Image Captioning with Pragmatics

Nico Chaves
Electrical Engineering
Stanford University

Noam Weinberger
Electrical Engineering
Stanford University

Reuben Cohn-Gordon
Linguistics
Stanford University

3/21/2017

Abstract

We replicate the approach of two papers ([9],[7]), which enrich neural image captioning systems to produce unambiguous captions for a target image in the presence of a "distractor" image. Broadly, this task amounts to a version of the Bayesian pragmatic RSA model ([2]) built on top of a neural speaker. We explore both modular and end-to-end implementations of this task, on the MSCOCO dataset.

1 Discriminative Image Captioning

Advances in deep learning have led to the success of end-to-end image captioning, through the encoder-decoder (CNN-RNN) architecture.

This task requires the neural architecture to achieve a degree of semantic competence - recognizing objects in an image and expressing this information in the form of a natural language sentence. More formally, the model must learn a distribution S_0=P(C|I), where C represents a caption and I an image.

However, this approach does not apply pragmatic reasoning, which humans can employ naturally. When using language and designing utterances, we try to model the reasoning of the agents with whom we are interacting.

In the context of image captioning, an example of pragmatic behavior is unambiguous captioning. Figure (1) features two girls sitting at a table. If our goal is to signal to a listener that we are referring to the girl on the left, then captions (1) and (2) are better than caption (3). Captions (1) and (2) are unambiguous while caption (3) may refer to either of the girls in the image.

Figure 1: Discriminative captioning example. The green dot indicates the target object, and the rest of the image is considered to be the distractor in this case.

(1) A girl wearing glasses and a pink shirt.

(2) The young girl sitting on the left.

(3) A girl sitting at a table.

From a computational perspective, the key to this task is Bayesian reasoning. Given images I and I′, we use Bayes' rule to calculate L = P(Images|C), as a model of a listener L, where Images is a distribution over a set of images (e.g. I and I′). The aim is then to produce the caption which maximizes the probability of the target image I, under L.

This Bayesian approach can either be built on top of a pre-existing non-discriminative image captioning system, or incorporated as part of an end-to-end image captioning architecture. ([9]) takes the former approach, while ([7]) takes the latter.

We attempt to replicate both approaches, and explore a range of variations to the task.

2 Overview of the Task

2.1 Agnostic Discriminative Captioning

We first describe the approach of ([9]). The basis of this approach is to apply Bayesian probability to a "literal speaker" model S_0 in order to obtain a "pragmatic speaker" S_1, which chooses utterances in a context. Here, the context is a distractor (which takes the form of another image to which the generated caption should *not* refer). In theory, the Bayesian approach works for any number of distractors, but the algorithms used for unrolling the generated caption will assume a single distractor.

The literal speaker S_0 is an image captioning model P(C|I,CL), where I is an image and CL a label describing the category the image belongs to. This additional parameter CL is a minor variation to normal image captioning, which

allows ([9]) to explore a secondary task, justification. In particular, they are able to provide a justification for their generated caption.

We can use P(C|I,CL) and P(C|I′,CL′) (where I′ and L′ are the respective distractor image and label) in order to generate a new probability L=P(Images,Classes|C), where *Images* is a Bernoulli distribution over I and I′ and *Classes* is a categorical distribution over classes. We can define L, using Bayes' rule, as P(C|I,CL) / (P(C|I,CL) + P(C|I′,CL′)).

([9]) uses a variation of this formula, and defines L as follows:

$$\text{(4)} \quad \text{L} = \log \frac{P(C|I, CL)}{P(C|I, CL')}$$

We can then in turn define a pragmatic speaker S_1=P(C|I,I',CL,CL'), which is a weighted combination of L and S, defined in (5). The weight, λ, mediates how much the pragmatic model focuses on well formed English, and how much it focuses on being discriminative.

$$\text{(5)} \quad \lambda * S_0 + (1-\lambda) * \text{L}$$

The difficulty with this approach is that the support of P(C|I'CL) is huge (though finite, if we assume captions are of a finite length, as they are in our model). As such, it is necessary to approximate P(C|I,CL) and P(C|I′,CL′). For ([9]), this is achieved with a beam search, applied during the unrolling of the sentence.

