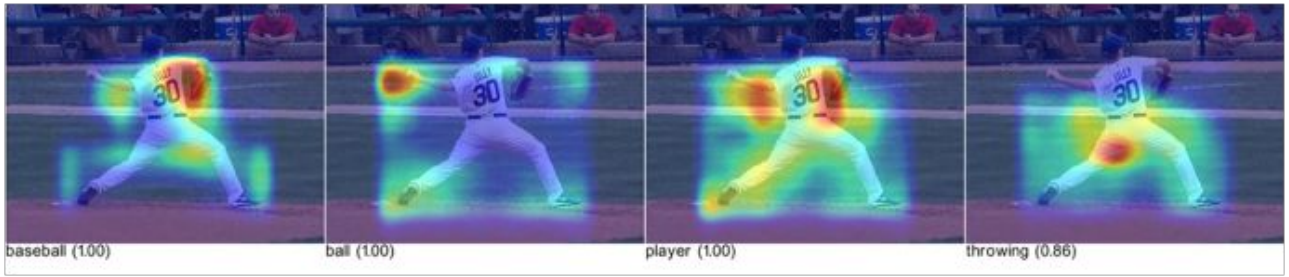
- Use cross entropy loss to optimize the CNN end-to-end as well as the V_w and U_w weights used in calculating by-region word probability, $p^w{}_{ij}$

- Once trained, a global threshold, **T**, is selected to pick the top words with probability $p^w{}_i$ above the threshold

Word Probability Maps

a man sitting on a chair with a dog in his lap

a pitcher holds his arm far behind him during a pitch



Word Detection Results

	Average Precision								Precision at Human Recall							
	NN	VB	JJ	DT	PRP	IN	Others	All	NN	VB	JJ	DT	PRP	IN	Others	All
Count	616	176	119	10	11	38	30	1000								
Chance	2.0	2.3	2.5	23.6	4.7	11.9	7.7	2.9								
Classification (AlexNet)	32.4	16.7	20.7	31.6	16.8	21.4	15.6	27.1	39.0	27.7	37.0	37.3	26.2	31.5	25.0	35.9
Classification (VGG)	37.0	19.4	22.5	32.9	19.4	22.5	16.9	30.8	45.3	31.0	37.1	40.2	29.6	33.9	25.5	40.6
MIL (AlexNet)	36.9	18.0	22.9	31.7	16.8	21.4	15.2	30.4	46.0	29.4	40.1	37.9	25.9	31.5	21.6	40.8
MIL (VGG)	41.4	20.7	24.9	32.4	19.1	22.8	16.3	34.0	51.6	33.3	44.3	39.2	29.4	34.3	23.9	45.7
Human Agreement									63.8	35.0	35.9	43.1	32.5	34.3	31.6	52.8

Biggest improvement from MIL are concrete objects

Language **Generation**
& Sentence **Re-Ranking**

Language Generation

Maximum Entropy Language Model:

- Generates novel image descriptions from a bag of likely words.
- Trained on 400,000 Image Descriptions
- A search over word sequence is used to find high-likelihood sentences

Sentence Re-ranking:

- Re-ranks set of sentences by a linear weight of the sentences features.
- Trained using Minimum Error Rate Training(MERT)
- Deep Multimodal Similarity Model Feature

Maximum Entropy LM

- Using maximum entropy LM conditioned on words chosen in previous step and only uses each word once

$$\Pr(w_l = \bar{w}_l | \bar{w}_{l-1}, \cdots, \bar{w}_1, <s>, \tilde{\mathcal{V}}_{l-1}) = \frac{\exp\left[\sum_{k=1}^{K} \lambda_k f_k(\bar{w}_l, \bar{w}_{l-1}, \cdots, \bar{w}_1, <s>, \tilde{\mathcal{V}}_{l-1})\right]}{\sum_{v \in \mathcal{V} \cup </s>} \exp\left[\sum_{k=1}^{K} \lambda_k f_k(v, \bar{w}_{l-1}, \cdots, \bar{w}_1, <s>, \tilde{\mathcal{V}}_{l-1})\right]}$$

- To train the model, the objective function is the log-likelihood of captions conditioned on the corresponding set of objects

$$L(\Lambda) = \sum_{s=1}^{S} \sum_{l=1}^{\#(s)} \log \Pr(\bar{w}_l^{(s)} | \bar{w}_{l-1}^{(s)}, \cdots, \bar{w}_1^{(s)}, <s>, \tilde{\mathcal{V}}_{l-1}^{(s)})$$

