Multimodal Introduction

ICBU

Shaola Ren

Alibaba.com®
Global trade starts here.™

Outline

- DeViSE: A Deep Visual-Semantic Embedding Model (2013)
- Zero-Shot Learning Through Cross-Modal Transfer (2013)
- Deep Fragment Embeddings for Bidirectional Image Sentence Mapping (2014.6)
- Deep Visual-Semantic Alignments for Generating Image Descriptions (2015.4)

DeViSE: A Deep Visual-Semantic Embedding Model (2013)

- 任务：图片的label预测，现在的视觉识别系统依然受限于物体category的数量，当类目数量增大时，往往标注过的训练样本就不够了

- 设想的解决方案：从另外的渠道引入信息，这里考虑的就是加入语言模型，propose了这个deep visual-semantic embedding model

- 评估方法：针对图片，1000-class ImageNet object recognition，zero-shot predictions即训练中从没出现过的category的预测

DeViSE: A Deep Visual-Semantic Embedding Model (2013)

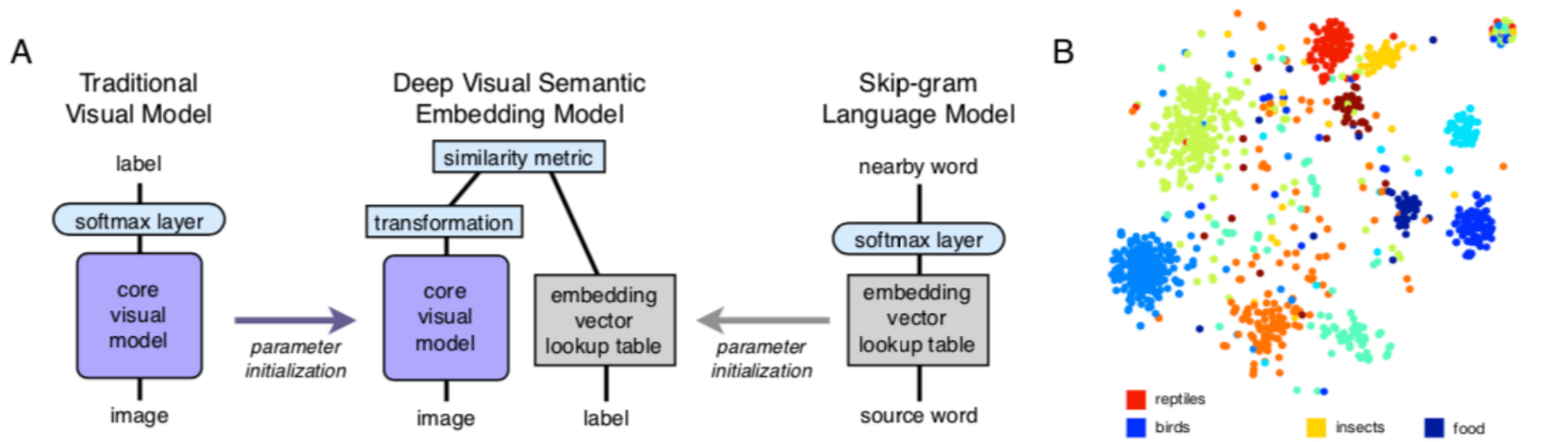


Figure 1: (a) Left: a visual object categorization network with a softmax output layer; Right: a skip-gram language model; Center: our joint model, which is initialized with parameters pre-trained at the lower layers of the other two models. (b) t-SNE visualization [19] of a subset of the ILSVRC 2012 1K label embeddings learned using skip-gram.

$$loss(image, label) = \sum_{j \neq label} \max[0, margin - \vec{t}_{label} M\vec{v}(image) + \vec{t}_j M\vec{v}(image)] \quad (1)$$

DeViSE: A Deep Visual-Semantic Embedding Model (2013)

- 训练word embedding
- 训练图片特征提取网络
- word embedding和图片特征提取网络的融合
- 图片特征从高维到word embedding维度的映射M
- 论文中bp只传图片网络这个分支；可以考虑和word embedding一起训练
- M were first trained while holding both the core visual model and the text representation fixed. In the later stages of training the derivative of the loss function was back-propagated into the core visual model to fine-tune its output

DeViSE: A Deep Visual-Semantic Embedding Model (2013)

$$loss(image, label) = \sum_{j \neq label} \max[0, margin - \vec{t}_{label} M \vec{v}(image) + \vec{t}_j M \vec{v}(image)] \quad (1)$$

- margin：保证正负例的rank分数之间至少有这么大的距离 这个模型预测时，是输入图片，生成图片的label向量
- 为什么不用L2 Loss：L2目标是降低向量之间的距离，如果有离图片距离近但和图片不匹配的label向量时，倾向于生成错误的label向量；实验效果也证明没有margin好。

DeViSE: A Deep Visual-Semantic Embedding Model (2013)

- 模型评估：输入图片，看生成的label向量在多大范围内命中ground truth的label，flat hit@k，hierarchical precision@k

- 步骤

  - 输入图片，经过图片特征提取网络，及transform的部分，会生成这个图片的label向量

  - 在所有图片的ground truth的label向量集合中查找和生成的label向量距离相近的向量，取出距离最近的k个，作为这个图片的label的预测值

1. Flat hit@k，预测的top k个label，有多少个测试样本的真实label包含在这top k中
2. Hierarchical precision@k，真实的label构成一个hierarchical的结构，预测的hierachical的结构的top k中有和真实的结构相同的测试样本占比

DeViSE: A Deep Visual-Semantic Embedding Model (2013)

Model type	dim	Flat hit@k (%)				Hierarchical precision@k			
		1	2	5	10	2	5	10	20
Softmax baseline	N/A	**55.6**	**67.4**	**78.5**	**85.0**	0.452	0.342	0.313	0.319
DeViSE	500	53.2	65.2	76.7	83.3	0.447	**0.352**	**0.331**	**0.341**
	1000	54.9	66.9	78.4	**85.0**	**0.454**	0.351	0.325	0.331
Random embeddings	500	52.4	63.9	74.8	80.6	0.428	0.315	0.271	0.248
	1000	50.5	62.2	74.2	81.5	0.418	0.318	0.290	0.292
Chance	N/A	0.1	0.2	0.5	1.0	0.007	0.013	0.022	0.042

Table 1: Comparison of model performance on our test set, taken from the ImageNet ILSVRC 2012 1K validation set. Note that hierarchical precision@1 is equivalent to flat hit@1. See text for details.

