Objectifying Captions

Image Caption Generation Using Object Information

W266
Noah Randolph, Harry Xu
22 April 2018

Abstract

A potential enhancement to image caption generation, building on a recent innovation called OBJECT2TEXT (Yin & Ordonez, 2017) is explored. OBJECT2TEXT is an encoder that can be used to train models on images from the MS-COCO dataset (Lin, et al., 2014), which have objects within images labeled with rectangular extents and object categories (e.g. dog, car, table). This paper attempts to capture potentially useful information in the form of relative object areas and perimeters within images to improve image caption generation. Several models are created and results are analyzed on size and depth descriptions in the resulting generated captions. Evidence in the form of image captions from baseline and modified models shows a difference in the presence of size and depth related words. However broad word counts across all the models reveal the effect to be small and inconsistent.

Introduction

Image captioning presents a unique challenge in two fields: computer vision and natural language processing. It showcases the ability for the computer to understand the objects in an image while producing human readable descriptions. There have been many approaches to image captioning, with the latest successes building from the deep learning framework, where neural networks are utilized to understand images as well as produce captions. We investigate a recent paper, "OBJ2TEXT: Generating Visually Descriptive Language from Object Layouts" (Yin & Ordonez, 2017), which utilizes a sequence to sequence model to encode the objects and their locations in an image as an input sequence to an LSTM and decode the representation with an LSTM language model. We implement their framework while exploring additional modifications to improve upon the results. Specifically, we seek to increase the model's understanding of the images by inputting additional object attributes such as the notion of object size.

We propose that by adding these features, our language model will be able to produce more meaningful captions with object relations better known.

Background

The paper by Yin & Ordonez, *OBJ2TEXT*, utilizes a related work, *You Only Look Once* (YOLO)*: Unified, Real-Time Object Detection* (Redmon & Farhadi, 2017). YOLO is a new approach to object detection which encapsulates the entire image detection task into one convolutional neural network. It treats object detection as a single regression problem and uses a CNN to train and predict the objects in an image's bounding box as well as classify the object type. YOLO's improvement on the previous image recognition frameworks is increased speed that can be offered by only having one CNN and which also allows for real time object detection.

OBJ2TEXT is combined with YOLO to encode objects for use in a natural language system. Using object locations in the images as well as the number of each category as features, OBJ2TEXT-YOLO is combined with an image caption generation model, based on Neuraltalk2 (K., 2016), that combines a convolutional neural net for image classification whose information is encoded into a vector for input into a recurrent neural network language model. OBJECT2TEXT-YOLO is an encoder that can be used to train models on images from the MS-COCO dataset (Lin, et al., 2014). MS-COCO dataset image annotations include objects labeled from a selection of categories (e.g. dog, car, table). The objects are identified in the images through two means, one of which is segmentation and the other using rectangular extents. OBJECT2TEXT-YOLO does not use the objects labeled in the MS-COCO dataset for training, and instead encodes them on its own (using YOLO) prior to training on MS-COCO image captions.
The idea behind encoding object information when you already have all of the image pixel information follows previous work using clipart scenes (Zitnick, et al., 2013), where the attempt is to separate pattern recognition (in pixels) from visual meaning.

Methods

Using 200,000 training iterations, the performance and caption results of the Location Encoder and category word embedding of each object in an image, with a max sampling decoder (temperature = 1.0), is used as a baseline. The baseline comes from Yin & Ordonez, 2017 and is briefly described below.

Both the encoder and decoder are trained according to Equation 1.

$$W^* = \underset{W}{\text{argmin}} \sum_{n=1}^{N} -\text{log}p(\boldsymbol{s}^n | h_L^n, W_2) \qquad \text{(Eqn. 1)}$$

Where W includes both the encoder and decoder parameters, W_1 and W_2. N is the number of training observations. $\boldsymbol{s}^n$ is a target caption. h_L^n is the encoded object layout and category of image n at time $t = T_1$ (the end of the encoding step) to generate caption $\boldsymbol{s}$, computed as in Equation 2.

$$h_t^e = \text{LSTM}(h_{t-1}^e, x_t; W_1) \qquad \text{(Eqn. 2)}$$

Where e stands for encoder. The input x_t is defined as in Equation 3.

$$x_t = W_o o_t + (W_l l_t + b_l) \qquad \text{(Eqn. 3)}$$

Where W_o is a category word embedding matrix, the categories are represented in a one-hot vector in o_t. W_l and b_l comprise the parameters of the object location encoder, and l_t is the object location vector containing the x and y coordinates as well as the width and height of the object input at time t.

