Image Captioning

A survey of recent deep-learning approaches

Kiran Vodrahalli
February 23, 2015

The task

- We want to automatically describe images with words
- Why?
  - 1) It's cool
  - 2) Useful for tech companies (i.e. image search; tell stories from album uploads, help visually impared people understand the web)
  - 3) supposedly requires a detailed understanding of an image and an ability to communicate that information via natural language.

Another Interpretation

- Think of Image Captioning as a Machine Translation problem
- Source: pixels; Target: English
- Many MT methods are adapted to this problem, including scoring approaches (i.e. BLEU)

Recent Work

- Oriol Vinyals' classification of image captioning systems:
- End-to-end vs. pipeline
- Generative vs. retrieval
- Main players:
  - Google, Stanford, Microsoft, Berkeley, CMU, UToronto, Baidu, UCLA
- We'll restrict this talk to summarizing/categorizing techniques and then speaking a bit to more comparable evaluation metrics

End-to-end vs. Pipeline

- Pipeline: separate learning the language model from the visual detectors (Microsoft paper, UToronto)
- End-to-end (Show and Tell Google paper):
  - Solution encapsulated in one neural net
  - Fully trainable using SGD
  - Subnetworks combine language and vision models
  - Typically, neural net used is combination of recurrent and convolutional

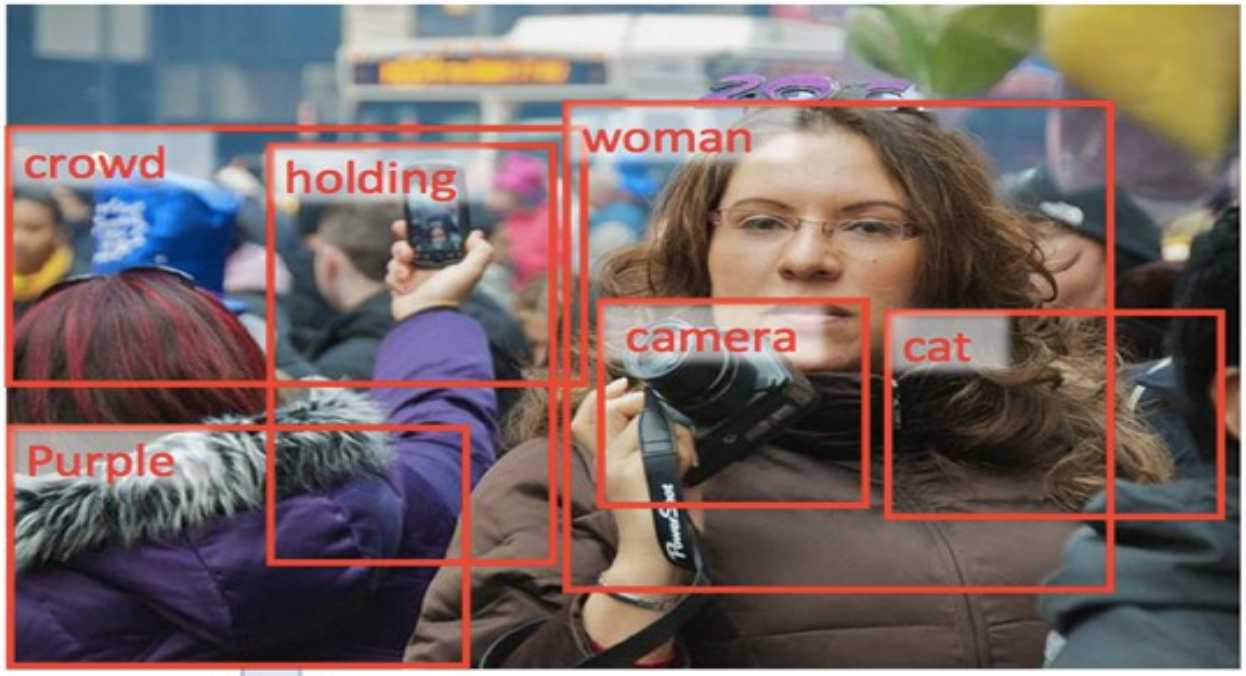
Generative vs. Retrieval

- Generative: generate the captions
- Retrieval: pick the best among a certain restricted set
- Modern papers typically apply generative approach
  - Advantages: caption does not have to be previously seen
  - More intelligent
  - Requires better language model

