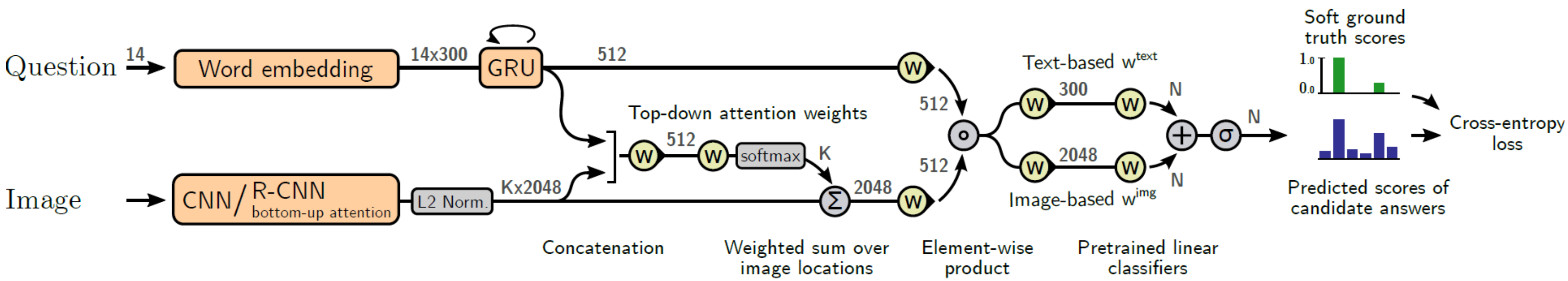
A Simple VQA Model with a Few Tricks and Image Features from Bottom-up Attention

Damien Teney[1], Peter Anderson[2*], David Golub[4*], Po-Sen Huang[3], Lei Zhang[3], Xiaodong He[3], Anton van den Hengel[1]

[1]*University of Adelaide* [2]*Australian National University*
[3]*Microsoft Research* [4]*Stanford University*
**Work performed while interning at MSR*

Proposed model



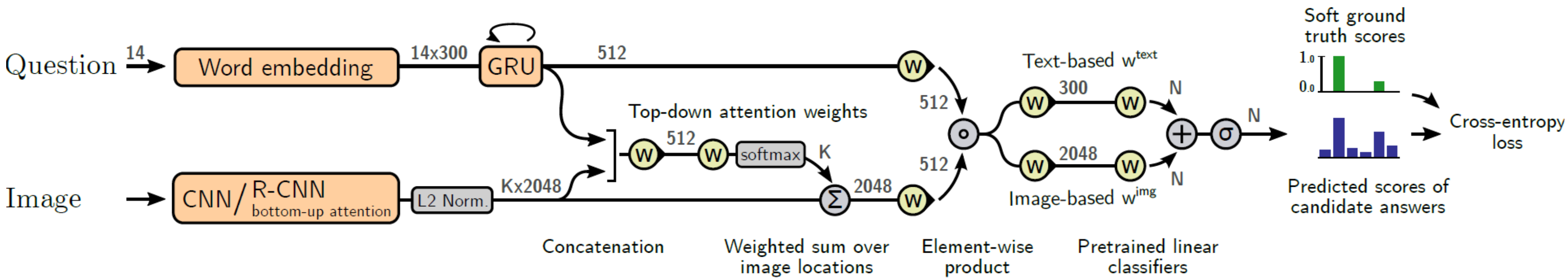
Straightforward architecture

- Joint embedding of question/image
- Single-head, question-guided attention over image
- Element-wise product

The devil is in the details

- Image features from Faster R-CNN
- Gated tanh activations
- Output as regression of answer scores, soft scores as target
- Output classifiers initialized with pretrained representations of answers