The generated caption $\boldsymbol{s}$ is determined according to Equation 4.

$$p(\boldsymbol{s}|h_L) = \prod_{t=1}^{T_2} \text{softmax}(W_h h_{t-1}^d + b_h) \qquad \text{(Eqn. 4)}$$

Where d stands for decoder and $W_h h_{t-1}^d + b_h$ computes a vector of the hidden states of the decoder LSTM. h_t^d is computed as in Equation 5.

$$h_t^d = \text{LSTM}(h_{t-1}, W_s s_t; W_2) \qquad \text{(Eqn. 5)}$$

Where W_s is a category embedding matrix for caption sequence of symbols.

With the baseline described as in the equations above, our modifications include altering the location configuration vector l_t by replacing the object width and height dimensions with object area and perimeter. The model is shown in Figure 1, below.

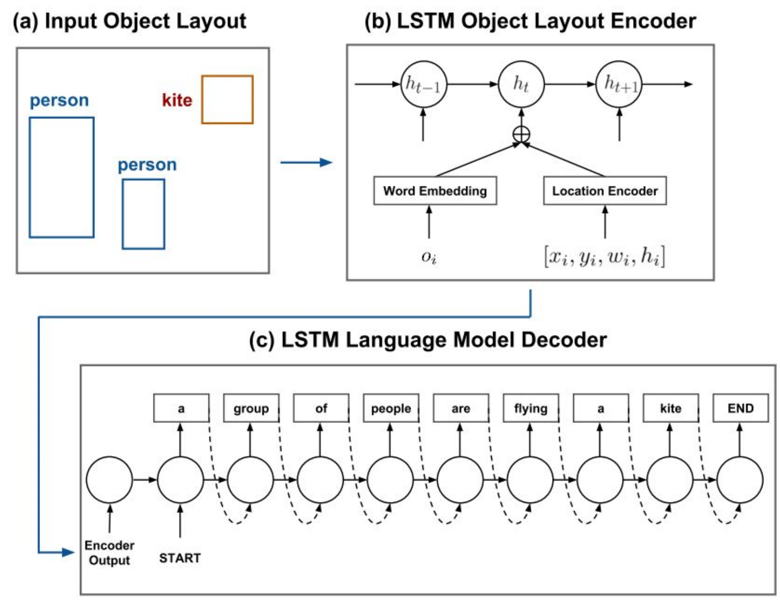

Figure 1. Model layout from Yin & Ordonez, 2017 is shown above. The Location Encoder in box (b) is modified from containing width w_i and height h_i to containing area $w_i * h_i$ and perimeter $2 * (w_i + h_i)$.

The modified Location Encoder that includes area (width x height) of the object bounding boxes is then also trained with 200,000 iterations to be compared against the baseline both qualitatively and with the Bleu-4 score.

Further analysis for comparison is made by including the Neuraltalk2 image caption generator word embeddings (K., 2016), which is the more familiar pixel-based pattern recognizing CNN-RNN caption generator, along with the object embeddings in OBJ2TEXT-YOLO. The full model is trained as another baseline to compare against another full model that includes the aforementioned Location Encoder modified to include area and perimeter.

Lastly, the baseline and modified models (objects only, no CNN derived word embeddings) are altered to include a beam search in the decoding RNN. Width 5 is used according to custom as well as to keep GPU time to a minimum.

For all models above, in addition to score metrics, qualitative assessments of size and orientation word counts (e.g. large, small, back, front) in the generated captions are computed to observe characteristic changes in captions generated between the models, in order to assess whether depth and size information is generated as predicted. Figure 2, below, is a summary of all the model configurations that were tested and will be discussed in the next session.