Representative Papers

- Microsoft paper: generative pipeline, CNN + fully-connected feedforward
- Show and Tell: generative end-to-end
- DRNNs: Show and Tell, CMU, videos → natural language
  - LSTM (most people), RNN, RNNLM (Mikolov); BRNN (Stanford – Karpathy and Fei-Fei)
  - Tend to be end-to-end
- Sometimes called other things (LRCN -Berkeley), but typically combination of RNN for language and CNN for vision

From Captions to Visual Concepts (Microsoft)



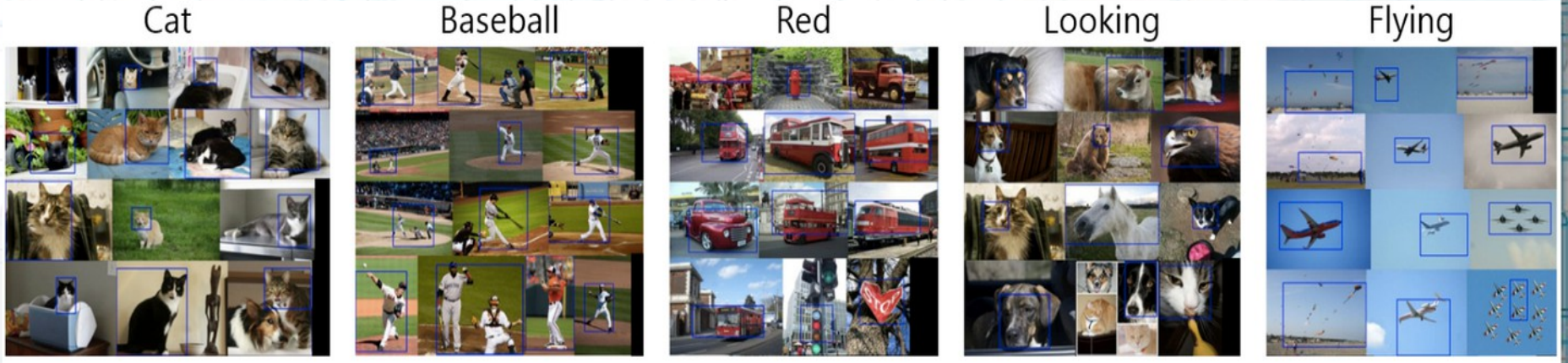
1. detect words → woman, crowd, cat, camera, holding, purple

2. generate sentences → A purple camera with a woman. A woman holding a camera in a crowd. ... A woman holding a cat.

3. re-rank sentences → #1 A woman holding a camera in a crowd.

From Captions to Visual Concepts (Microsoft) (2)

- 1) Detect words: edge-based detection of potential objects in the image (Edge Boxes 70), apply fc6 layer from convolutional net trained on ImageNet to generate high-level feature for each potential object
  - Noisy-OR version of Multiple Instance Learning to figure out which region best matches each word



Multiple Instance Learning

- Common technique
- Set of bags, each containing many instances of a word (bags here are images)
- Labeled negative if none of the objects correspond to a word
- Labeled positive if at least one object corresponds to a word
- Noisy-Or: box j, image i, word w, box feature (fc6) Φ, probability p_{ij}^w

$$1 - \prod_{j \in b_i} \left(1 - p_{ij}^w\right) \qquad \frac{1}{1 + \exp\left(-v_w \phi(b_{ij}) - u_w\right)},$$

From Captions to Visual Concepts (Microsoft) (3)

- 2) Language Generation: Defines probability distribution over captions
- Basic Maximum Entropy Language Model
  - Condition on previous words seen AND
  - {words associated w/image not yet used}
  - Objective function: standard log likelihood
  - Simplification: use Noise Contrastive Elimination to accelerate training
- To generate: Beam-Search