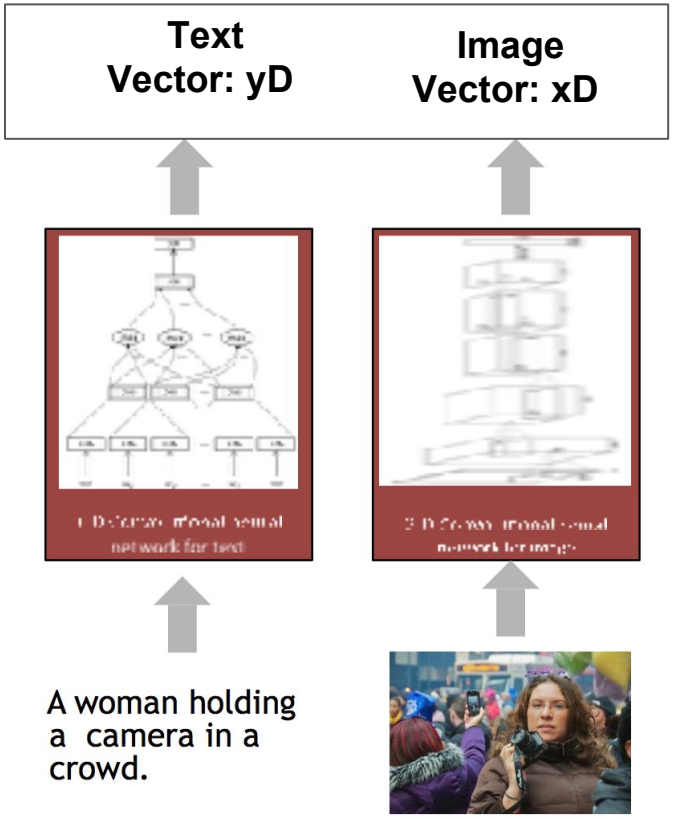
- Sentences are generated using **Beam Process**

Sentence Re-Ranking

- **MERT used to rank sentence likelihood**
  - Uses linear combination of features over whole sentence.
    - Log-likelihood of the sequence
    - Length of the sequence
    - The log-probability per word of the sequence
    - The logarithm of the sequences rank in the log-likelihood
    - 11 binary features indicating whether number of objects were mentioned
    - DMSM Score between word sequence and the Image
- Deep Multimodal Similarity Model(DMSM) is a feature of MERT that measures similarity between images and text.

Deep Multimodal Similarity Model(DMSM)

- DMSM is used to improve the quality of the sentences.
- Trains two neural networks jointly that map images and text fragments to a common vector representation



Deep Multimodal Similarity Model(DMSM)

Relevance(R) = cosine(Text, Image)

For every text-image pair, we compute:

$$P(D|Q) = \frac{\exp(\gamma R(Q, D))}{\Sigma_{D' \in \mathbb{D}} \exp(\gamma R(Q, D'))}$$

The loss function:

$$L(\Lambda) = -\log \prod_{(Q, D^+)} P(D^+|Q)$$

Results

System	PPLX	BLEU	METEOR	≈human	>human	≥human
1. Unconditioned	24.1	1.2%	6.8%			
2. Shuffled Human	–	1.7%	7.3%			
3. Baseline	20.9	16.9%	18.9%	9.9% (±1.5%)	2.4% (±0.8%)	12.3% (±1.6%)
4. Baseline+Score	20.2	20.1%	20.5%	16.9% (±2.0%)	3.9% (±1.0%)	20.8% (±2.2%)
5. Baseline+Score+DMSM	20.2	21.1%	20.7%	18.7% (±2.1%)	4.6% (±1.1%)	23.3% (±2.3%)
6. Baseline+Score+DMSM+ft	19.2	23.3%	22.2%	–	–	–
7. VGG+Score+ft	18.1	23.6%	22.8%	–	–	–
8. VGG+Score+DMSM+ft	18.1	25.7%	23.6%	26.2% (±2.1%)	7.8% (±1.3%)	**34.0%** (±2.5%)
Human-written captions	–	19.3%	24.1%			

Questions?