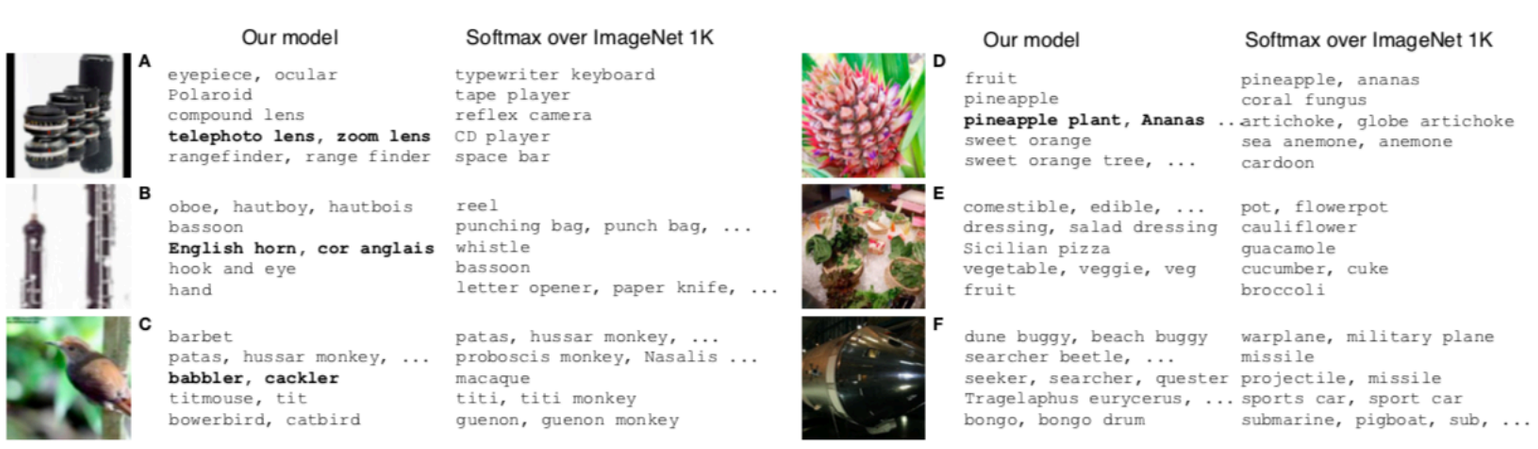


Figure 2: For each image, the top 5 zero-shot predictions of DeViSE+1K from the 2011 21K label set and the softmax baseline model, both trained on ILSVRC 2012 1K. Predictions ordered by decreasing score, with correct predictions in bold. Ground truth: (a) *telephoto lens, zoom lens*; (b) *English horn, cor anglais*; (c) *babbler, cackler*; (d) *pineapple, pineapple plant, Ananas comosus*; (e) *salad bar*; (f) *spacecraft, ballistic capsule, space vehicle*.

非zero-shot的数据上，其实并没有多大提升

zero-shot的数据上，有可能把正确的label预测出来；原因是有了模型的word embedding这部分网络，模型可以将图片的特征映射成和ground truth的label距离近的向量，虽然训练数据中没有出现过这样的label，依然可以把正确的label预测出来

Zero-Shot Learning Through Cross-Modal Transfer

- This work introduces a model that can recognize objects in images even if no training data is available for the objects

