Sentence-based image description with scalable, explicit models

Micah Hodosh Julia Hockenmaier
University of Illinois at Urbana-Champaign
{mhodosh2, juliahmr}@illinois.edu

Abstract

Associating photographs with complete sentences that describe what is depicted in them is a challenging problem. This paper examines how an approach that is inspired by image tagging techniques which can scale to very large data sets performs on this much harder task, and examines some of the linguistic difficulties that this bag-of-words model faces.

1. Introduction

The ambitious task of creating models for annotating photographs with natural sounding English language descriptions has begun to attract attention in recent work [19, 13, 14, 18, 9, 11]. However, many of these works rely on pre-trained classifiers such as object detectors [19, 13, 14, 18]. Relying on out-of-domain data and classifiers may impede performance on more diverse or unique datasets. In addition, these models often produce and are evaluated on the quality of novel system-generated sentences [13, 14, 18, 9]. Automated metrics of quality such as BLEU [20] or ROUGE [16] have been proposed for evaluation of this task, but have been found to correlate poorly with human judgements of quality [11, 13]. This makes a direct, large scale comparison between these models on novel sentences difficult.

Recently in [11], we proposed to frame image description as the task of ranking a large pool of human provided captions. Unlike generation-based approaches [13, 14, 18, 9], this isolates the primary problem of judging the semantic accuracy of captions associated with an image from the secondary issues of measuring the linguistic quality of automatically produced sentences. Given an unseen image, we propose evaluating models by scoring a pool of unseen captions, where at least one is a known accurate caption for the image. This allows to us to provide objective quantitative judgments of quality that do not need to rely on human judgements.

The models of [11] utilize Kernel Canonical Correlation Analysis (KCCA) [1, 10] to induce a common space for images and descriptions of images. However, KCCA requires explicitly storing multiple kernel matrices corresponding to the pairwise computation of a kernel function across all training examples (in addition to learned weights of the same dimensionality), causing memory requirements and necessary kernel function computations to grow quadratically with the size of the training data. As increasingly large datasets of images and descriptions become available, it is important to develop models that scale to such; with KCCA, it may quickly become infeasible. In addition, since KCCA works in the implicit space defined by the kernels, it can make qualitative performance analysis difficult. Therefore, we sought to analyze a model that will scale better and is similar to what has been used before to associate images and text. Here, we evaluate the performance of a model based on Grangier *et al*.'s PAMIR [7], which has had success with retrieving images from tag-based queries, and the RankSVM of Joachims [12]. This model operates in the primal space, allowing for the memory requirements to depend approximately linearly on the number of examples rather than quadratically. Furthermore, by using a model that has been used with tags, we can highlight some of the unique issues when dealing with sentences and therefore the need to develop specialized models for image description.

2. Dataset

For this task, we have started with the Flickr 8K dataset of Rashtchian *et al*. [21], which contains 8,100 images from Flickr that were each annotated with five English single sentence captions. These images focus on actions being performed by people (or animals). For some of our experiments, we have further augmented this dataset with approximately 15,000 Creative Commons licensed Flickr images. These images are of a similar domain, as they focused on collecting depictions of people performing a variety of actions. Using Amazon Mechanical Turk accessed through CrowdFlower.com, these images were annotated with five English language captions following similar guidelines to the Flickr 8K dataset.

Figure 1 shows an example of an image and its associated captions. The associated captions describe literally

A boy bites hard into a treat while he sits outside.

The boy eats his food outside at the table

A child biting into a baked good

A small boy putting something in his mouth with both hands

The boy is eating pizza over a tin dish

Figure 1. An example from our dataset

what is being depicted in the image with minimal speculation of related information. It is important to notice that although all five captions describe the same image, the language being used and the information being conveyed differs. For example, only two of the annotators make reference to the (highly visual) fact that the image takes place outside, and what the boy is eating is referred to as "food", "pizza", or just "something".