Rather than just conditioning the distribution over captions on images, ([9]) conditions on both images and labels (or classes), i.e. the type of object that is being described. This allows for a second Bayesian task, which they refer to as justification: given a target *class* and distractor *class*, a speaker produces the most unambiguous caption for the target. This task is termed justification on account of the captions that are produced amounting to explanations of why the target class was produced over the distractor.

2.2 End-to-End Discriminative Captioning

The approach of ([9]) is designed so that a system can be trained on non-discriminative captions and then used for discriminative ones.

However, given the ubiquity of pragmatic reasoning in language ([2]), an appealing approach on theoretical linguistic grounds is to learn directly from data which clearly makes use of pragmatic reasoning.

This approach is pursued in ([7]), which trains a model end-to-end through a Bayesian computation.

This model assumes that the captions in the training data are themselves discriminative. As such, this model cannot be trained on captions of single images, which is what the MSCOCO dataset provides.

([7]) solves this problem by creating a new dataset called Google Refexp, of referring expressions. These are captions which humans produce to refer to a single region in a larger image.

The input to the end-to-end network consists of a target region, the image of which it is part, and a partial caption. The target and distractor are both fed through an encoder-decoder CNN-RNN model to produce two distributions over the next word. These distributions are then combined with the same Bayesian function as in (4).

3 The Data

Following ([7]), we make use of the Google Refexp Dataset, which is built on top of MSCOCO. While MSCOCO provides human-generated captions for a large collection of images, Refexp adds referring expressions for distinct bounding boxes, i.e. regions of the image. For the purposes of our models, we stretch these regions to the same size as full images (tensors of shape (224,224,3)), so that they can be accommodated by the same model.

We preprocess the Refexp dataset so that each training item consists of a partial caption and image as input X, and the next word of the caption, as output y.

Our vocabulary consists of all the words which appear in any of the MSCOCO captions (of which there are 28000).

For purposes of batching, we pad the partial captions with stop tokens, so that they are all of the same length, and mark the beginning and end of each caption with a start and stop token respectively.

4 The Architecture

4.1 The Basic Image Captioning Model

The architecture of our simple, non-pragmatic image captioning system is as follows. The model takes a a partial caption and image as input. We first use the pretrained 300-dimensional GloVe embeddings ([8]) to produce a sequence of vectors of length CAPTION_LENGTH, which we freeze during training. We then apply a dense layer to convert each timestep to size 512.

We then use the VGG 19 layer CNN, with pretrained weights (which we freeze at train time), and extract as a feature vector from the input image the final

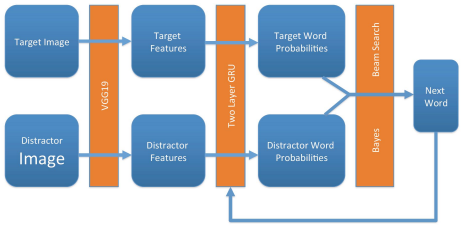

Figure 2: The model architecture.

dense layer of the network (before the classification layer). We repeat this image CAPTION_LENGTH times, and apply a dense layer of output size 512 at each timestep.

Similarly, we extract the final "prediction" layer of the VGG CNN, which we similarly repeat CAPTION_LENGTH times and apply a dense layer to at each timestep, this time of output size 64.

We then take each of these three sequences and concatenate them at each timestep, to obtain a tensor of shape (BATCH SIZE, CAPTION_LENGTH, 512)

Finally, we run a two layer GRU over this sequence, each with a hidden layer size of 512, and run the final output through a dense layer with a softmax activation to produce a distribution over V classes, where V is the size of the vocabulary.

The padded elements of the partial caption are masked, and dropout is used in dense layers of 0.5 and recurrent layers of 0.2. No other regularization is used. For optimization, we use RMSprop. We build our system in Keras (`https://keras.io/`) and use the Google Refexp toolkit to help preprocess our data (`https://github.com/mjhucla/Google_Refexp_toolbox`). Figure (2) summarizes the model architecture.

The result of this architecture is a model of P(C|I,CL), where I is a featurized image and CL a distribution over possible categories, provided by the VGGnet.

5 The Pragmatic Model

5.1 The Modular Approach (Non-End-to-End)

Using the basic captioning model described above, we then produce a pragmatic speaker S_1, with the formula in (5). This takes two images, a target and distractor, and for each time step of the unrolling (where a time step consists of the generation of the next word of the caption being generated), uses (5) to predict the next word.

