Explaining Image Captioning Models Through Attention Maps, Image Perturbations, and Object Importance Maps

Alejandro Fosado
University of California, San Diego
afosado@ucsd.edu

Yuexiang Zhang
University of California, San Diego
yuz719@ucsd.edu

Jordan Levy
University of California, San Diego
jdlevy@ucsd.edu

Abstract

Image captioning models are complex because they work on object detection as well as caption generation. When these models fail it is hard to understand where and why they fail. To explain how an image captioning model works, we use attention maps to visualize the relationships between generated words and objects in an image. Moreover, we utilize an image perturbation model to alter regions of images to see how the captions change and to test the robustness of our model by measuring the similarity between captions generated before and after the altering of the image.

1. Introduction

Image captioning is an interesting area to investigate because it is a combination of object detection and natural language processing, which corresponds to the application of convolutional neural network (CNN) and recurrent neural network (RNN). A good image captioning model mimics the action of a human that it is able to understand and describe what an image includes. In the field of deep learning, attention maps are a widely used technique. With attention maps, the model can learn which parts in the image it needs to pay more attention to when generating the next word in the caption. In our project, we implemented attention maps as a tool to understand why the model made the predictions it did. At the same time, it is important that an image captioning model is robust, which means it could make reasonable adjustments to the caption it generates when the input image is changed. In our project, we also implemented an image perturbation model that alters the input image to evaluate the robustness of our image captioning model. We used information from the image perturbation model along with the attention maps to evaluate each objects' importance to the original caption generated from the model.

2. Data

At the very beginning, we wanted to train the image captioning model with the combination of the COCO dataset and the Visual Genome dataset. After we realized it is hard to combine them and we had little use for the relational data in the Visual Genome dataset, we decided to only use the COCO dataset to train this model. The reason why we use the COCO dataset in the end is that COCO is large and comprehensive enough to train our model. It contains 330k images and 5 captions per image. We use the CelebA (CelebFaces Attributes Dataset) dataset to train our image perturbation model. CelebA is a large-scale face attributes dataset with more than 200K celebrity images, each with 40 attribute annotations. The images in this dataset cover large pose variations and background clutter.

3. Methods

This model can be separated into three parts: encoding, decoding, and attention. The first part, encoding, we use a CNN model, which takes an image as input and outputs the multiple learned channels that encode the features of the image. The second part, decoding, we part use an RNN model, which uses the encoded data from the CNN as a starting point to generate a sequence of words. Lastly, we use attention during the decoding process allows the RNN model to learn where it should pay more attention when generating the next words. We can later use this learned attention to generate attention maps to add interpretability to our model.

In the image perturbation process, our model chooses an object from the raw image, creates a mask to indicate where the object in located, then outputs a new image in which the object it removed through generative inpainting.

After integrating the image captioning and image perturbation model, we generated a caption from the raw image and a new caption for each object we removed from the raw image. Then we calculated the similarity between the original and counterfactual image. To calculate the similarity, we embedded these two captions into vectors with a pre-trained BERT model and then calculated the cosine sentence distance between them. We expected to see that the more important the removed object is, the larger distance we get or similarly the more the caption changes.

3.1. Attention Map

In order to better understand the caption generation process of a CNN-RNN image captioning model, we generated attention maps for each word generated by our model. We used deterministic "soft" attention as the research paper Show, Attend, Tell does. This differs from stochastic "hard" attention as during the learning process we do not sample attention locations each time but can instead take the expectation of a context vector. A benefit of using deterministic attention is that the whole model is smooth and differentiable so we can use standard back-propagation to train the model.

Using these attention maps we can see where the model is looking when it outputs a given word and with this information we can better understand the model especially when it makes mistakes. In our first example we can see the model captions the image well. Looking at the attention map it is clear the model is looking at the correct sections of the images as when it generates the word "dog" its attention is on the dog's body and when it generates the word "toilet" its attention is on the toilet.