YOLO object position code

MODELS ⇩	$\underbrace{346}_{x}\ \underbrace{234}_{y}\ \underbrace{989}_{w}\ \underbrace{456}_{h}$	$\underbrace{346}_{x}\ \underbrace{234}_{y}\ \underbrace{989}_{\text{perim.}}\ \underbrace{456}_{\text{area}}$	$\underbrace{346}_{x}\ \underbrace{234}_{y}\ \underbrace{989}_{w}\ \underbrace{456}_{\text{area}}$	$\underbrace{346}_{x}\ \underbrace{234}_{y}\ \underbrace{989}_{w}\ \underbrace{456}_{\text{perim.}}$
objects only	✓	✓	✓	✓
CNN + objects	✓		✓	
objects only + beam search	✓	✓		

Figure 2. Summary of the model variations tested. The first column contains the baselines, where the YOLO-generated object positions are not modified.

Results and Discussion

Initial results were very promising, as the image in Figure 3, below, was quickly found to show evidence of the position code modifications having an effect. You can see how the modified object embedding (using area) may have caused the difference in size of the elephants in the photo's generated caption.

Training captions: ● Two elephant walking, the one in the foreground kicking up a some dirt. ● Two elephants walk around in a large grassy field. ● two elephants with long walking through the grass ● An elephant kicking some dirt on the ground. ● two very big elephants walking in the wild
Objects-only generated baseline caption: ● two elephants standing in a field with trees in the background
Objects-only generated caption with position code including area & width: ● a **baby** elephant standing next to a **larger** elephant (bolded words for emphasis)

Figure 3. Initial promising results showing potential effect of object position code modifications, reflected in size words present in generated caption.

Observing the size-and-depth-related word counts between the baseline and different variations of area and width in the object position code leads to much less conclusive results, as seen below in Figure 4. Area and perimeter models dominate in counts of the words ‘background,’ ‘large,’ ‘back,’ ‘short,’ and ‘tall,’ especially the words ‘back’ and ‘background.’ However, the baseline dominates the counts of ‘front’ and ‘short.’