Gated layers



Non-linear layers: gated hyperbolic tangent activations

- Defined as: input x, output y

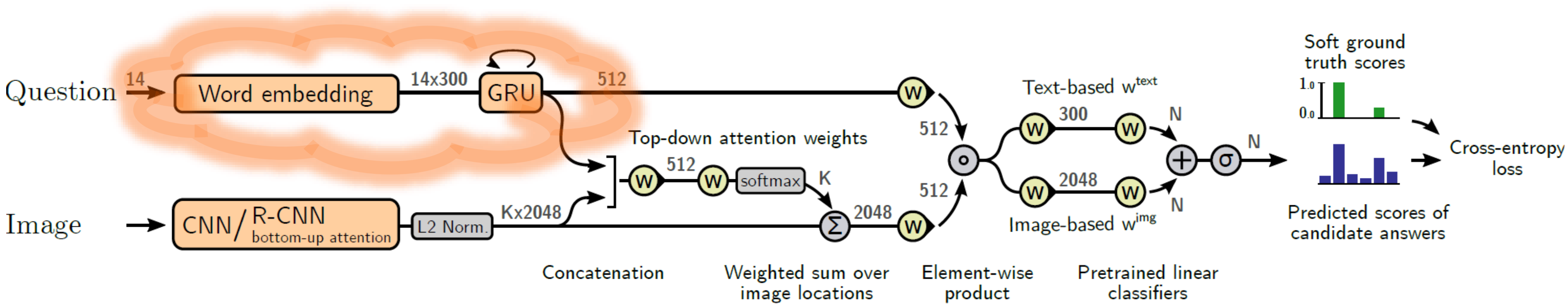
$\tilde{y} = tanh(Wx + b)$	intermediate activation
$g = \sigma(W'x + b')$	gate
$y = \tilde{y} \circ g$	combine with element-wise product

- Inspired by gating in LSTMs/GRUs
- Empirically better than ReLU, tanh, gated ReLU, residual connections, etc.
- Special case of highway networks; used before in:

[1] Dauphin et al. Language modeling with gated convolutional networks, 2016.

[2] Teney et al. Graph-structured representations for visual question answering, 2017.

Question encoding



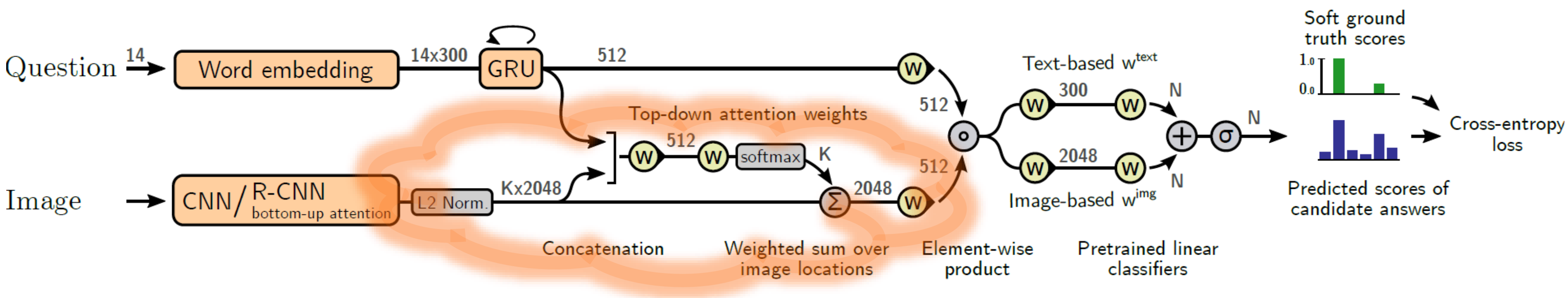
Chosen implementation

- Pretrained GloVe embeddings, d=300
- GRU encoder

Better than....