- 方法：除了图片还利用了大量文本语料

- 创新点：在识别object的时候，加了个outlier detection，然后分别对known classes和unseen classes做识别

- 感觉虽然这是一个toy级别的模型，但能看到图片的特征是有可能和语义特征对应起来的，我们有可能从图片中提取出语义标签的

Zero-Shot Learning Through Cross-Modal Transfer

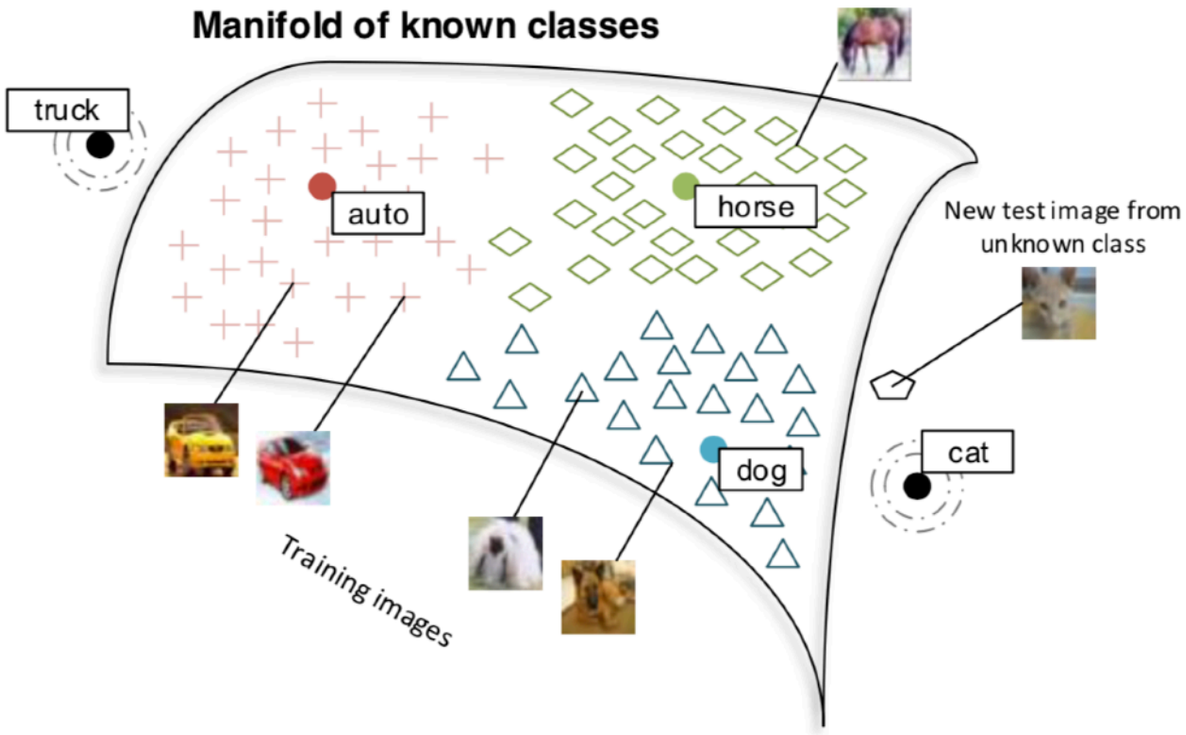


Figure 1: Overview of our multi-modal zero-shot model. We first map each new testing image into a lower dimensional semantic space. Then, we use outlier detection to determine whether it is on the manifold of seen images. If the image is not on the manifold, we determine its class with the help of unsupervised semantic word vectors. In this example, the unseen classes are truck and cat.

$$J(\theta) = \sum_{y \in Y_s} \sum_{x^{(i)} \in X_y} \|w_y - \theta x^{(i)}\|^2.$$

1. 生成word embedding
2. 生成图片的特征向量，这里用的是比较早的一个unsupervised的方法Sparse coding
3. 用L2 loss，将图片特征project到word embedding的空间

Zero-Shot Learning Through Cross-Modal Transfer

- outlier detection

$$p(y|x, X_s, W, \theta) = \sum_{V \in \{s,u\}} P(y|V, x, X_s, W, \theta) P(V|x, X_s, W, \theta).$$

$$\text{marginal: } P(x|X_s, W_s, \theta) = \sum_{y \in Y_s} P(x|y)P(y) = \sum_{y \in Y_s} \mathcal{N}(\theta x|w_y, \Sigma_y)P(y).$$

$$P(V = u|x, X_s, W, \theta) := \mathbb{1}\{P(x|X_s, W_s, \theta) < T\}$$

Gaussian的参数估计，mean是对应class的label的word embedding，covariance matrix是通过所有map到这个label的数据计算的

1. 用所有有label的图片的图片特征和label向量做了个混合Gaussian
2. 输入一个图片的向量就会得到一个概率值，如果小于某个阈值，就判定这个图片known classes里面
3. 如果是known classes按传统的分类方法判定图片的label。如果是unseen，虽然label在训练数据中没出现过，但预测的时候可能会有哪些label是知道的，看图片向量map到label语义的哪部分来判断类别。

In the case of a known class, a softmax classifier on the original F-dimensional features is used. For the zero-shot case, assuming an isometric Gaussian distribution around each of the zero-shot semantic word vectors.

Zero-Shot Learning Through Cross-Modal Transfer

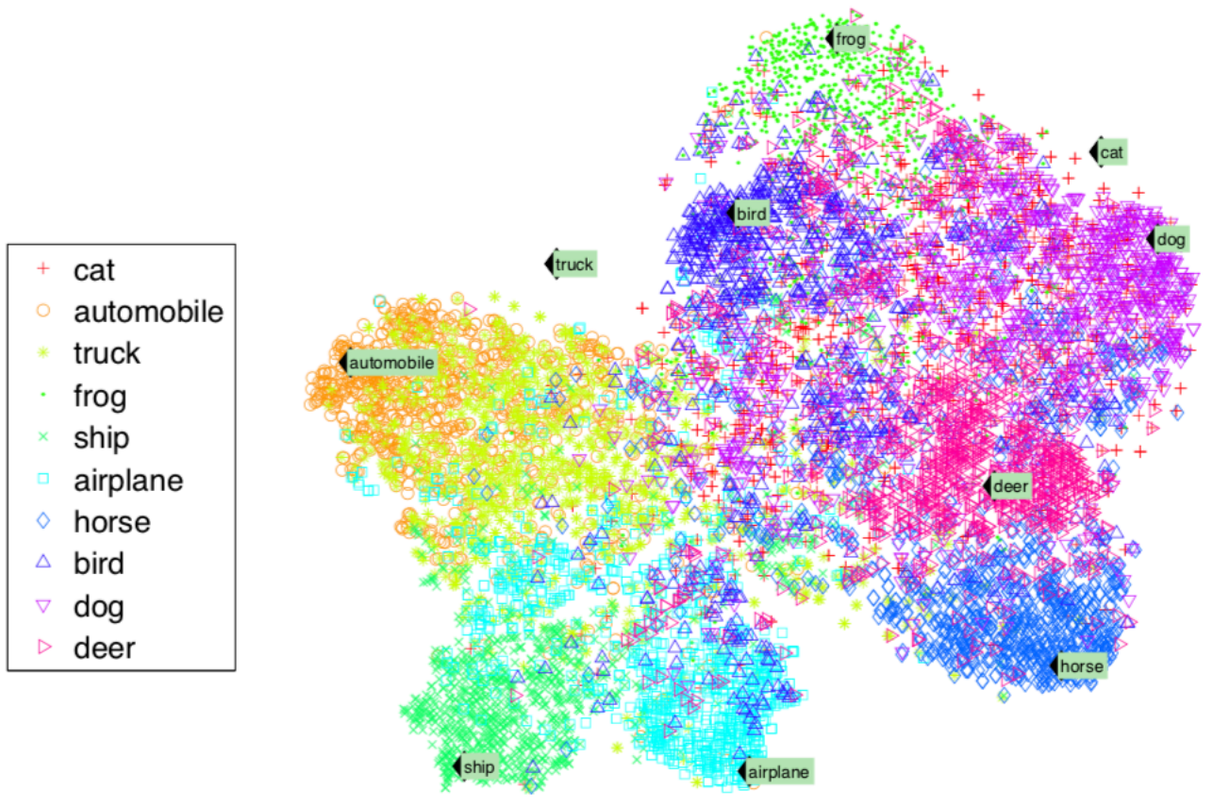


Figure 2: T-SNE visualization of the semantic word space. Word vector locations are highlighted and mapped image locations are shown both for images for which this mapping has been trained and unseen images. The unseen classes are cat and truck.