Max Entropy LM

$$\Pr(w_l = \bar{w}_l | \bar{w}_{l-1}, \cdots, \bar{w}_1, <s>, \tilde{\mathcal{V}}_{l-1}) = \frac{\exp\left[\sum_{k=1}^{K} \lambda_k f_k(\bar{w}_l, \bar{w}_{l-1}, \cdots, \bar{w}_1, <s>, \tilde{\mathcal{V}}_{l-1})\right]}{\sum_{v \in \mathcal{V} \cup </s>} \exp\left[\sum_{k=1}^{K} \lambda_k f_k(v, \bar{w}_{l-1}, \cdots, \bar{w}_1, <s>, \tilde{\mathcal{V}}_{l-1})\right]} \quad (3)$$

where $<s>$ denotes the start-of-sentence token, $\bar{w}_j \in \mathcal{V} \cup </s>$, and $f_k(w_l, \cdots, w_1, \tilde{\mathcal{V}}_{l-1})$ and λ_k respectively denote the k-th max-entropy feature and its weight. The ba-

s is index of sentence, #(s) is length of sentence

$$L(\Lambda) = \sum_{s=1}^{S} \sum_{l=1}^{\#(s)} \log \Pr(\bar{w}_l^{(s)} | \bar{w}_{l-1}^{(s)}, \cdots, \bar{w}_1^{(s)}, <s>, \tilde{\mathcal{V}}_{l-1}^{(s)}) \quad (4)$$

Re-rank Sentences

- Language model produces list of M-best sentences
- Uses MERT to re-rank (log-linear stat MT)
  - Uses linear combination of features over whole sentence
  -
  -
  -
  -
  - Not redundant: can't use sentence length as prior in the generation step
  - Trained with BLEU scores
  - DMSM: Deep Multimodal Similarity

Table 2. Features used by MERT.

1. The log-likelihood of the sequence.
2. The length of the sequence.
3. The log-probability per word of the sequence.
4. The logarithm of the sequence's rank in the log-likelihood.
5. 11 binary features indicating whether the number of mentioned objects is x ($x = 0, \ldots, 10$).
6. The DMSM score between the sequence and the image.

Deep Multi-modal Similarity

- 2 neural networks that map images and text fragments to common vector representation; trained jointly
- Measure similarity between images and text with cosine distance
- Image: Deep convolutional net
  - Initialize first 7 layers with pre-trained weights, and learn 5 fully-connected layers on top of those
  - 5 was chosen through cross-validation

DMSM (2)

- Text Model: Deep fully connected network (5 layers)
- Text fragments → semantic vectors

  instead of fixed size word count vector, input is fixed size letter-trigram count vector → reduces size of input layer
- Generalizes to unseen/infrequent and mispelled words
- Bag-of-words esque

DMSM (3)

- Trained jointly; mini-batch grad descent
- Q = image, D = document, R = relevance
- Loss function = negative log posterior probability of seeing caption given image
- Negative sampling approach (1 positive document D+, N negative documents D-)

$$R(Q, D) = \text{cosine}(y_Q, y_D) = \frac{{y_Q}^T y_D}{\|y_Q\| \|y_D\|} \quad (5)$$

For a given image-text pair, we can compute the posterior probability of the text being relevant to the image via:

$$P(D|Q) = \frac{\exp(\gamma R(Q, D))}{\Sigma_{D' \in \mathbb{D}} \exp(\gamma R(Q, D'))}. \quad (6)$$

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

Results summary

- Used COCO (82000 training, 40000 validation), 5 human-annotated captions/ image; validation split into validation and test
- Metrics for measuring image captioning:
  - Perplexity: ~ how many bits on average required to encode each word in LM
  - BLEU: fraction of n-grams (n = 1 → 4) in common btwn hypothesis and set of references
  - METEOR: unigram precision and recall
    - Word matches include similar words (use WordNet)

Results (2)

- Their BLEU score

To understand the highest possible BLEU score we could attain, we tested one human-written caption (as a hypothetical "system") vs. four others. I'm happy to report that, in terms of BLEU score, we actually beat humans! Our system achieved 21.05% BLEU score, while the human "system" scored 19.32%.

Now, you should take this superhuman BLEU score with a gigantic boulder of salt. BLEU has many limitations that are well-known in the machine translation community. We also tried testing with the METEOR metric, and got somewhat below human performance (20.71% vs 24.07%).

- Piotr Dollár: "Well BLEU still sucks"
- METEOR is better, new evaluation metric: CIDEr
- Note: comparison problem w/results from various papers due to BLEU

Show and Tell

- Deep Recurrent Architecture (LSTM)
- Maximize likelihood of target description given image
- Generative model
- Flickr30k dataset: BLEU: 55 → 66
- End-to-end system

Show and Tell (cont.)