- Word embeddings learned from scratch
- GloVe of dimension 100, 200
- Bag-of-words (sum/average of embeddings)
- GRU backwards
- GRU bidirectional
- 2-layer GRU

Classical "top-down" attention on image features



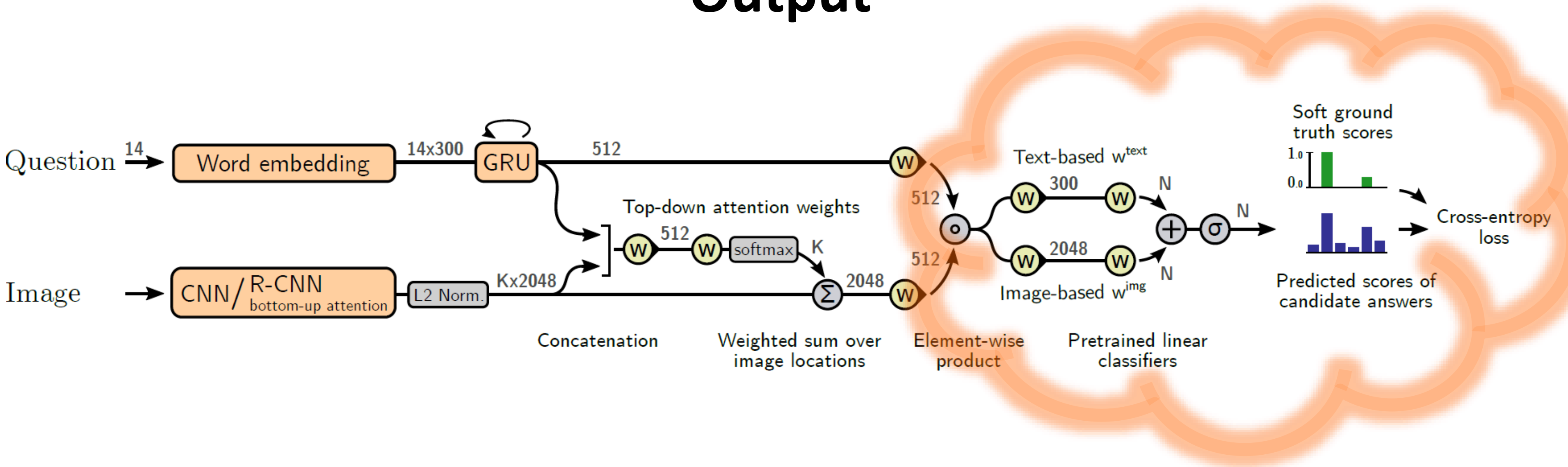
Chosen implementation

- Simple attention on image feature maps
- One head
- Softmax normalization of weights

Better than....

- No L2 normalization
- Multiple heads
- Sigmoid on weights

Output



Chosen implementation

- **Sigmoid** output (regression) of answer scores:

  allows multiple answers per question

- Soft targets in [0,1]

  allows uncertain answers

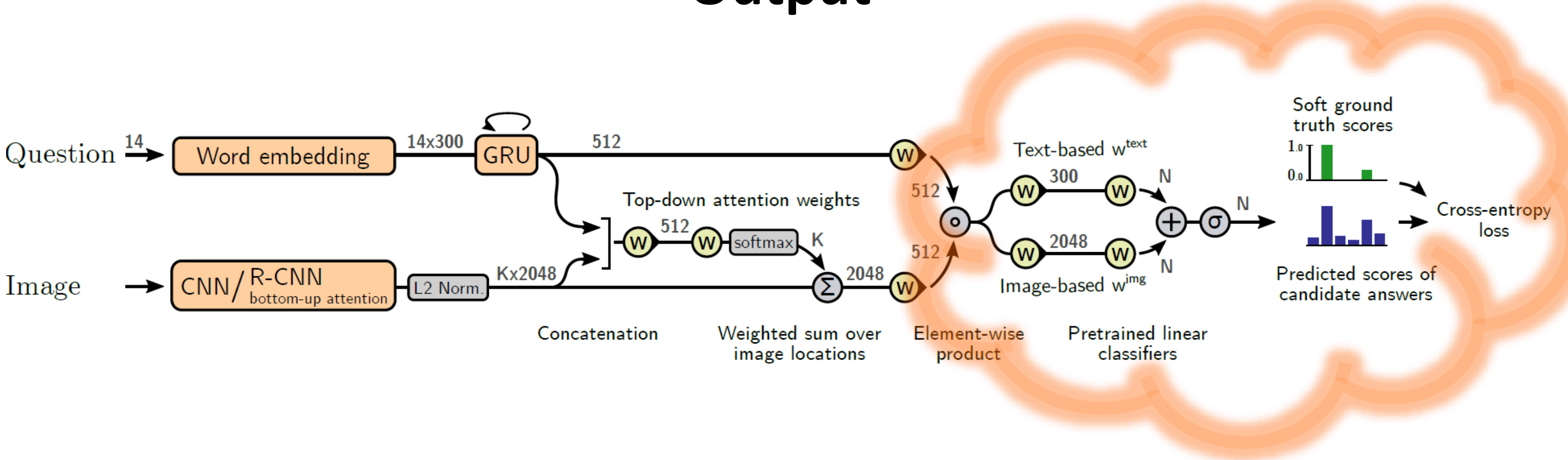
- Initialize classifiers with representations of answers