3. The Model

As in [11], for every image in the dataset, there is a set of accurate captions that should be preferred among a larger set of all possible captions to retrieve. Our approach is to treat the problem of learning these preferences as solving the RankSVM optimization problem, as proposed by Joachims [12]. In order to encode such preferences, we explicitly train a linear classifier to have a stronger response to a relevant caption than other captions for a given image. The set of training examples we train our linear classifier on, D_{train}, is a set of triples of the form (i, c^+, c^-), where i is an image of the dataset, c^+ is a caption that is relevant for the image, and c^- is a caption that does not fit the image. Since we only care that c^+ is ranked higher than c^- for i, the loss we want to consider, $\ell((i, c^+, c^-), \mathbf{w})$, is the hinge loss of applying the current model weights, $\mathbf{w}$, to the difference between the representation of c^+ and c^- when paired with image i:

$$\ell((i, c^+, c^-), \mathbf{w}) = \max(0, 1 - \langle \mathbf{w}, \Phi(i, c^+) - \Phi(i, c^-) \rangle)$$

The objective we seek to minimize (based on [22]) is:

$$\min_{\mathbf{w}} \frac{\lambda}{2}||\mathbf{w}||^2 + \frac{1}{|D_{train}|} \sum_{(i,c^+,c^-) \in D_{train}} \ell((i, c^+, c^-), \mathbf{w})$$

where the parameter λ balances minimizing the loss on the training data with regularization. Explicitly modeling the representation of the entries of D_{train} can become intractable as the size of the dataset grows, and calculating the loss of all pairs in D_{train} can be quickly become computationally prohibitive. However, by training this SVM through online updates in the primal space, we can avoid the potentially intractable memory and computation costs. Therefore, we use a modified version of the Sofia-ml toolkit [22]'s Pegasos [23] implementation to minimize the objective. The modifications allow for storing the image and caption indices that define D_{train} separately from their respective feature representations. This significantly cuts down on redundancy in memory (at the expense of some computational efficiency). Pegasos is an iterative approach that involves sampling an image and a caption of each type, and then taking a local gradient step to minimize the objective based on the sampled triplet similar to stochastic gradient decent.[1] Furthermore, after each iteration the weight vector is projected to bound the maximum L2 norm of the weights during training, with provides theoretical convergence time guarantees.

Although designed for image search based on tags rather than English sentences, the framing of the objective is nearly identical to PAMIR [7], except that PAMIR is trained as a perceptron which results in a different form of regularization and Grangier *et al*. explicitly trained on all possible positive and negative pairs each iteration.

3.1. Feature representation

As a basic model, as with PAMIR [7], we define our feature representation for an image-caption pair, $\Phi(i, c)$, to be the outer product of the independent image and caption features: $\Phi(i, c)_{m,n} = \Phi(i)_m \Phi(c)_n$. By decomposing the feature space into separate image and caption features, storing the explicit representation of every used pair is not needed, and instead just the representation of the images and captions separately, at the expense of some computational time.

3.2. Image Features

In order to scale to large datasets, we wanted to use image features that are efficient to compute and require a small amount of memory per image, while still being expressive enough to serve as a starting point for image description. Therefore, we use the binary meta-class features of Bergamo and Torresani [3] which were designed for object classification. When combined with a linear classifier, the meta-class features were found to be state of the art on the the Caltech256 benchmark [8] and were competitive on the 2010 ImageNet Challenge [2]. Each image is represented as a binary feature vector, in which each bit corresponds to the output of a pre-trained variant of the LP-β classifier [6]. The original LP-β classifier relies on non-linear classification through kernels and has shown state-of-the-art results on multiple image categorization benchmarks. Bergamo and Torresani approximate the kernels of LP-β through Vedaldi and Zisserman's "lifting" method [25], re-

[1] Typically Pegasos samples k training examples at a time and updates based on the gradient involving all those examples. However Shalev-Shwartz *et al*. [23] found performance to be roughly constant for a fixed value of kT where T is the number of iterations on their examined tasks

sulting in speedups of several orders of magnitude during training. They take images from 8,000 randomly sampled synsets of ImageNet [4] and train two types of classifiers for meta-class representation. Then, they train one-vs-all classifiers on these synsets, and also classifiers that partition the synsets into a tree-structure. This results in a final dimensionality of 15,232 bits.

3.3. Text features

As a baseline representation, we simply define a binary bag-of-words representation of each caption c, where $\Phi(c)_w = 1$ if word w appears in caption c and 0 otherwise. We started with a binary representation since words are often not repeated in a caption. Naively, when a word such as "man" is repeated, it is unclear if the appropriate representation should be twice as much "man". The image could have two separate men that ideally should be modeled separately or the caption could just be referring to different aspects of the same man. We also apply IDF weighting to each of the text features, where if word w occurs in caption c, $\Phi(c)_w = \lambda_w = \log(\frac{|C_{train}|}{|C_{train}(w)|})$ where C_{train} is the set of captions associated with training images and $C_{train}(w)$ is the set of those captions that contain word w. Therefore, features corresponding to rarer words that are more discriminative for the dataset are up-weighted. In order to account for differences in sentence length, we also consider normalizing the final feature vector of each caption (with or without IDF weighting beforehand) to make its L2 norm equal to 1.