要识别unseen的图片的内容，只有当训练数据中有和unseen的图片相似的类别时效果才比较好，比方说cat的图在训练数据中没出现过，但训练数据中有dog，则模型效果还可以，如果连dog都没有，那效果就比较差了。

Deep Fragment Embeddings for Bidirectional Image Sentence Mapping

- 以前的图片文本multimodal都是把整张图片或整个句子映射到共同的embedding space，这里介绍的模型从更细的粒度上考虑这个问题，图片和句子都分成一个个的片段，将这些小粒度的图片和句子映射到共同的embedding空间，并且考虑图片fragment和句子fragment之间的alignment。

- bidirectional：给定图片和句子集合，用这个模型可以通过图片查找相关的句子(image annotation)，也可以通过句子查找相关的图片(image search)

Deep Fragment Embeddings for Bidirectional Image Sentence Mapping

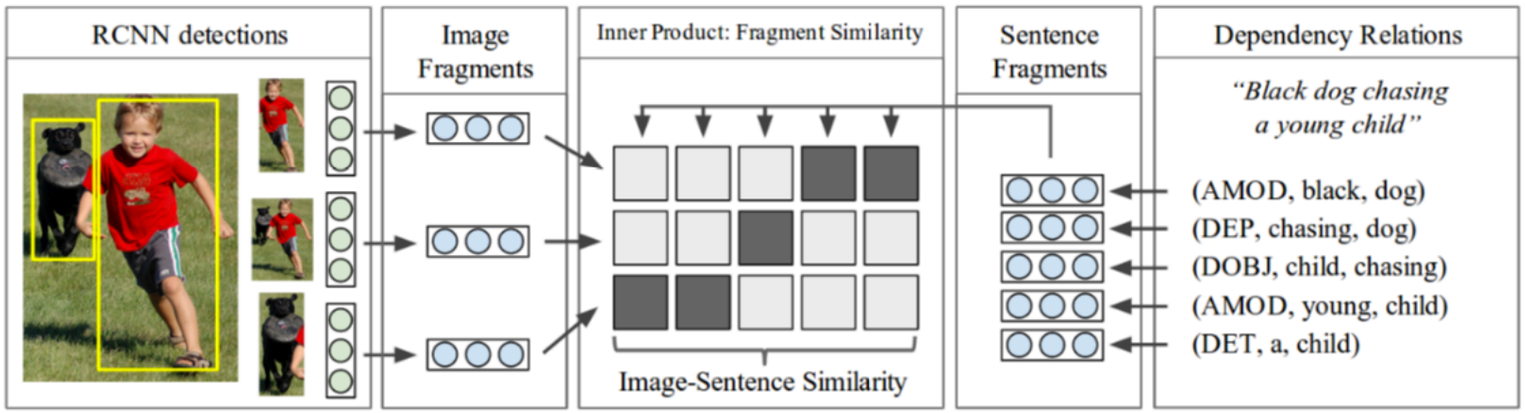


Figure 2: Computing the Fragment and image-sentence similarities. **Left:** CNN representations (green) of detected objects are mapped to the fragment embedding space (blue, Section 3.2). **Right:** Dependency tree relations in the sentence are embedded (Section 3.1). Our model interprets inner products (shown as boxes) between fragments as a similarity score. The alignment (shaded boxes) is latent and inferred by our model (Section 3.3.1). The image-sentence similarity is computed as a fixed function of the pairwise fragment scores.

$$v = W_m[CNN_{\theta_c}(I_b)] \qquad \mathcal{C}(\theta) = \mathcal{C}_F(\theta) + \beta\mathcal{C}_G(\theta) + \alpha||\theta||_2^2 \qquad s = f\left(W_R \begin{bmatrix} W_e\mathbf{w}_1 \\ W_e\mathbf{w}_2 \end{bmatrix} + b_R\right)$$

$$\mathcal{C}_0(\theta) = \sum_i \sum_j \kappa_{ij} max(0, 1 - y_{ij} v_i^T s_j)$$

$$\mathcal{C}_F(\theta) = \min_{y_{ij}} \mathcal{C}_0(\theta)$$

$$\text{s.t.} \quad \sum_{i \in p_j} \frac{y_{ij}+1}{2} \geq 1 \quad \forall j$$

$$y_{ij} = -1 \quad \forall i, j \quad \text{s.t.} \quad m_v(i) \neq m_s(j) \text{ and } y_{ij} \in \{-1, 1\}$$

$$S_{kl} = \frac{1}{|g_k|(|g_l| + n)} \sum_{i \in g_k} \sum_{j \in g_l} max(0, v_i^T s_j)$$

$$\mathcal{C}_G(\theta) = \sum_k \Big[\underbrace{\sum_l max(0, S_{kl} - S_{kk} + \Delta)}_{\text{rank images}} + \underbrace{\sum_l max(0, S_{lk} - S_{kk} + \Delta)}_{\text{rank sentences}} \Big]$$

1. Region Convolutional Neural Network (RCNN), top 19 detected locations and the entire image as the image fragments
2. represent every word using 1-of-k encoding vector w using a dictionary of 400,000 words and map every dependency triplet (R, w1, w2) into the embedding space, f->Relu
3. Objective funciton: global ranking objective and fragment alignment objective