  $y = \sigma(Wx)$ $\quad W$ of dimensions $nAnswers$ x d

Better than....

- Softmax classifier
- Binary targets {0,1}
- Classifiers learned from scratch

Output



Chosen implementation

- Sigmoid output (regression) of answer scores:

  allows multiple answers per question

- Soft targets in [0,1]

  allows uncertain answers

- Initialize classifiers with representations of answers

$y = \sigma(W^{\text{text}} x^{\text{text}} + W^{\text{img}} x^{\text{img}})$ Initialize W^{text} with GloVe word embeddings

Initialize W^{img} with Google Images (global ResNet features)

Training and implementation

- Additional training data from Visual Genome: questions with matching answers and matching images (about 30% of Visual Genome, *i.e.* ~485,000 questions)
- Keep all questions, even those with no answer in candidates, and with 0<score<1

- Shuffle training data but keep balanced pairs in same mini-batches
- Large mini-batches of 512 QAs; sweet spot in {64, 128, 256, 384, 512, 768, 1024}
- 30-Network ensemble: different random seeds, sum predicted scores

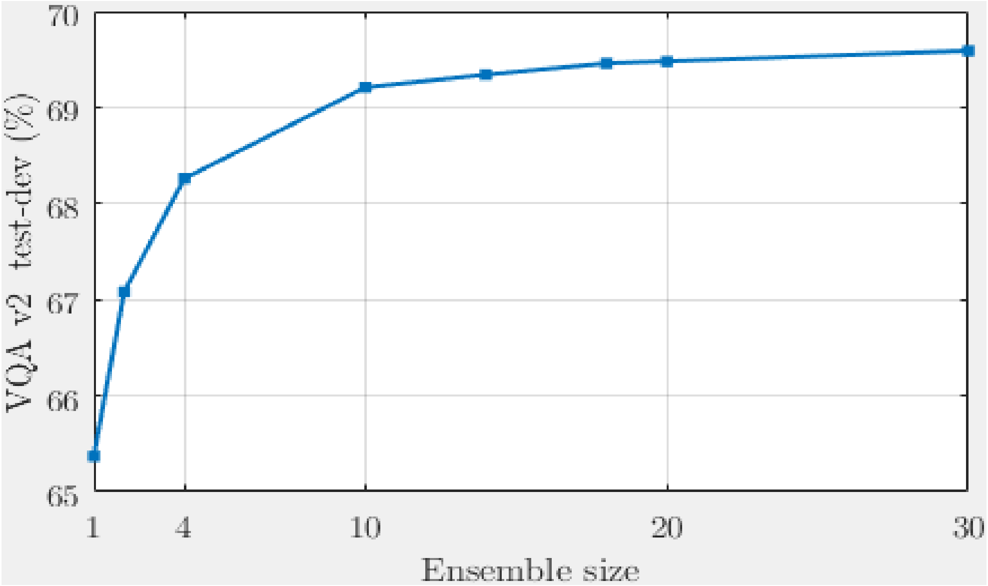
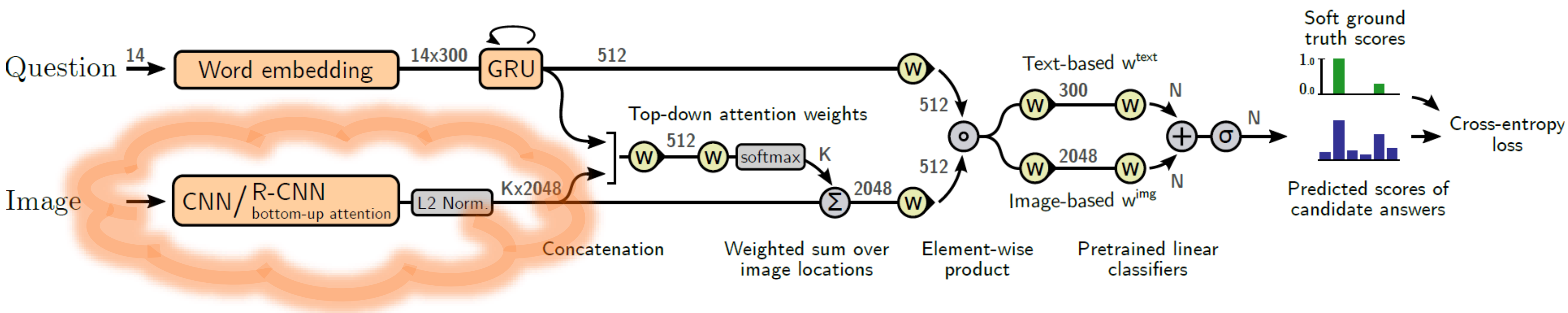
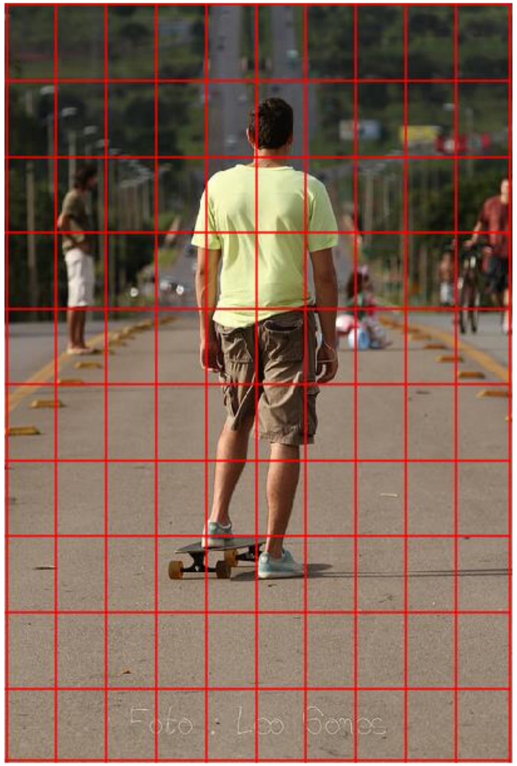


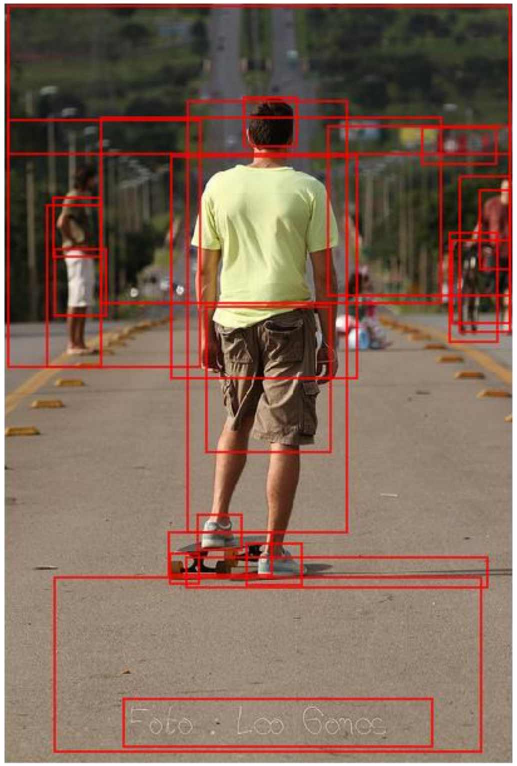
Image features from bottom-up attention



- Equally applicable to VQA and image captioning
- Significant relative improvements: 6 – 8 % (VQA / CIDEr / SPICE)
- Intuitive and interpretable (natural approach)