In [11], we found significant performance increases with incorporating multiple word sequences and similarity between different words into their kernel representation. However, since the kernelized representation avoids explicitly modeling the text representation, the complexity of the kernels can be increased without directly increasing the dimensionality of the representation or model complexity. In a more explicit setting, there would have to be an active feature corresponding to every possible word sequence of a caption, and incorporating the similarity of [11] would require an active feature for every sequence that has a non-zero similarity to a sequence of the caption, as well. It remains unclear how to include these features and have the model remain tractable in computation and memory in a more explicit setting.

4. Experiments

In order to provide quantitative results, we base our experiments on the experiments of [11]. We take the Flickr 8K dataset [21] and partition the dataset into 6,000 training images and 1,000 validation and test images. One arbitrary caption from each test image is chosen to create the pool of (unseen) candidate sentences for the test set. To evaluate the test set, the model ranks the quality of each caption for each test image and the model is evaluated based on how well the model retrieves the original captions for the test images. Although a caption can potentially describe multiple images in this dataset, this experiment considers a caption only "relevant" for an image if it was originally written for the image. However, in general, a better semantic model for this task should rank relevant captions higher than irrelevant captions. Recall-based metrics on the original caption were found to approximate the rankings based on human judgement of results in [11]. Since multiple captions in the data could apply to the same image, these metrics only approximate true recall performance, but were found to correlate well with human judgements of performance in [11].

Because the PAMIR based model can scale to training data of a size beyond what the KCCA models of [11] can handle, we consider two different data splits, the original split of [11] and one where the extra images were added to the training set, resulting in 21,757 training images.[2]. In addition, we vary the use of IDF weighting and L2 normalization on the text representation. The captions were lemmatized and stop words were removed. We restrict our vocabulary to words that appear in 20 or more sentences when using the training set of [11] and 50 or more sentences when we use the additional images. These cutoffs were chosen based on validation set performance and result in lexicons of 820 words and 1,320 words respectively. For every training image, the triplets in D_{train} are composed of all 5 original sentences written for an image of the training set as relevant captions paired with 2,000 random sentences from other training images as the irrelevant captions. The regularization parameter λ was chosen based on performance on the 1,000 image validation set, where each setting was trained 3 times and ran for 30 million iterations (chosen based on early validation performance).

Models for comparison As a baseline, we use a set of independently trained binary SVM classifiers for each term. Each SVM was trained using the LIBLINEAR toolkit [5], where the regularization parameter was chosen to maximize the area under the precision/recall curve (AUC) for each word on a validation set. During training, an image is considered to be a positive example for a word if the word appeared in any of the original captions. Similar to the jointly trained model, for every image and sentence pair, the final score is calculated by adding the real valued response of each word of the sentence's classifier. As a comparison to current state-of-the-art performance on this task, we also compare ourselves to the final model of [11].

[2]In addition, we also used additional extra images for validation purposes on held out data (for instance Figure 4) although they are not used for this quantitative experiment in order to better mirror the setup of [11]

5. Results

In Table 1, we present the results of the quantitative experiment aggregated over the 3 trained models. Since there is only one "relevant" caption for every image in the test set, we follow [11] and report the recall at 1, 5, and 10 and the median position of the caption (the point where recall is 50%). The independent baseline performs the worst across all metrics at this task, reinforcing the benefit of not treating each word as being independently classified. The best overall jointly trained model uses the extra training examples and IDF weighting but not L2 normalization. Although the increased amount of training data yielded some increase in these metrics, it is important to remember that the test set is drawn entirely from the domain of the original Flickr 8K dataset (in order to compare with [11]).

The final KCCA model of [11] outperforms all other models, but as mentioned above, it is unlikely to scale to increasingly large datasets and incorporates textual features beyond what is presented in this paper. It is also important to note that in addition to the more advanced linguistic modeling, the KCCA model of [11] uses different, lower level image features through a spatial pyramid kernel [15] incorporating SIFT [17], texture [24], and color features. It remains an open question what kind of performance may be achievable by state of the art kernels such as the ones used for the LP-β classifier [6].