- Idea from MT: encoder RNN and decoder RNN (Sequential MT paper)
- Replace encoder RNN with deep CNN
- Fully trainable network with SGD
- Sub-networks for language and vision
- Others use feedforward net to predict next word given image and prev. words; some use simple RNN
- Difference: direct visual input + LSTM
- Others separate the inputs and define joint-embeddings for images and words, unlike this model

Show and Tell (cont.)

- Standard objective: maximize probability of correct description given the image
- Optimize sum of log probabilities over whole training set using SGD

$$\log p(S|I) = \sum_{t=0}^{N} \log p(S_t|I, S_0, \ldots, S_{t-1})$$

- The CNN follows winning entry of ILSVRC 2014
- On next page: W_e: word embedding function (takes in 1-of-V encoded word S_i); outputs probability distribution p_i; S_0 is start word, S_N is stop word
- Image input only once

The Model

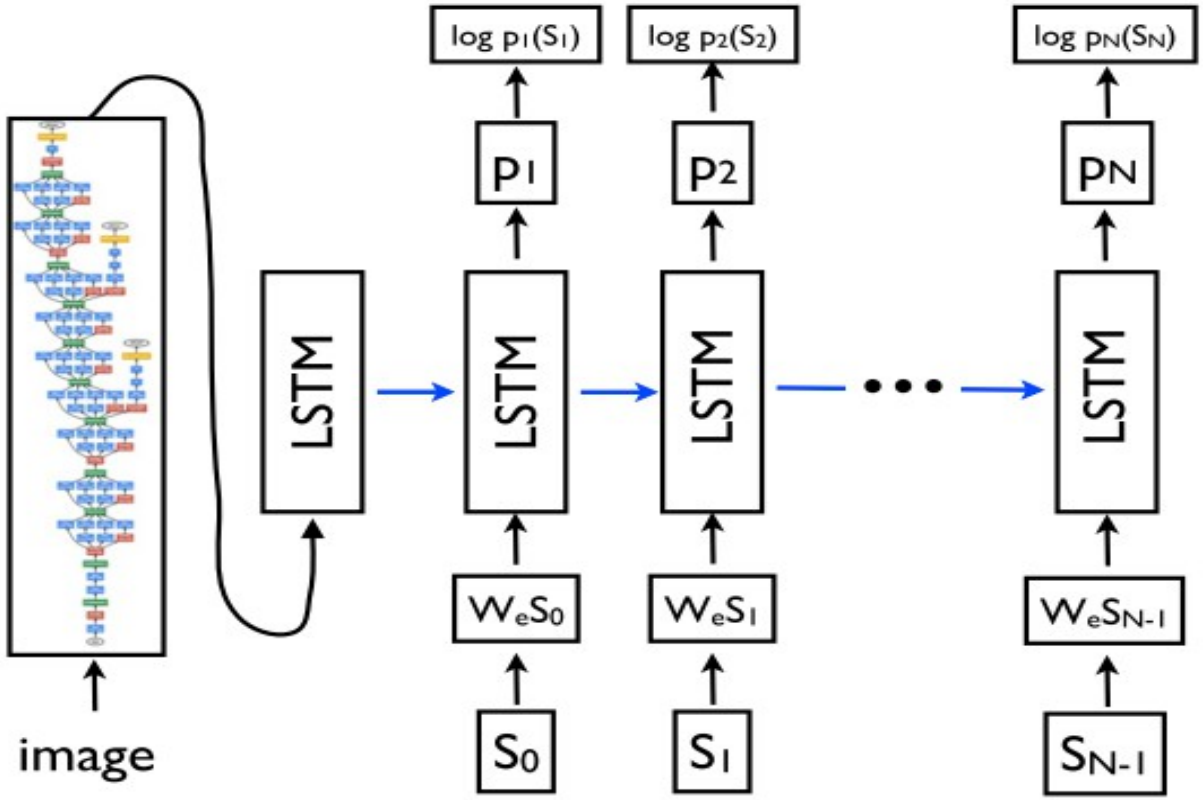


Figure 3. LSTM model combined with a CNN image embedder (as defined in [30]) and word embeddings. The unrolled connections between the LSTM memories are in blue and they correspond to the recurrent connections in Figure 2. All LSTMs share the same parameters.

Model (cont).