Bottom-up image attention

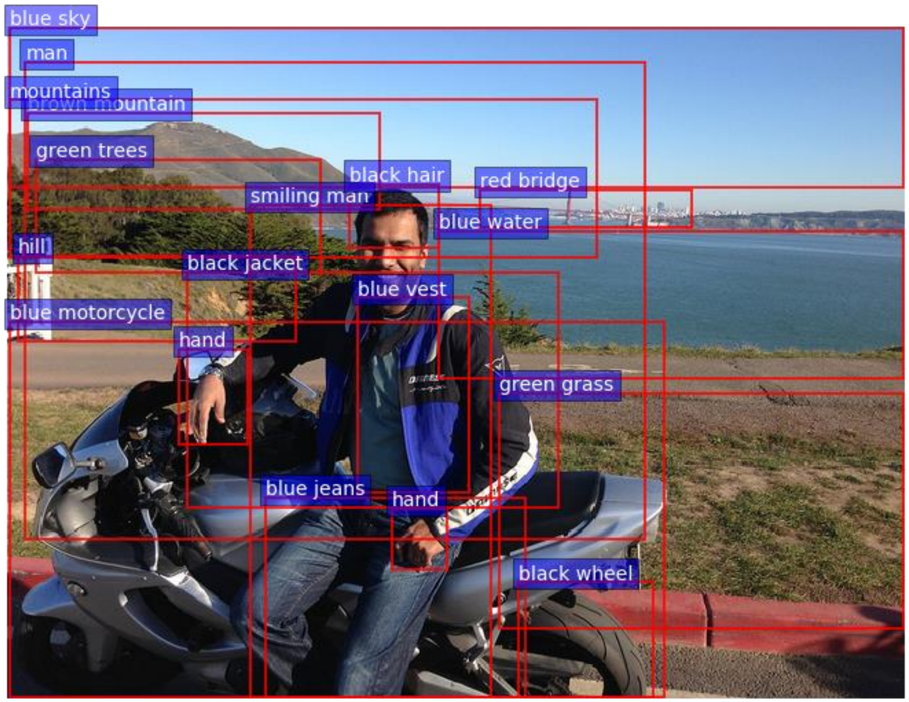


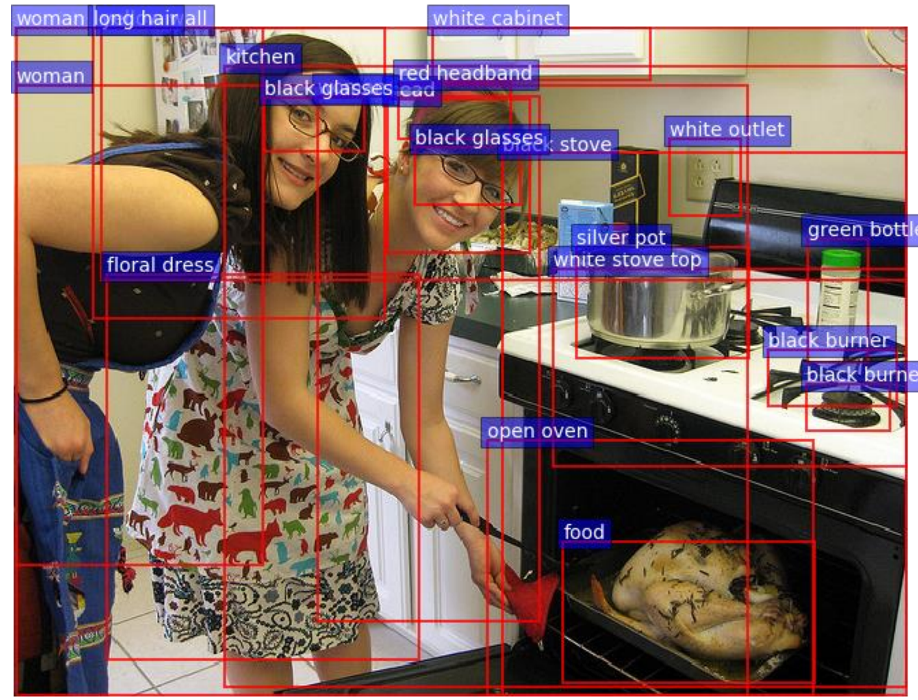


Typically, attention models operate on the spatial output of a CNN

We calculate attention at the level of objects and other salient image regions

Can be implemented with Faster R-CNN[1]





- Pre-train on 1600 objects and 400 attributes from Visual Genome[2]
- Select salient regions based on object detection confidence scores
- Take the mean-pooled ResNet-101[3] feature from each region

[1]NIPS 2015, [2]http://visualgenome.org, [3]CVPR 2016