Because the KCCA model of [11] operates in the implicit kernel space, it does not directly model the visual meaning of individual words. By contrast, our joint model directly learns weights for pairs of words and visual features. The model's overall response for an image-caption pair can be thought of as the sum of the responses to the active text features (in this case, words of the caption). In Figure 2, we examine the response to individual words of one of the reported "Joint IDF w/ Extra Training" models. We display the top responding images to each word from a set of 2,000 held out images.

Its important to remember that in this jointly trained setup, the model can rely on other commonly co-occurring words to disambiguate, and therefore the responses do not necessarily cleanly correspond to "classifications" of the term, unlike in the independently trained baseline. In the independently trained baseline, each term has roughly the same range of possible values in the output independent of how often it is needed to explain the preference of the caption in the training data. In Figure 3, we compare the response of an image and its original caption between a jointly trained model and the independent baseline. The single word "toss" is able to significantly (and incorrectly) hurt the overall value in the independent case.

5.1. Text representation analysis

An explicit model allows for directly observing the effects a certain textual representation can have on the results. This highlights some important characteristics of working directly with sentence descriptions.

L2 Normalization Although the captions can differ significantly in length, we were unable to find clear benefit to L2 normalizing the captions. The jointly trained model has stronger responses to words that more strongly influence the look of an image. Therefore, one would expect L2 normalization to cause the model to prefer shorter sentences that focus on more visually salient details of the images. However, a shorter sentence may not mention the background and other largely visual words and instead just describe the principal actor and action, and L2 normalization would effectively down weight the possible impact of that extra information in a longer sentence. Furthermore without L2 normalization, if a "relevant" caption is short, the largest possible negative response the model could report is less than a longer caption. And therefore, only having a few words can effectively bound the worst case position in the ranking task for that image query. This potentially has the effect of artificially inflating the performance metrics. In addition, by using only one caption from each test image, the caption pool may not be large enough to contain a significant number of both short and long captions that are relevant for a given image.

Differing information content Related to the issues of L2 normalization is the fact that annotators may not use the same amounts of detail to describe the same scene. This especially becomes an issue when the annotators fail to mention highly visual concepts such as "beach" or "street". For instance in Figure 4, the first caption does not mention "restaurant", "meal", or "dinner", words which have high positive responses according to our model. This makes it difficult for the first caption to be rated highly for the image. As a result, an ideal model might have to capture the potential scene even when it is not explicitly mentioned.

In the strict ranking evaluation framework, where only the original captions are considered relevant, recovering this missing information can be tricky. For instance, consider Figure 5, which shows two images of the dataset that involve bikes. One of the original captions for the image on the left is simply "A man on an orange bike" despite the image being a picture of a person jumping through the air with the bike and not a more typical action such as riding the bike down a street . We may also not know that this caption cannot apply to the biking image on the right, especially if we cannot accurately check the color of the bike in the image. One possibility to alleviate this problem is allowing for

Model	R@1	R@5	R@10	Median
Independent Baseline	4.1	13.2	20.3	51.0
Joint	5.4 ± 0.2	18.8 ± 0.5	26.4 ± 0.2	37.7 ± 2.3
Joint IDF	5.6 ± 0.2	17.9 ± 0.5	26.5 ± 0.2	39.5 ± 1.8
Joint w/ Extra Training	6.3 ± 0.2	18.8 ± 0.9	27.5 ± 0.6	38.5 ± 1.3
Joint IDF w/ Extra Training	6.8 ± 0.1	19.2 ± 0.3	28.7 ± 0.3	34.7 ± 1.5
Joint L2 w/ Extra Training	5.6 ± 0.2	17.8 ± 0.2	27.6 ± 0.2	37.5 ± 1.3
Joint L2 IDF w/ Extra Training	7.0 ± 0.6	18.7 ± 1.0	27.0 ± 1.4	36.7 ± 1.5
KCCA [11]	8.3	21.6	30.3	34.0

Table 1. Results for the quantitative ranking evaluation of searching for the relevant caption among candidates for a given image query

Word	Highest ranked images	Word	Highest ranked images
Crowd		Bicycle	
Table		Two	

Figure 2. An example of the top responding images to certain words of our lexicon among held out images for a jointly trained model (Joint IDF w/ Extra Training)