- LSTM model trained to predict word of sentence after it has seen image as well as previous words
- Use BPTT (Backprop through time) to train
- Recall we unroll the LSTM connections over time to view as feedforward net..
- Loss function: negative log likelihood as usual

$$L(I, S) = -\sum_{t=1}^{N} \log p_t(S_t) \ .$$

Generating the sentence

- Two approaches:
  - Sampling: sample word from p1, then from p2 (w/ corresponding embedding of the previous output as input) until reach a certain length or until we sample the EOS token
  - Beam search: keep k best sentences up to time t as candidates to generate t+1 size sentence.
    - Typically better, what they use
    - Beam size 20
    - Beam size 1 degrades results by 2 BLEU pts

Training Details

- Key: dealing with overfitting
- Purely supervised requires larger datasets (only 100000 images of high quality in given datasets)
- Can initialize weights of CNN (on ImageNet) → helped generalization
- Could init the W_e (word embeddings) → use Mikolov's word vectors, for instance → did not help
- Trained with SGD and no momentum; random inits except for CNN weights
- 512-size dims for embeddings

Evaluating Show and Tell

- Mech Turk experiment: human raters give a subjective score on the usefulness of descriptions
- each image rated by 2 workers on scale of 1-4; agreement between workers is 65% on average; take average when disagree
- BLEU score – baseline uses unigram, n = 1 to N gram uses geometric average of individual gram scores
- Also use perplexity (geometric mean of inverse probability for each predicted word), but do not report (BLEU preferred) – only used for hyperparameter tuning

Results

NIC is this paper's result.

Approach	MS COCO	PASCAL (xfer)	Flickr 30k	Flickr 8k	SBU
Im2Text [23]					11
TreeTalk [17]					19
BabyTalk [15]		25			
Tri5Sem [11]				48	
m-RNN [20]			55	58	
MNLM [13][4]			56	51	
SOTA		25	56	58	19
NIC	**67**	**59**	**66**	**63**	**28**
Human	69	69	68	70	

Table 1. BLEU-1 scores. We only report previous work results when available. SOTA stands for the current state-of-the-art.

Approach	BLEU-1	BLEU-2	BLEU-3
m-RNN [20]	58	28	23
NIC-8k	63	41	27
NIC-30k	67	45	30

Table 2. BLEU-{1,2,3} scores, on Flickr8k. NIC-30k is our model trained on Flickr30k, while NIC-8k is trained on Flickr8k.

Datasets Discussion

- Typically use MSCOCO or Flickr (8k, 30k)
  - Older test set used: Pascal dataset
  - 20 classes. The train/val data has 11,530 images containing 27,450 ROI annotated objects and 6,929 segmentations.
- Most use COCO
- SBU dataset also (Stonybrook) → descriptions by Flickr image owners, not guaranteed to be visual or unbiased

Evaluation Metrics: Issues w/Comparison

(2a) As far as evaluation metrics, the captioning community is in a bit of disarray. While the BLEU metric was adopted by many of the groups, the results in the various papers are NOT comparable due to various subtleties/choices in the BLEU metric. Lame. The COCO team is working to remedy this by setting up an automatic evaluation server where authors upload captions and comparisons are automatically generated (very standard stuff). We should have this up and running in a few weeks, many of the teams seem interested in uploading and comparing results once this is up and running. Then we will know who is best!

Furthermore, BLEU isn't even that good – has lots of issues

Motivation for a new, unambiguous and good metric

Evaluation Metrics Continued

- BLEU sucks (can get computer performance beating human performance)
- METEOR typically better (more intelligent, uses WordNet and doesn't penalize similar words)
- New metric: CIDEr by Devi Parikh
- Specific to Image Captioning
  - Triplet method to measure consensus
  - New datasets: 50 sentences describing each image

CIDEr (2)

- Goal: measure “human-likeness” - does sentence sound like it was written by a human?
- CIDEr: Consensus-based Image Description Evaluation
- Use Mech Turk to get human consensus
- Do not provide an explicit concept of similarity; the goal is to get humans to dictate what similarity means

CIDEr (3)

Reference Sentences

R1: A bicyclist makes a gesture as he rides along

R2: A cyclist posing on his bicycle while riding it.

R3: A disabled biker rides on the road.

R4: A man in racing gear riding a bike and making a funny face.

R5: The man is riding his bike on the street.

R6: A man riding his bike outside.

R7: A man riding his bike.

Candidate Sentences

C1: A man rides a bike with one hand.

C2: A male biker dressed in white rides on pavement with a landscape of tree and grass behind him.

Triplet Annotation

Which of the sentences, B or C, is more similar to sentence A?