Deep Fragment Embeddings for Bidirectional Image Sentence Mapping

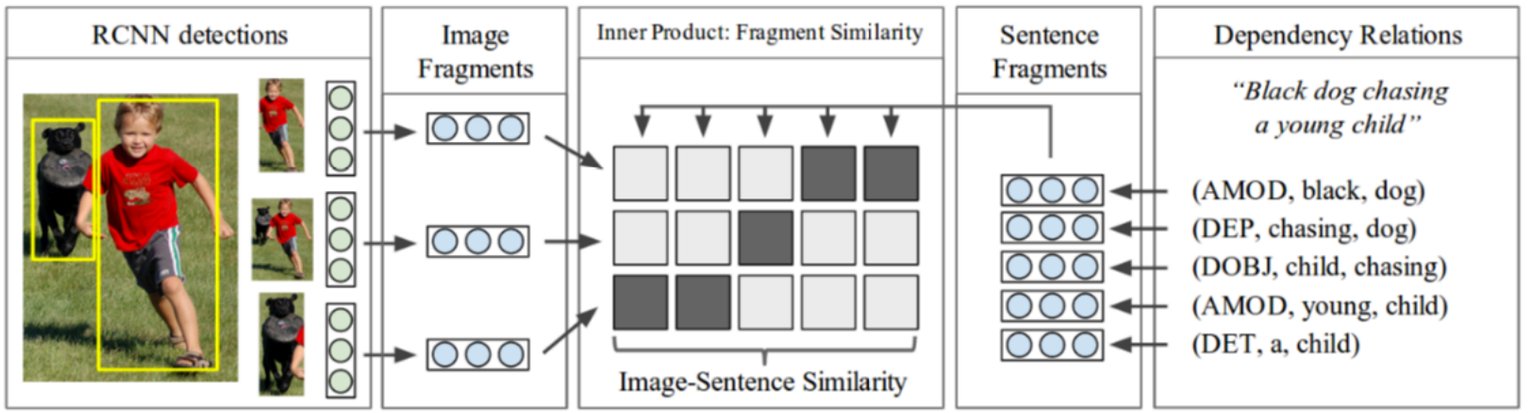


Figure 2: Computing the Fragment and image-sentence similarities. **Left:** CNN representations (green) of detected objects are mapped to the fragment embedding space (blue, Section 3.2). **Right:** Dependency tree relations in the sentence are embedded (Section 3.1). Our model interprets inner products (shown as boxes) between fragments as a similarity score. The alignment (shaded boxes) is latent and inferred by our model (Section 3.3.1). The image-sentence similarity is computed as a fixed function of the pairwise fragment scores.

$$v = W_m[CNN_{\theta_c}(I_b)] \qquad \mathcal{C}(\theta) = \mathcal{C}_F(\theta) + \beta\mathcal{C}_G(\theta) + \alpha||\theta||_2^2 \qquad s = f\left(W_R \begin{bmatrix} W_e\mathbf{w}_1 \\ W_e\mathbf{w}_2 \end{bmatrix} + b_R\right)$$

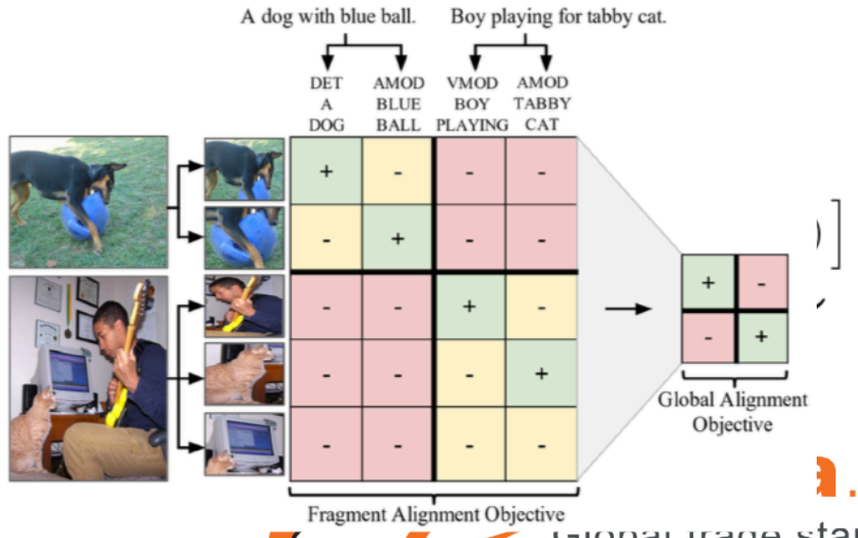
$$\mathcal{C}_0(\theta) = \sum_i \sum_j \kappa_{ij} max(0, 1 - y_{ij} v_i^T s_j)$$

$$\mathcal{C}_F(\theta) = \min_{y_{ij}} \mathcal{C}_0(\theta)$$

$$\text{s.t.} \quad \sum_{i \in p_j} \frac{y_{ij} + 1}{2} \geq 1 \quad \forall j$$

$$y_{ij} = -1 \quad \forall i, j \quad \text{s.t.} \quad m_v(i) \neq m_s(j) \text{ and } y_{ij} \in \{-1, 1\}$$

Figure 3: The two objectives for a batch of 2 examples. **Left:** Rows represent fragments v_i, columns s_j. Every square shows an ideal scenario of $y_{ij} = sign(v_i^T s_j)$ in the MIL objective. Red boxes are $y_{ij} = -1$. Yellow indicates members of positive bags that happen to currently be $y_{ij} = -1$. **Right:** The scores are accumulated with Equation 6 into image-sentence score matrix S_{kl}.



1. Region Convolutional Neural Network (RCNN), top 19 detected locations and the entire image as the image fragments
2. represent every word using 1-of-k encoding vector w using a dictionary of 400,000 words and map every dependency triplet (R, w1, w2) into the embedding space, f->Relu
3. Objective funciton: global ranking objective and fragment alignment objective

Deep Fragment Embeddings for Bidirectional Image Sentence Mapping

3.4 Optimization

We use Stochastic Gradient Descent (SGD) with mini-batches of 100, momentum of 0.9 and make 15 epochs through the training data. The learning rate is cross-validated and annealed by a fraction of $\times 0.1$ for the last two epochs. Since both Multiple Instance Learning and CNN finetuning benefit from a good initialization, we run the first 10 epochs with the fragment alignment objective $\mathcal{C}_0$ and CNN weights θ_c fixed. After 10 epochs, we switch to the full MIL objective $\mathcal{C}_F$ and begin finetuning the CNN. The word embedding matrix W_e is kept fixed due to overfitting concerns. Our implementation runs at approximately 1 second per batch on a standard CPU workstation.