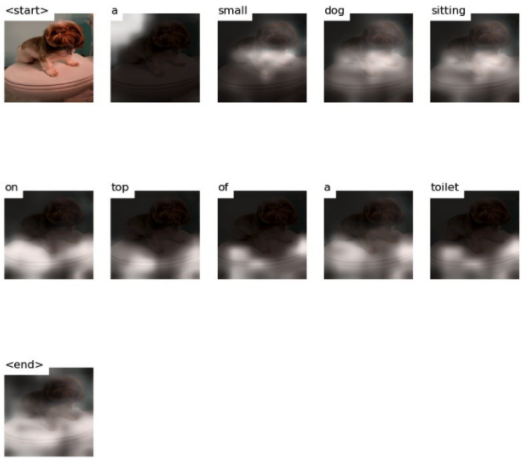

Figure 1: Example 1

In this second example, we see the model again captions correctly and looks at reasonable parts of the images when generating words. When the word "men" is generated the attention is focused on sections of the men in the image and when the word "soccer" is generated, the attention is clearly on the soccer ball.

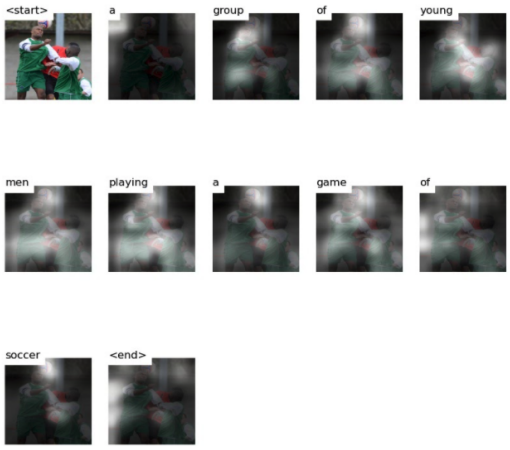

Figure 2: Example 2

An example of when our model does poorly but is explainable is shown below. One of the captions for this image is "a table with some cellphones and other objects". The model gets the part about the table correct but incorrectly assumes that this is the contents of the purse. However, this assumption makes sense as the contents of a purse are usually an assortment of small objects like the image and furthermore, the model is clearly paying attention to the correct sections of the image and the cause of the miscaptioning is not a lack of attention.

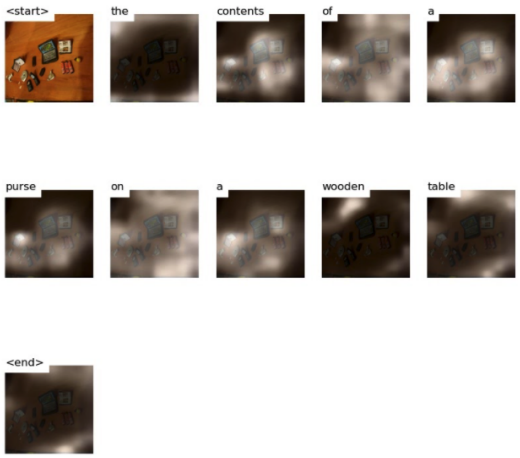

Figure 3: Example 3

Lastly, we see that model can sometimes perform poorly with no easily discernible explanation. In the below image the boy is neither sitting nor looking at his phone. When the model generates the word "sitting" the attention is on the boy's body possibly indicating his stance looks similar to someone sitting. However, when the word "phone" is generated, it makes very little sense as to how regions of the boy's hood could be mistaken for a phone. We can extrapolate the possibilities of why this image was miscaptioned but it seems the model cannot always produce a reason explanation, through attention maps, of why an image was miscaptioned.

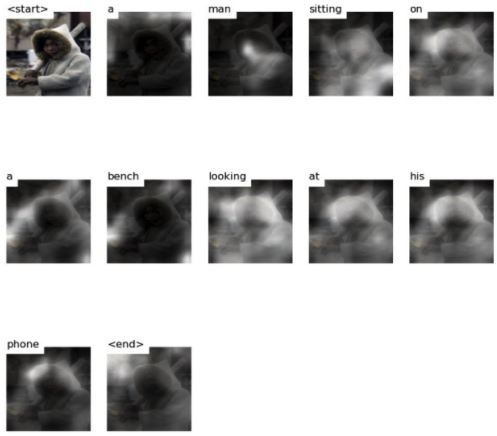

Figure 4: Example 4

3.2. Image Perturbation

Once we produce the attention maps for a given image and its produced caption, we extend on this idea by removing an object from the source image and regenerating the produced caption to assess how it differs from the original. The method we choose to remove objects from an image is called Generative Inpainting. Given an input image and a binary mask, generative in-painting uses a pre-trained gated convolutional neural network to remove the part of the image in the mask and replace it with the model's best guess of what the original image would look like without the masked portion. For example we can take an image of several surfers riding a wave, draw a white mask image where one of the surfers is located, then remove that surfer from the image by inferring what the background image should look like.