Sentence A : Anyone from R1 to R50

Sentence B : C1

Sentence C : C2

(a) (b) (c)

Figure 2: We show an illustration of our triplet annotation modality. Given an image (a), with reference sentences (b) and a pair of candidate sentences (c), we match them with a reference sentence one by one to form triplets. Subjects are shown these 50 triplets (red box) on Amazon Mechanical Turk and asked to pick which sentence (B or C) is more similar to sentence A.

CIDEr Metric

- Measure of consensus should encode how often n-grams in candidate sentence are present in references
- More frequent n-grams in references are less informative if n-gram is in candidate
- → use TF-IDF weighting for each n-gram
  - (term frequency - inverse doc frequency)
  - s_ij sentence, h_k(s_ij) count for w_k in s_ij

$$g_k(s_{ij}) = \frac{h_k(s_{ij})}{\sum_{\omega_l \in \Omega} h_l(s_{ij})} \log\left(\frac{|I|}{\sum_{I_p \in I} \min(1, \sum_q h_k(s_{pq}))}\right)$$

CIDEr Metric (2)

For a fixed size of n-gram:

$$\text{CIDEr}_n(c_i, S_i) = \frac{1}{m} \sum_j \frac{\boldsymbol{g}^n(c_i) \cdot \boldsymbol{g}^n(s_{ij})}{\|\boldsymbol{g}^n(c_i)\| \|\boldsymbol{g}^n(s_{ij})\|},$$

Over all n-grams considered (up to N):

$$\text{CIDEr}(c_i, S_i) = \sum_{n=1}^{N} w_n \text{CIDEr}_n(c_i, S_i),$$

Empirically: w_i = 1 is best, N = 4

CIDEr Metric (3)

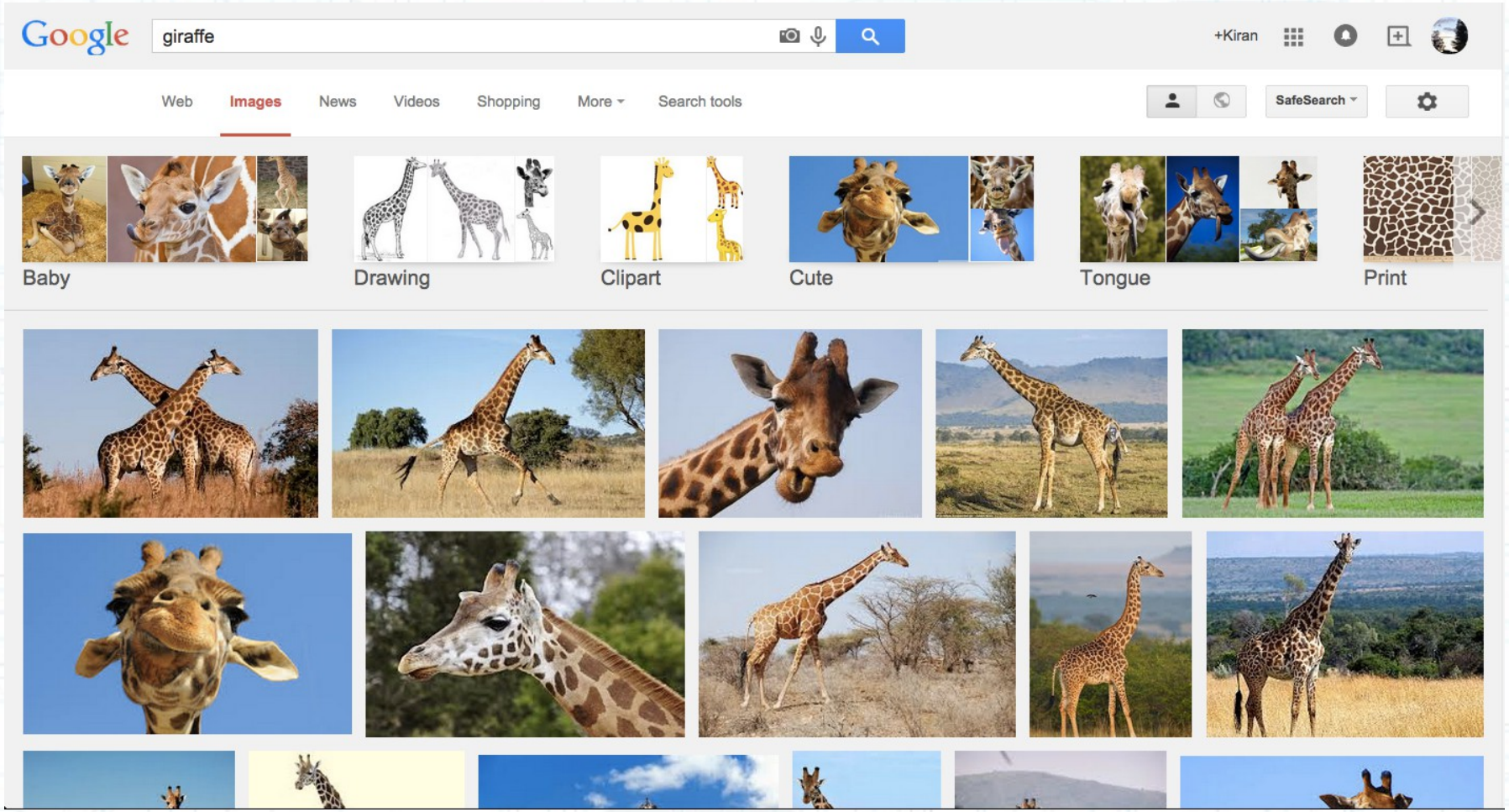
We also experimented with non-uniform w_n that weighs the longer n-grams more, soft word-level semantic similarity measures such as WordNet similarity [24] and word2vec similarity [23], as well as various relevance weighting schemes from information retrieval [31]. We found that the simple metric we present here performs the best.