评估方法：

1. 测试集是image-sentence对，say n组数据
2. 测试模型对image的效果，输入image，通过图像的网络得到图片的向量表示；n个sentence通过句法依从网络计算出对应的向量表示；计算输入图片和所有句子的S_kl，按降序排列，top k即为和这个输入图片高相关的k个句子
3. 测试模型对sentence的效果，输入sentence，通过句法依从网络计算出对应的向量表示；n个图片通过图像的网络得到对应的向量表示；计算输入句子和所有图片的S_kl，按降序排列，top k即为和这个输入句子高相关的k个图片

效果评估和下面一篇一起看

Deep Visual-Semantic Alignments for Generating Image Descriptions

- 前面这几篇论文中图片和文本之间的学习是匹配的方法，这篇尝试生成图片的描述

- 分三个步骤：1. 图片句子的匹配模型，2. 文本和图片区域对齐，3. 用图片或图片区域生成句子片段

- 1是一个图片句子ranking的匹配模型，2是dot product，不需要单独训练，3是一个以图片为初始条件输入的RNN模型

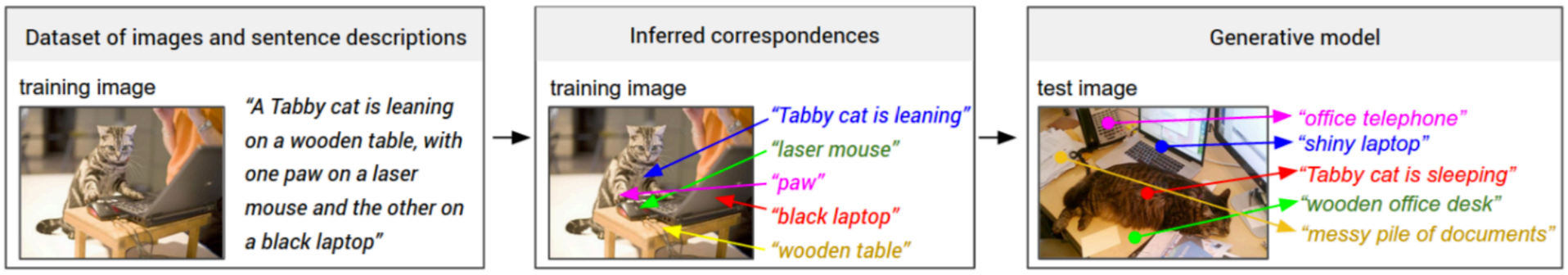


Figure 2. Overview of our approach. A dataset of images and their sentence descriptions is the input to our model (left). Our model first infers the correspondences (middle, Section 3.1) and then learns to generate novel descriptions (right, Section 3.2).

Deep Visual-Semantic Alignments for Generating Image Descriptions

第1、2个步骤

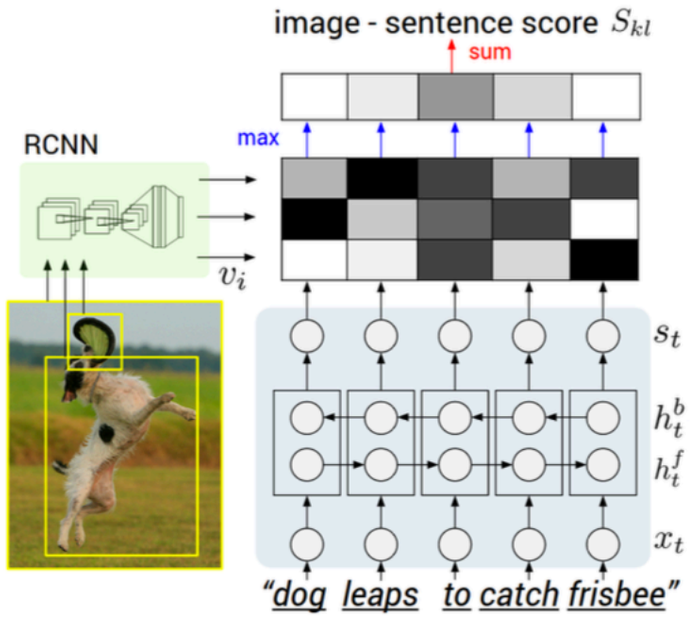


Figure 3. Diagram for evaluating the image-sentence score S_{kl}. Object regions are embedded with a CNN (left). Words (enriched by their context) are embedded in the same multimodal space with a BRNN (right). Pairwise similarities are computed with inner products (magnitudes shown in grayscale) and finally reduced to image-sentence score with Equation 8.

RCNN

CNN提前训好，其它参数学习得到

$$v = W_m[CNN_{\theta_c}(I_b)] + b_m, \tag{1}$$

BRNN

x是提前训好的word2vec
其它参数学习得到

$$x_t = W_w \mathbb{I}_t \tag{2}$$
$$e_t = f(W_e x_t + b_e) \tag{3}$$
$$h_t^f = f(e_t + W_f h_{t-1}^f + b_f) \tag{4}$$
$$h_t^b = f(e_t + W_b h_{t+1}^b + b_b) \tag{5}$$
$$s_t = f(W_d(h_t^f + h_t^b) + b_d). \tag{6}$$

$$S_{kl} = \sum_{t \in g_l} max_{i \in g_k} v_i^T s_t. \tag{8}$$

Loss

$$\mathcal{C}(\theta) = \sum_k \Big[\underbrace{\sum_l max(0, S_{kl} - S_{kk} + 1)}_{\text{rank images}} + \underbrace{\sum_l max(0, S_{lk} - S_{kk} + 1)}_{\text{rank sentences}} \Big]. \tag{9}$$