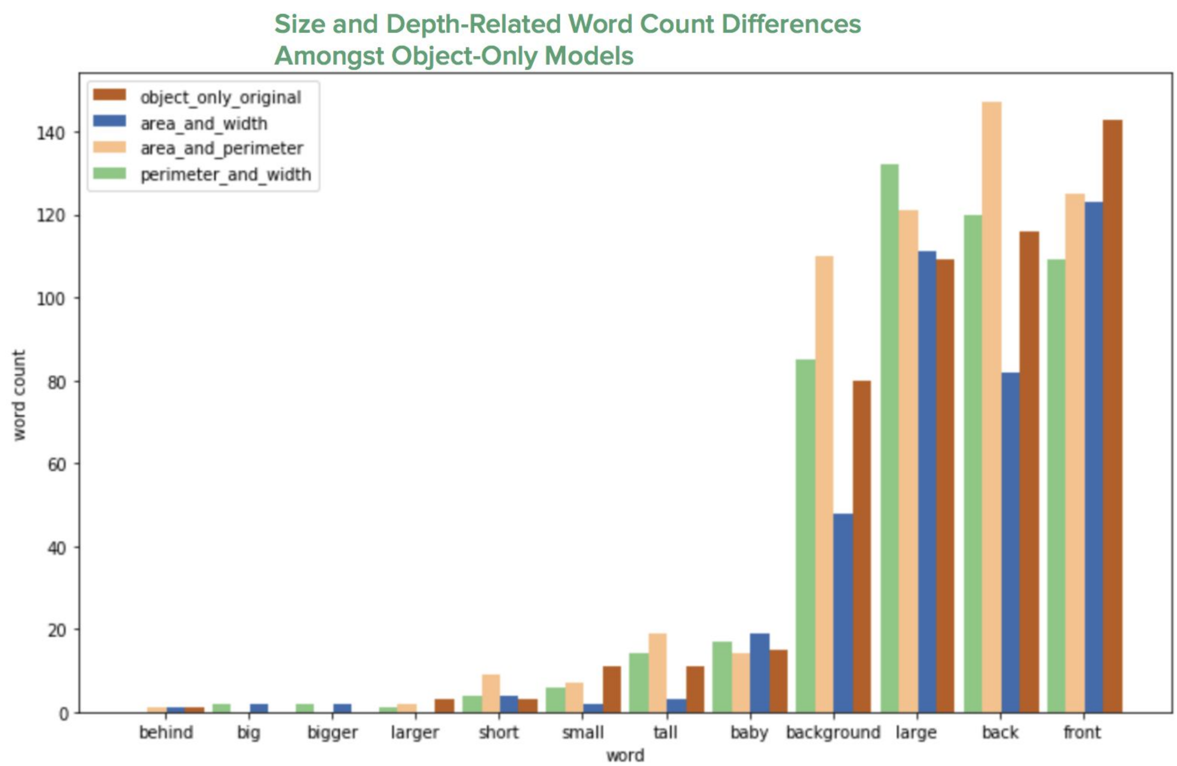

Figure 4. Word counts of size-and-depth-related words between the baseline (red) and the modified object-only models (yellow, blue, and green) reveal mixed results.

Given the mixed results of the models above, two more variations of the model were implemented, one of which included adding the CNN-RNN Neuraltalk2 image caption generator word embeddings (K., 2016). The other variation involved changing the decoder from max sampling (temperature = 1.0) to a beam search decoder with beam width k = 5. Results of the two variations are shown in Figure 5 and Figure 6, below. The size-and-depth-related word counts are even less conclusive than the original object-only results. In fact, the CNN-RNN Neuraltalk2 inclusion seemed to reverse the effect of the width and area in the object position code (K., 2016). Both models in Figures 5 and 6 narrow the gap between word count differences amongst the baseline and modified models when compared to the models in Figure 4.

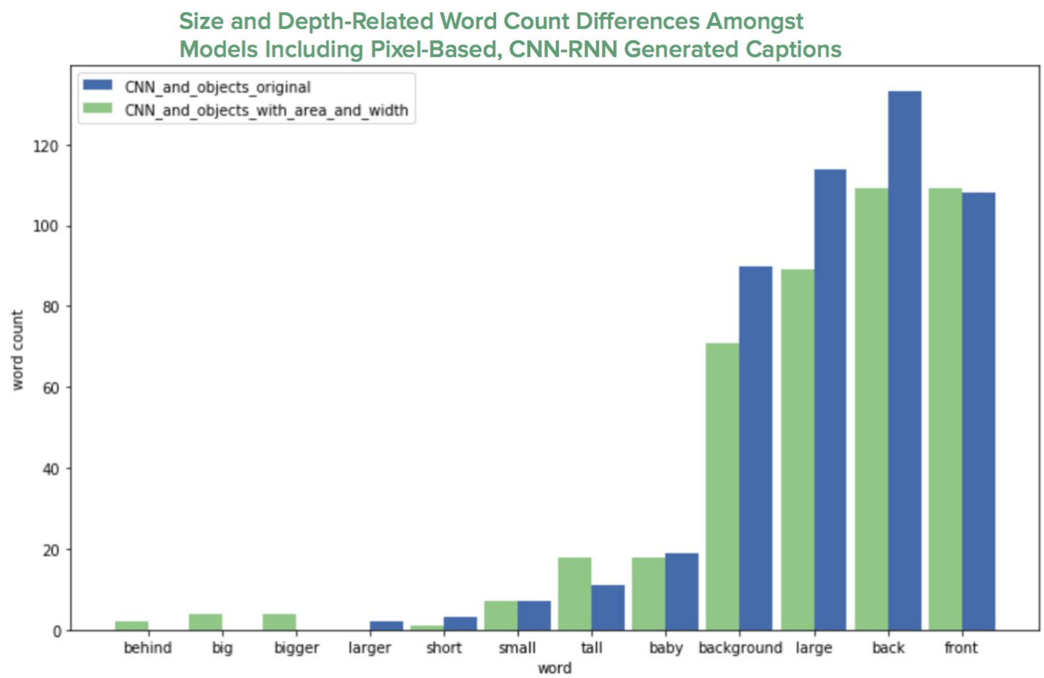

Figure 5. This plot shows the baseline and modified object position code models with included CNN-RNN NeuralTalk2 word embeddings (K., 2016).