Joint IDF w/ Extra Training Model responses			
Overall	**0.35**	person	0.05
group	0.18	toss	-0.10
large	-0.03	yard	0.25

Independent Baseline Model responses			
Overall	**-0.37**	person	0.07
group	0.33	toss	-0.87
large	0.26	yard	-0.16

Figure 3. A comparison between the overall responses of a trained Joint IDF w/ Extra Training model and the independent baseline on a held out image paired with its original caption. In addition, the response for the models on the image paired with individual words are displayed. The values cannot be directly compared, but "toss" causes the performance to be worse for the baseline

A well lit room, with three glasses on the table and two plates.

Model responses					
Overall	**-0.85**	plate	-0.12	three	-0.45
glasses	0.04	room	0.02	two	-0.26
lit	-0.05	table	0.22	well	-0.25

A meal is on a table in a restaurant.

Model responses			
Overall	0.96	table	0.22
meal	0.39		
restaurant	0.34		

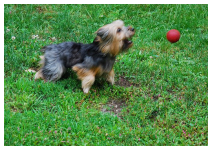
Figure 4. An example of a Joint IDF w/ Extra Training model's response for a held out image on two of the original captions (and constituent words) for that image

the addition of plausible "scene words" to the short caption. However, we may add "woods" or "street" to the caption, allowing for a greater (incorrect) response for the right image. In addition, it may be unclear from the training data that phrases such as "jumping" or "in the air" are plausible to be implied by the annotator rather than explicitly stated when describing pictures of people on bicycles.

Figure 5. Biking images from our dataset with different overall contexts

Visual Saliency Words in an image's captions do not necessarily correspond to a significant portion of the image.

A man is getting into a red car



A small dog tries to catch a red ball

A woman in red shoes walking in the street

Figure 6. "Red" images. The amount of red actually in the pictures varies significantly

This is especially apparent for adjectives, such as colors like "red" as shown in Figure 6. The red car takes up a significant portion of the left image, and therefore modeling "red" to describe the entire image is plausible. However in the image on the right, the redness of the shoes is salient enough to be mentioned, but it represents very few pixels of the image. This extends to the actors and objects as well, as the "shoes" themselves may not have been mentioned if the redness did not make them stand out. Unsurprisingly, certain words are more likely to influence the overall composition of an image and lend themselves as being more meaningful for a non-localized model. For example, words such as "lake" in Figure 7 and "crowd" and "table" in Figure 2 can strongly constrain the overall look of a photograph.

Duplicate mentions A caption may contain multiple words that describe the same concept in an image. Consider the caption "Three dogs play in a grassy field". For images with a "grassy field", the overall response in our model to both "grassy" and "field" are likely to be positive. As a result, the caption will be considered more favorable than if just "field" was mentioned, even though both "grassy" and "field" refer to the same concept in the image. For a more concrete example consider Figure 7 where the caption on the right is not completely accurate for the image while the the caption on the left is. The caption on the right is scored higher by a model because the annotator said "lake or river".

Determiners The word "two" has little meaning on its own, unless it is combined with a noun, i.e. "two skateboarders". As a result, to use these determiners correctly, a model would require the ability to count and localize objects. However, we still found that leaving these words in the text representation can potentially be beneficial. As shown in Figure 2, the use of the word "two" in the dataset is correlated with the kinds of images where an annotator would say there were two salient people in the image, which our model is able to capture to a certain degree. As a result these words improve performance for such images even if the model isn't explicitly "counting". This also has the effect of hurting performance when "two" is used differently, as in the case of the first caption of Figure 4. By listing the number of plates and glasses in the caption, the model incorrectly expects an image of a small group of people.

Context Consider the example in Figure 8 of pictures of people sleeping. In the left image, the fact that the person is asleep in a chair means that the person's position and the general layout of the image is closer to a "sitting" picture than the example of someone sleeping on a couch on the right. Modeling such concepts as "sleeping" as a single class to predict regardless of context (even with jointly learning the concepts) may not be expressive enough to be accurate. A related issue is the fact that annotators can often refer to the same object with different words. However, given enough training data, this may not be a significant issue. For instance, in Figure 7, as shown in the right sentence, both "lake" and "river" have strong responses to the image. It is also important to remember that by casting the task as ranking, our jointly trained models do not explicitly treat words that are not used in a caption as being unable to be applied to the image.