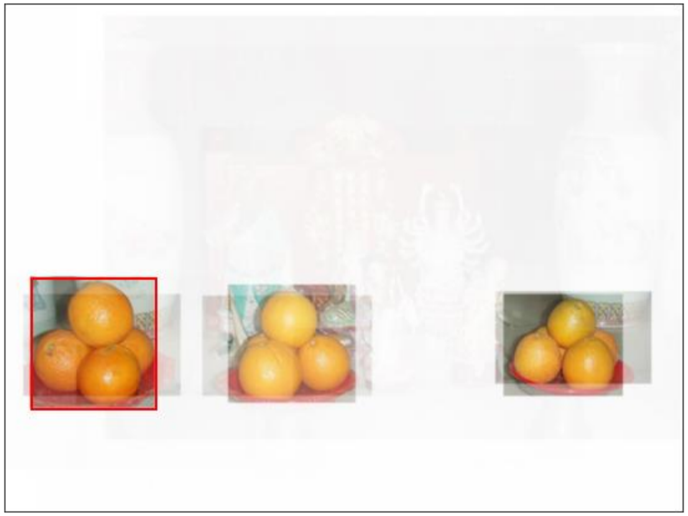
Qualitative differences in attention methods

Q: Is the person wearing a **helmet**?

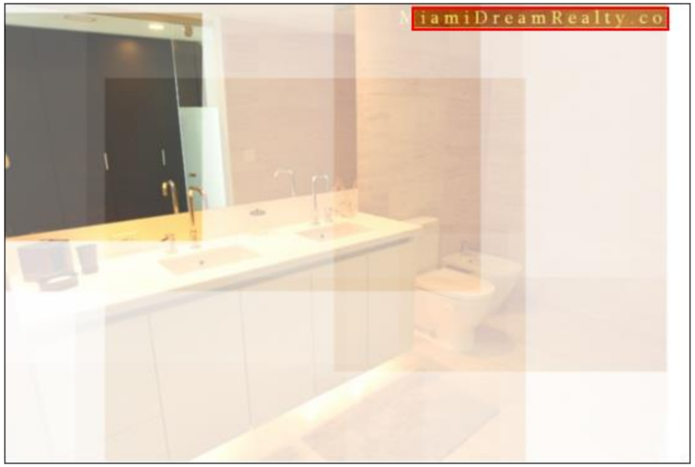
Q: What **foot** is in front of the other **foot**?

VQA failure cases: counting, reading

Q: **How many** oranges are sitting on pedestals?



Q: What is the **name** of the realtor?



Equally applicable to Image Captioning

ResNet baseline: A man sitting on **a** ~~toilet~~ in a bathroom.

Up-Down attention: A man sitting on a **couch** in a bathroom.