Next Tasks for Image Captioning

- Recall: why is Image Captioning an interesting task?
  - Supposedly requires a detailed understanding an image and an ability to communicate that information via natural language.
- This is not necessarily true though – the problem can be solved with only partial image understanding and rudimentary langauge modeling (recall Microsoft paper only used basic language model)

The Giraffe-Tree Problem

“ A giraffe standing next to a tree”



Alternative Tasks

- We want more challenging tasks!
- Some suggestions: Question-answer (ask question about an image, get an answer in natural language)
- Issue: large-scale QA datasets are difficult to define and build
- Video Captioning Dataset
  - Linguistic descriptions of movies
  - 54000 sentences, snippets from 72 HD movies
- Defining challenges is an open problem

Thank you for listening!

Citations

- 1. Vedantam, R., Zitnick, C. L. & Parikh, D. CIDEr: Consensus-based Image Description Evaluation. (2014). at <http://arxiv.org/abs/1411.5726>
- 2. Karpathy, A. Deep Visual-Semantic Alignments for Generating Image Descriptions.
- 3. Mao, J., Xu, W., Yang, Y., Wang, J. & Yuille, A. L. Explain Images with Multimodal Recurrent Neural Networks. 1–9 (2014). at <http://arxiv.org/abs/1410.1090v1>
- 4. Fang, H. et al. From Captions to Visual Concepts and Back. (2014). at <http://arxiv.org/abs/1411.4952v2>
- 5. Rohrbach, A., Rohrbach, M., Tandon, N. & Schiele, B. A Dataset for Movie Description. (2015). at <http://arxiv.org/abs/1501.0253>
- 6. Krizhevsky, A. & Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. 1–9
- 7. Chen, X. & Zitnick, C. L. Learning a Recurrent Visual Representation for Image Caption Generation. (2014). at <http://arxiv.org/abs/1411.5654v1>
- 8. Donahue, J. et al. Long-term Recurrent Convolutional Networks for Visual Recognition and Description. (2014). at <http://arxiv.org/abs/1411.4389v2>
- 9. Vinyals, O. & Toshev, A. Show and Tell: A Neural Image Caption Generator.
- 10. Venugopalan, S. et al. Translating Videos to Natural Language Using Deep Recurrent Neural Networks. (2014). at <http://arxiv.org/abs/1412.4729>
- 11. Kiros, R., Salakhutdinov, R. & Zemel, R. S. Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models. 1–13 (2014). at <http://arxiv.org/abs/1411.2539v1>
- 12. Och, F. J. Minimum Error Rate Training. (2003).
- 13. https://pdollar.wordpress.com/2015/01/21/image-captioning/
- 14. http://blogs.technet.com/b/machinelearning/archive/2014/11/18/rapid-progress-in-automatic-image-captioning.aspx