6. Conclusion

Associating images with English captions is a difficult task. In this paper, we examine a model that has been proposed for image tagging which can scale to large datasets. English sentences are not like the labels of typical image classification and annotation tasks, but models for image tagging should nevertheless be considered important baselines for this task. Unlike approaches such as KCCA, the model we examine here operates on an explicit text representation. The analysis of our experimental results reveals a number of linguistic issues that arise in this setting. In particular, captions may only provide partial descriptions of an image, and different captions convey different amounts of detail. Future models also need to address the problems of visual polysemy (i.e. the fact that the same word can have multiple visual interpretations) and visual salience (i.e. the fact that different parts of the sentence may correspond to smaller or larger portions of an image).

7. Acknowledgements

This research is supported by NSF grants 0803603, 1053856 and CNS-1205627. We would like to thank Peter Young for help in collecting the dataset used in this paper.

A little boy at a lake watching a duck

Model responses			
Overall	**1.20**	lake	0.89
boy	0.13	little	-0.08
duck	0.17	watch	0.09

A man standing on a deck above a lake or river

Model responses			
Overall	**1.81**	man	-0.03
deck	0.17	river	0.70
lake	0.89	stand	0.09

Figure 7. A Joint IDF w/ Extra Training model's responses to an held out image to two different captions (and their constituent words). The left caption is an originally paired caption. By saying "river" and "lake" the incorrect caption is considered better by the model

A man asleep in a chair in front of a full bookshelf

A woman in a red shirt is sleeping on a tan couch.

Figure 8. Sleeping images. Depending on the context, a person sleeping can look very different

References

[1] F. R. Bach and M. I. Jordan. Kernel independent component analysis. *JMLR*, 3:1–48, 2002.

[2] A. Berg, J. Deng, and L. Fei-Fei. Imagenet large scale visual recognition challenge 2010 http://www.image-net.org/challenges/lsvrc/2010/, 2010.

[3] A. Bergamo and L. Torresani. Meta-class features for large-scale object categorization on a budget. In *CVPR*, 2012.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009.

[5] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *JMLR*, 9:1871–1874, 2008.

[6] P. V. Gehler and S. Nowozin. On feature combination for multiclass object classification. In *ICCV*, 2009.

[7] D. Grangier and S. Bengio. A discriminative kernel-based approach to rank images from text queries. *PAMI*, 30:1371–1384, August 2008.

[8] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. 2007.

[9] A. Gupta, Y. Verma, and C. V. Jawahar. Choosing linguistics over vision to describe images. In *AAAI*, 2012.

[10] D. R. Hardoon, S. R. Szedmak, and J. R. Shawe-taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural Comput.*, 16:2639–2664, December 2004.

[11] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: data, models and evaluation metrics. *In Submission*.

[12] T. Joachims. Optimizing search engines using clickthrough data. In *KDD*, pages 133–142, 2002.

[13] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby talk: Understanding and generating simple image descriptions. In *CVPR*, 2011.

[14] P. Kuznetsova, V. Ordonez, A. Berg, T. Berg, and Y. Choi. Collective generation of natural image descriptions. In *ACL*, 2012.

[15] S. Lazebnik, C. Schmid, and J. Ponce. Spatial pyramid matching. In B. S. S. Dickinson, A. Leonardis and M. Tarr, editors, *Object Categorization: Computer and Human Vision Perspectives*, chapter 21, pages 401–415. Cambridge University Press, 2009.

[16] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, 2004.

[17] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, November 2004.

[18] M. Mitchell, J. Dodge, A. Goyal, K. Yamaguchi, K. Stratos, X. Han, A. Mensch, A. Berg, T. Berg, and H. Daume III. Midge: Generating image descriptions from computer vision detections. In *EACL*, 2012.

[19] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*. 2011.

[20] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002.

[21] C. Rashtchian, P. Young, M. Hodosh, and J. Hockenmaier. Collecting image annotations using Amazon's Mechanical Turk. In *NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, 2010.

[22] D. Sculley. Large scale learning to rank. In *NIPS 2009 Workshop on Advances in Ranking*, 2009.

[23] S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal estimated sub-gradient solver for svm. In *ICML*, 2007.

[24] M. Varma and A. Zisserman. A statistical approach to texture classification from single images. *IJCV*, 62:61–81, April 2005.

[25] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *PAMI*, 34(3), 2011.