Alignment: MRF

$$E(\mathbf{a}) = \sum_{j=1\ldots N} \psi_j^U(a_j) + \sum_{j=1\ldots N-1} \psi_j^B(a_j, a_{j+1}) \tag{10}$$
$$\psi_j^U(a_j = t) = v_i^T s_t \tag{11}$$
$$\psi_j^B(a_j, a_{j+1}) = \beta \mathbb{1}[a_j = a_{j+1}]. \tag{12}$$

Deep Visual-Semantic Alignments for Generating Image Descriptions

第1、2个步骤

- detect objects in every image with a Region Convolutional Neural Network (RCNN)
- The CNN is pre-trained on ImageNet and finetuned on the 200 classes of the ImageNet Detection Challenge
- use the top 19 detected locations in addition to the whole image and compute the representations based on the pixels inside each bounding box

RCNN

$$v = W_m[CNN_{\theta_c}(I_b)] + b_m, \tag{1}$$

- initialize with 300-dimensional word2vec weights and keep fixed due to overfitting concerns
- BRNN consists of two independent streams of processing, one moving left to right and the other right to left
- The final h-dimensional representation for the t-th word is a function of both the word at that location and also its surrounding context in the sentence
- set the activation function f to the rectified linear unit (ReLU)

BRNN

$$x_t = W_w \mathbb{I}_t \tag{2}$$

$$e_t = f(W_e x_t + b_e) \tag{3}$$

$$h_t^f = f(e_t + W_f h_{t-1}^f + b_f) \tag{4}$$

$$h_t^b = f(e_t + W_b h_{t+1}^b + b_b) \tag{5}$$

$$s_t = f(W_d(h_t^f + h_t^b) + b_d). \tag{6}$$

- k = l denotes a corresponding image and sentence pair
- This objective encourages aligned image-sentences pairs to have a higher score than misaligned pairs, by a margin

$$S_{kl} = \sum_{t \in g_l} max_{i \in g_k} v_i^T s_t. \tag{8}$$

Loss

$$\mathcal{C}(\theta) = \sum_k \Big[\underbrace{\sum_l max(0, S_{kl} - S_{kk} + 1)}_{\text{rank images}} + \underbrace{\sum_l max(0, S_{lk} - S_{kk} + 1)}_{\text{rank sentences}} \Big]. \tag{9}$$

Deep Visual-Semantic Alignments for Generating Image Descriptions

第3个步骤

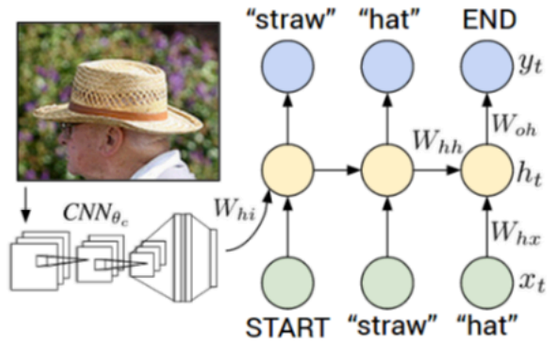


Figure 4. Diagram of our multimodal Recurrent Neural Network generative model. The RNN takes a word, the context from previous time steps and defines a distribution over the next word in the sentence. The RNN is conditioned on the image information at the first time step. START and END are special tokens.

$$b_v = W_{hi}[CNN_{\theta_c}(I)] \quad (13)$$
$$h_t = f(W_{hx}x_t + W_{hh}h_{t-1} + b_h + \mathbb{1}(t=1) \odot b_v) \quad (14)$$
$$y_t = softmax(W_{oh}h_t + b_o). \quad (15)$$

- input: images and their textual descriptions, full images and their sentence descriptions, or regions and text snippets
- RNN: achieved by defining a probability distribution of the next word in a sequence given the current word and context from previous time steps, and additionally conditioning the generative process on the content of an input image
- CNN pretrained
- with special START vector and END token
- beam size 7
- The generative RNN is more difficult to optimize, party due to the word frequency disparity between rare words and common words. RMSprop is used

评估指标

1. Recall@k，预测的top k个label，有多少个测试样本的真实label包含在这top k中(which measures the fraction of times a correct item was found among the top K results)
2. Med r，真实label在top n预测结果list中的位置，所有测试样本的这个指标的中位数(median rank of the closest ground truth result in the list)
3. BLEU，通俗的说就是，生成的文本和参考文本在不同粒度上的重合度

Deep Visual-Semantic Alignments for Generating Image Descriptions

Model	Image Annotation R@1	R@5	R@10	Med *r*	Image Search R@1	R@5	R@10	Med *r*
Flickr30K								
SDT-RNN (Socher et al. [49])	9.6	29.8	41.1	16	8.9	29.8	41.1	16
Kiros et al. [25]	14.8	39.2	50.9	10	11.8	34.0	46.3	13
Mao et al. [38]	18.4	40.2	50.9	10	12.6	31.2	41.5	16
Donahue et al. [8]	17.5	40.3	50.8	9	-	-	-	-
DeFrag (Karpathy et al. [24])	14.2	37.7	51.3	10	10.2	30.8	44.2	14
Our implementation of DeFrag [24]	19.2	44.5	58.0	6.0	12.9	35.4	47.5	10.8
Our model: DepTree edges	20.0	46.6	59.4	5.4	15.0	36.5	48.2	10.4
Our model: BRNN	**22.2**	**48.2**	**61.4**	**4.8**	**15.2**	**37.7**	**50.5**	**9.2**
Vinyals et al. [54] (more powerful CNN)	23	-	63	5	17	-	57	8
MSCOCO								
Our model: 1K test images	38.4	69.9	80.5	1.0	27.4	60.2	74.8	3.0
Our model: 5K test images	16.5	39.2	52.0	9.0	10.7	29.6	42.2	14.0

Table 1. Image-Sentence ranking experiment results. **R@K** is Recall@K (high is good). **Med** *r* is the median rank (low is good). In the results for our models, we take the top 5 validation set models, evaluate each independently on the test set and then report the average performance. The standard deviations on the recall values range from approximately 0.5 to 1.0.