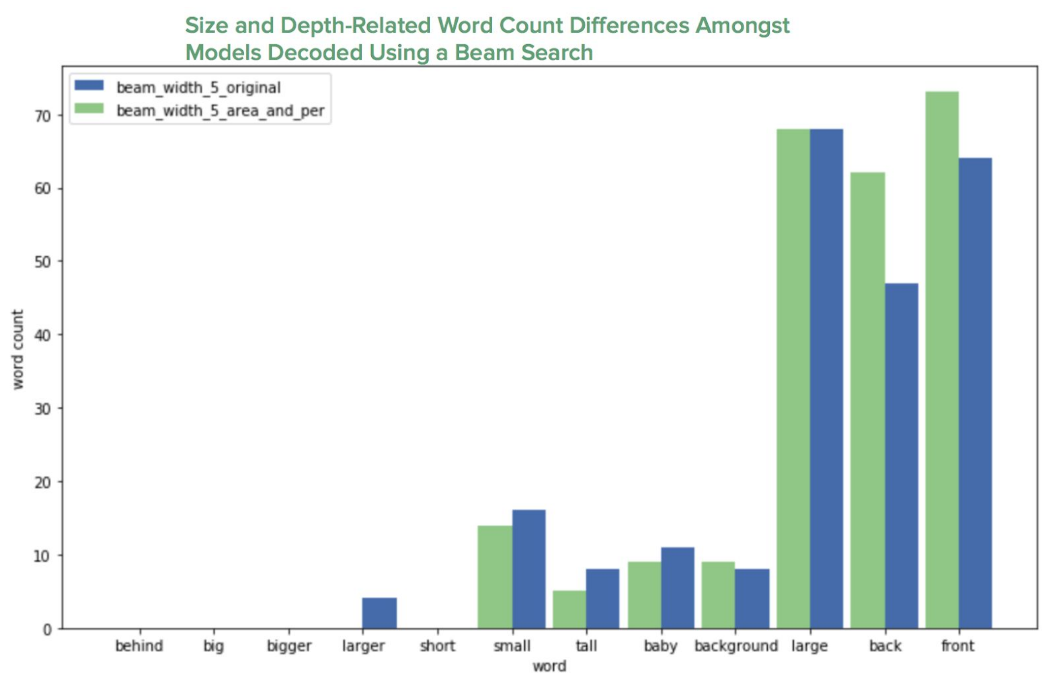

Figure 6. This plot shows the objects-only baseline and modified object position code models with a decoder beam search of width k = 5.

Below, we look in detail at several captions generated by the models described thus far. First, the five ground-truth captions from the COCO dataset are shown, followed by captions generated by each aforementioned model.

Figure 7. Image ID 562805

Ground-truth captions:

1. A modern transportation building with busses lined up for passengers.
2. A bus station with buses and people nearby.
3. Buses parked at a bus stop while unloading passengers.
4. A bus depot with buses parked in front.
5. some people buildings and three different buses and a tree

Generated captions:

	original, unmodified object position code	**with modified object position code**
max search decoder	a group of people standing on a street next to a bus	a bus is driving down the street with a bus **behind** it
beam search decoder	a group of people standing in a parking lot	a couple of buses that are sitting in the street
including CNN-RNN	a group of people standing on a city street	a bus is driving down the street with people walking

For the captions generated by the Figure 7, only one model (in the top right cell) included a size or depth related word to describe the relation between two similar objects. The top left cell includes the positional phrase ‘next to,’ but it’s referring to two different objects (people and bus). Since two different objects can have different sizes, it’s unlikely for ‘next to’ in this case to be related to the object position code (which happens to be unmodified). It would only make sense for the sizes or positions of two similar objects (that have presumably the same real world size, like two buses) to be affected by the object position code.

Figure 8. Image ID 198178

Ground-truth captions:

1. A large elephant standing next to a small elephant.
2. An adult and baby elephant walking beside each other.
3. Adult elephant with young walking in grassy area.
4. And elephant walks with its baby elephant.
5. A big and a small elephant out in the sun

Generated captions:

	original, unmodified object position code	**with modified object position code**
max search decoder	a large elephant standing next to a tree	a **large** elephant standing next to a **baby** elephant
beam search decoder	an elephant standing in the middle of a field	a **large** elephant standing next to a **baby** elephant
including CNN-RNN	a couple of elephants standing next to each other	a **baby** elephant standing next to a **bigger** elephant

The iconic image of two elephants in Figure 8 results in captions with modified object position codes clearly expressing size relationships between the elephants. Contrast that to the captions generated by the original object position code, where there are no relational words to describe size or depth differences between elephants. The top left cell's caption does include the word 'large' but it is not in relation to another elephant.