Figure 3: Captioning example where the target and distractor images are similar (the target is on the left). In this example, a good caption must be discriminative. Our literal model's caption: "A giraffe that is standing in the grass". Our pragmatic model's caption: "Two giraffes standing near a fence with water in the background".

To perform this unrolling, we try both using a greedy strategy (picking the most likely word at each time step) as well as beam search. We observe the best results (with a reasonable prediction time) given a beam size of 10.

Since we train a literal speaker model that is conditioned on both an image vector and a class vector, we have the choice of supplying a distractor in which either one or both of these is different.

Changing both the image and the distractor results in a pragmatic speaker S_1 which generates unambiguous captions for a target relative to a distractor image and class. Changing just the class for the distractor produces an S_1 with the "justification" behavior explored by ([9]).

For the task of unambiguous captioning, where both class and image are changed in the distractor, we find that a λ of 0.5 produces the most successful captions. When λ is too low, the captions become incoherent, and when too high, the literal and pragmatic speaker modules produce very similar output.

6 Evaluation

A demonstration of the behavior of the agnostic discriminative system is shown in figure (3). As can be seen, the system generates the caption which is more informative when presented with a similar looking target and distractor.

Quantitative evaluation of image captioning, particularly when discriminative, is difficult, owing to the large range of good ways of captioning any given image. Furthermore, we faced difficulties in training our basic captioning model to the point where captions where it consistently produced high quality descriptions, and were not successful in optimizing the end-to-end model at all.

The most successful model, therefore, is the non-end-to-end pragmatic captioning model, as described in (2.1). We find that unrolling with beam search

Figure 4: Captioning example where the target and distractor images are not similar.

produces the best results, at a beam size of 10.

We find that when the distractor images are chosen at random (rather than being similar to the target images, or having similar captions to the target images), there is still a qualitative improvement in the captioning. This seems to reflect the fact that human speakers attempt to be informative in general language use. As an example, see figure (4: the target image on the left is literally captioned by (6 and pragmatically captioned by (7) in the presence of the distractor on the right. The pragmatic caption is more detailed, despite the unrelated nature of the distractor.

(6) A man is riding a wave on a wave.

(7) A man riding a wave on skis on a snow covered slope.

7 Conclusions and Further Work

These models demonstrate that it is possible to apply Bayesian models of pragmatics to complex machine learning tasks, in a way which produces qualitatively pragmatic behavior.

This raises the natural question of whether other varieties of pragmatic inference are possible in this setting. Models of metaphor ([3]), hyperbole ([4]) have both been shown to work in simple cases (i.e. with hand-produced data).

Relatedly, a number of other image-language tasks seem ripe for pragmatic enrichment. One such task is visual question answering. The rich Visual Genome dataset would provide a useful dataset for richer pragmatic tasks.

Another direction to pursue would be character based language modeling. Typos are common is the MSCOCO captions, and character based models have been successful in other language modeling tasks (e.g. ([5])).

A final direction to explore is the reverse task, where an image is generated according to a caption (using a GAN ([1]) or variational autoencoder ([6])). Here, the aim would be to produce an image given a target *caption* and distractor *caption*, which unambiguously depicted the target and not the distractor.

References

[1] Ian Goodfellow et al. "Generative adversarial nets". In: *Advances in neural information processing systems*. 2014, pp. 2672–2680.

[2] Noah D Goodman and Michael C Frank. "Pragmatic language interpretation as probabilistic inference". In: *Trends in Cognitive Sciences* 20.11 (2016), pp. 818–829.

[3] Justine T Kao, Leon Bergen, and Noah Goodman. "Formalizing the Pragmatics of Metaphor Understanding." In: *CogSci*. 2014.

[4] Justine T Kao et al. "Nonliteral understanding of number words". In: *Proceedings of the National Academy of Sciences* 111.33 (2014), pp. 12002–12007.

[5] Yoon Kim et al. "Character-aware neural language models". In: *arXiv preprint arXiv:1508.06615* (2015).

[6] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[7] Junhua Mao et al. "Generation and comprehension of unambiguous object descriptions". In: (2016), pp. 11–20.

[8] Jeffrey Pennington, Richard Socher, and Christopher D Manning. "Glove: Global Vectors for Word Representation." In: *EMNLP*. Vol. 14. 2014, pp. 1532–1543.

[9] Ramakrishna Vedantam et al. "Context-aware Captions from Context-agnostic Supervision". In: *arXiv preprint arXiv:1701.02870* (2017).