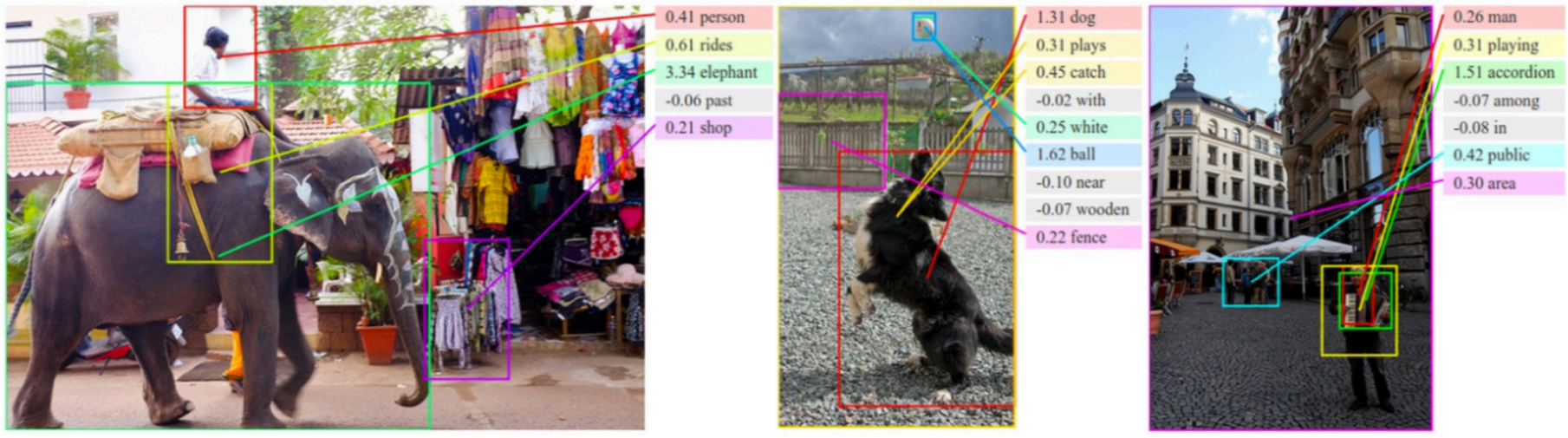


Figure 5. Example alignments predicted by our model. For every test image above, we retrieve the most compatible test sentence and visualize the highest-scoring region for each word (before MRF smoothing described in Section 3.1.4) and the associated scores ($v_i^T s_t$). We hide the alignments of low-scoring words to reduce clutter. We assign each region an arbitrary color.

Deep Visual-Semantic Alignments for Generating Image Descriptions

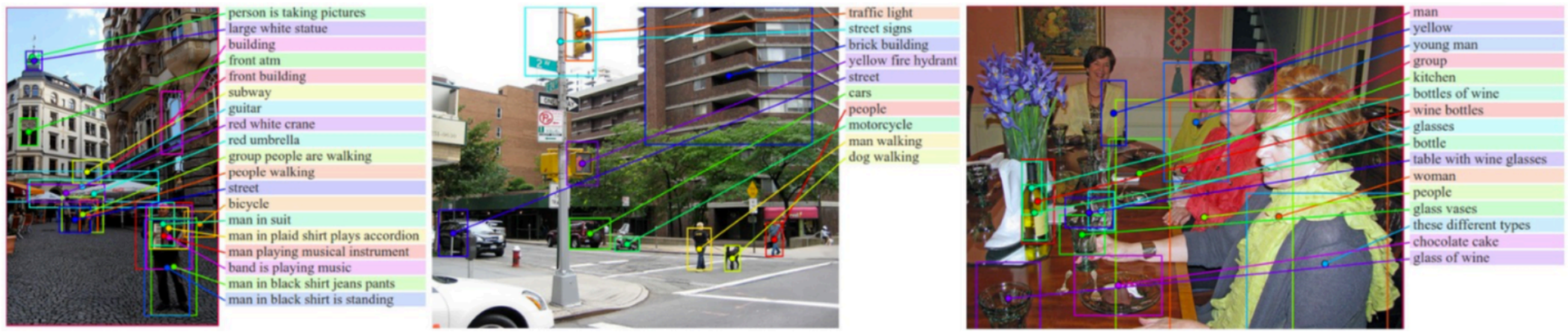


Figure 7. Example region predictions. We use our region-level multimodal RNN to generate text (shown on the right of each image) for some of the bounding boxes in each image. The lines are grounded to centers of bounding boxes and the colors are chosen arbitrarily.

Model	B-1	B-2	B-3	B-4
Human agreement	61.5	45.2	30.1	22.0
Nearest Neighbor	22.9	10.5	0.0	0.0
RNN: Fullframe model	14.2	6.0	2.2	0.0
RNN: Region level model	**35.2**	**23.0**	**16.1**	**14.8**

Table 3. BLEU score evaluation of image region annotations.

p_n = Bleu score on n-grams only p_1, p_2, p_3, p_4

Combined Bleu score: $BP \exp\left(\frac{1}{4}\sum_{n=1}^{4} p_n\right)$

BP = brevity penalty

$$BP = \begin{cases} 1 & \text{if MT\_output\_length} > \text{reference\_output\_length} \\ \exp(1 - \text{MT\_output\_length}/\text{reference\_output\_length}) & \text{otherwise} \end{cases}$$

Example: Reference 1: The cat is on the mat.

Reference 2: There is a cat on the mat.

MT output: The cat the cat on the mat. $(\hat{y})$

$$p_1 = \frac{\sum_{\text{unigram} \in \hat{y}} \text{Count}_{\text{clip}}(\text{unigram})}{\sum_{\text{unigram} \in \hat{y}} \text{Count}(\text{unigram})} \qquad p_n = \frac{\sum_{n\text{-gram} \in \hat{y}} \text{Count}_{\text{clip}}(n\text{-gram})}{\sum_{n\text{-gram} \in \hat{y}} \text{Count}(n\text{-gram})}$$

unigram n-gram

[Papineni et. al., 2002. Bleu: A method for automatic evaluation of machine translation]

Andrew Ng