Figure 9. Image ID 322307

Ground-truth captions:

1. A mama elephant standing next to a baby elephant in a cage at a zoo.
2. An elephant in a cage with its baby.
3. a baby elephant and a large elephant standing near one another

4. A ELEPHANT IS WALKING NEXT TO ITS BABY CUTIE!
5. The baby elephant stays close to its mother.

Generated captions:

	original, unmodified object position code	**with modified object position code**
max search decoder	two elephants are standing in a field of grass	a **baby** elephant standing next to a **larger** elephant
beam search decoder	a couple of elephants standing next to each other	a **large** elephant standing next to a **baby** elephant
including CNN-RNN	a **baby** elephant is standing in front of a **larger** elephant	a **baby** elephant standing next to a **bigger** elephant

Note that there are many elephant photos in the COCO dataset. The captions generated by Figure 9, as in Figure 8, include size relations between the elephant objects in the image. However, this time the model with the original object position code that includes the NeuralTalk2 CNN-RNN caught the size differences of the elephants as well.

Figure 10. Image ID 490860

Ground-truth captions:

1. A baby elephant panting on a white canvas.
2. A baby elephant is painting a picture with it's trunk.
3. An elephant is standing in the dirt drawing on an easel with its trunk.
4. An elephant touching a drawing on an easel with his trunk.
5. An elephant drawing a picture with it's trunk.

Generated captions:

	original, unmodified object position code	**with modified object position code**
max search decoder	a couple of elephants standing next to each other	a **large** elephant standing next to a **baby** elephant
beam search decoder	a couple of elephants standing next to each other	a **large** elephant standing next to a **baby** elephant
including CNN-RNN	a group of elephants standing in a field	a couple of elephants standing next to each other

Captions generated by Figure 10, as with the previous figures' generated captions,, show a tendency to include more size words with the modified object position code than with the original object position code.

While the above examples display evidence that the modified object position code may have the predicted effect of influencing size and depth relations in generated captions, the effect is mild and inconsistent. It is also not possible, with the image selection methods used, to rule out random chance in seeing generated captions as shown above. In fact, in order to find the images used as examples, search algorithms were used that actually lead to the results shown. More rigorous statistical analysis is needed to conclusively declare whether the modified object position code has the predicted effect (or to reject the null hypothesis that there is no effect of the modified object position code).

Finally, for completeness, the Bleu-4 scores are shown below, in Figure 11. Note that performance improvement of standard scoring metrics was not a goal of this project.