MS COCO Image Captioning Leaderboard

Results

#	User	BLEU-1		BLEU-2		BLEU-3		BLEU-4		METEOR		ROUGE-L		CIDEr-D	
		c5 ▲	c40 ▲	c5 ▲	c40 ▲	c5 ▲	c40 ▲	c5 ▲	c40 ▲	c5 ▲	c40 ▲	c5 ▲	c40 ▲	c5 ▲	c40 ▲
1	panderson_msr	0.802 (1)	0.952 (1)	0.641 (1)	0.888 (1)	0.491 (1)	0.794 (1)	0.369 (1)	0.685 (1)	0.276 (1)	0.367 (2)	0.571 (4)	0.724 (1)	1.179 (1)	1.205 (2)
2	TencentAI.v1	0.793 (2)	0.943 (2)	0.633 (2)	0.880 (2)	0.483 (4)	0.781 (2)	0.363 (4)	0.669 (3)	0.275 (2)	0.363 (5)	0.572 (3)	0.719 (3)	1.178 (2)	1.207 (1)
3	xxzhu	0.786 (5)	0.935 (4)	0.629 (3)	0.871 (3)	0.485 (2)	0.778 (3)	0.368 (2)	0.670 (2)	0.275 (3)	0.364 (4)	0.572 (1)	0.721 (2)	1.173 (3)	1.194 (3)
4	CASIA_IVA	0.786 (4)	0.934 (5)	0.629 (4)	0.870 (4)	0.484 (3)	0.776 (4)	0.368 (3)	0.669 (4)	0.274 (4)	0.362 (7)	0.572 (2)	0.719 (4)	1.170 (4)	1.188 (4)
5	Anonymous	0.787 (3)	0.937 (3)	0.627 (5)	0.867 (5)	0.476 (5)	0.765 (5)	0.356 (5)	0.652 (5)	0.270 (6)	0.354 (14)	0.564 (5)	0.705 (10)	1.160 (5)	1.180 (5)
6	etiennem	0.781 (6)	0.931 (6)	0.619 (6)	0.860 (6)	0.470 (6)	0.759 (6)	0.352 (6)	0.645 (7)	0.270 (5)	0.355 (13)	0.563 (6)	0.707 (8)	1.147 (6)	1.167 (6)

- Bottom-up attention adds 6 – 8% improvement on SPICE and CIDEr metrics

  (see arXiv: [Bottom-Up and Top-Down Attention for Image Captioning and VQA]())

- First place on almost all MS COCO leaderboard metrics

VQA experiments

- Current best results Ensemble, trained on tr+va+VG, eval. on test-std
  Yes/no: 86.52 Number: 48.48 Other: 60.95 **Overall: 70.19**

- Bottom-up attention adds 6% relative improvement
  (even though the baseline ResNet has twice as many layers)

Single-network, trained on tr+VG, eval. on va

	Yes/No	Number	Other	Overall
Ours: ResNet (1×1)	76.0	36.5	46.8	56.3
Ours: ResNet (14×14)	76.6	36.2	49.5	57.9
Ours: ResNet (7×7)	77.6	37.7	51.5	59.4
Ours: Up-Down	**80.3**	**42.8**	**55.8**	**63.2**
Relative Improvement	3%	14%	8%	6%

Take-aways and conclusions

- Difficult to predict effects of architecture, hyperparameters, …
  - Engineering effort: good intuitions are valuable, then need fast experiments
  - Performance ≈ (# Ideas) * (# GPUs) / (Training time)
- Beware of experiments with reduced training data
- Non-cumulative gains, performance saturates
  - Fancy tweaks may just add more capacity to network
  - May be redundant with other improvements
- Calculating attention at the level of objects and other salient image regions (bottom-up attention) significantly improves performance
  - Replace pretrained CNN features with pretrained bottom-up attention features

Questions ?

arXiv:1708.02711: **Tips and Tricks for Visual Question Answering: Learnings from the 2017 Challenge**

arXiv:1707.07998: **Bottom-Up and Top-Down Attention for Image Captioning and VQA**

Damien Teney, Peter Anderson, David Golub, Po-Sen Huang,
Lei Zhang, Xiaodong He, Anton van den Hengel