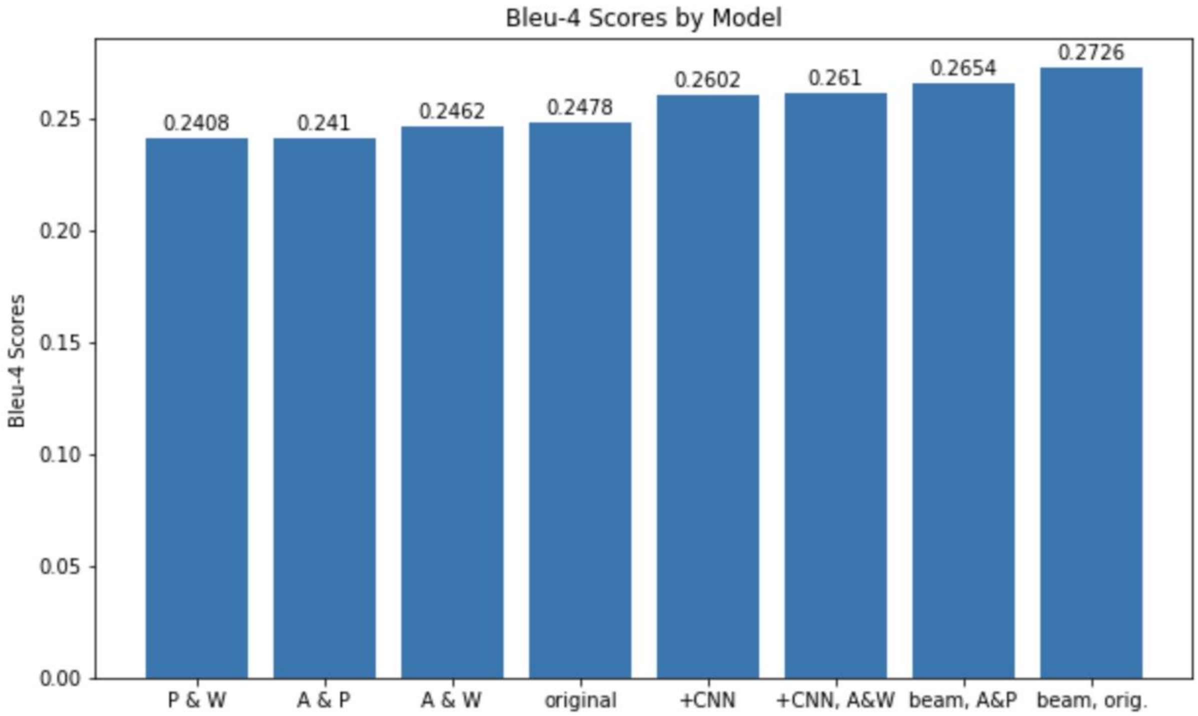

Figure 11. The performance standard (Bleu-4) scoring for each of the caption generating models. For the modified object position code models, “P” stands for perimeter, “W” stands for width, and “A” stands for area.

The results in Figure 11 are likely to be unrelated to the object position code. It is not surprising that the models including the CNN from NeuralTalk2 (K., 2016) are higher than the object only, max search decoded models, since the image pixels constitute many more features from which the model acquires information. Nor is it surprising that the models utilizing a beam search decoder outperform models with a max search decoder, since by definition there are more opportunities to pick the best sequence of words.

Conclusion

We proposed that by adding the area and perimeter of object bounding boxes to our language model, it would be able to produce more meaningful captions with object relations better known. While our attempts at adding the object attribute of size had little improvement in increasing the Bleu score of our model, we did find instances where encoding these attributes appeared to lead the model to be more descriptive in regard to the object size differences in images, as long as the objects being compared in the caption were of the same category. We have some evidence that our attribute addition also allowed for our model to have a better sense of depth and relation within the image of the objects. Implementing OBJ2TEXT and training on the COCO dataset was challenging and required twelve hours per 200,000 images utilizing four GPUs. Given the computational expense of running OBJ2TEXT, we were limited in only exploring a few model additions. Given more time, we would perform a rigorous statistical analysis of the results to separate actual effect from random chance.

References

Karpathy, A., & Fei-Fei, L. (2015, 06). Deep visual-semantic alignments for generating image descriptions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. doi:10.1109/cvpr.2015.7298932

K. (2016, September 23). Karpathy/neuraltalk2. Retrieved from https://github.com/karpathy/neuraltalk2/

Lin, T., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., . . . Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. *Computer Vision – ECCV 2014 Lecture Notes in Computer Science,* 740-755. doi:10.1007/978-3-319-10602-1_48

Redmon, J., & Farhadi, A. (2017, 07). YOLO9000: Better, Faster, Stronger. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. doi:10.1109/cvpr.2017.690

T. (2018, January 20). Tensorflow/models. Retrieved from https://github.com/tensorflow/models/tree/master/research/im2txt

Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2017, 04). Show and Tell: Lessons Learned from the 2015 MSCOCO Image Captioning Challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(4), 652-663. doi:10.1109/tpami.2016.2587640

Yin, X., & Ordonez, V. (2017). Obj2Text: Generating Visually Descriptive Language from Object Layouts. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. doi:10.18653/v1/d17-1017

Zitnick, C. L., Parikh, D., & Vanderwende, L. (2013, 12). Learning the Visual Interpretation of Sentences. *2013 IEEE International Conference on Computer Vision*. doi:10.1109/iccv.2013.211