Figure 5: Raw Image

Figure 6: Input Image

Figure 7: Output Image

In the examples given above, we are taking a raw image from the COCO data set, generating a white binary mask to place over one of the surfers, then removing that surfer from the image through our pre-trained generative model. The goal here is to produce a counterfactual, or an image with slight modifications from the original, in order to assess how much our caption changes without a given object. We expect that objects in the image more pertinent to the caption will have a greater impact on the counterfactual's generated caption when removed from the image.

There are some limitations to producing counterfactuals with generative in-painting, and some questions we must tackle before automating the production of counterfactuals. First off, we struggled to produce realistic-looking counterfactuals when the object was too large, as seen in figures 8-10 below.

Figure 8: Raw Image

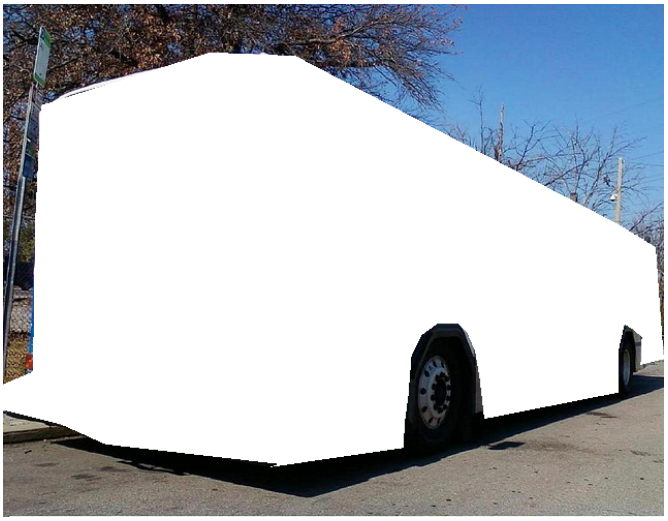

Figure 9: Input Image

Figure 10: Output Image

As we see in the three figures above, the image perturbation model struggles to accurately remove objects that take up a large portion of the image. Our solution to this problem would be to restrict the size of the objects we remove to a specified fraction of the image's size, or to skip the image altogether if this cannot be done.

Another issue we face is in choosing which objects to remove. Our current method entails removing objects at random, based on the list of objects provided to us in the image's annotation. However, this may lead to us removing objects that aren't truly important to the image and will have little affect on the generated caption. To address this issue, we will try and produce an importance metric for each object we remove by comparing not only the difference in generated captions, but also the difference in attention maps produced by the counterfactual and original image. We hope to find that in objects that are more important to the image, we will have a greater change in captions produced, as well as a quantifiable difference in the attention maps generated.

4. Results

4.1. Model Performance

In general, our image captioning model has good performance evaluated by the BLEU metric:

Table 1: BLEU Score of Image Captioning Model

BLEU-1	BLEU-2	BLEU-3	BLEU-4
70.0	52.3	38.2	27.3

These BLEU scores are slighly different than those achieved by the "soft" attention model *Show, Attend, Tell*. The similarity of these scores indicate a successful emulation of their model. We attribute the slight difference in BLEU scores to using different data partitions for the training, validation, and test sets and the natural variation of training models with different hyper-parameters.

4.2. Determining Object Importance

In order to determine how important an object is to the overall caption, we compare what the caption looks like with and without that object. We can make this comparison quantitatively using a pre-trained BERT model that converts sentences into vectors, then compare those vectors using the cosine similarity metric.

Here is an example of our counterfactual changing the produced caption, as well as attention maps:

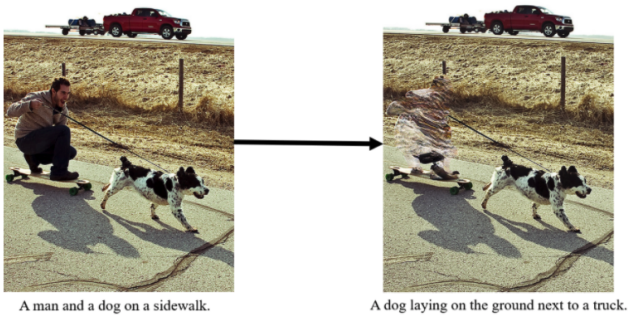

Figure 11: Raw Image and Perturbated Image

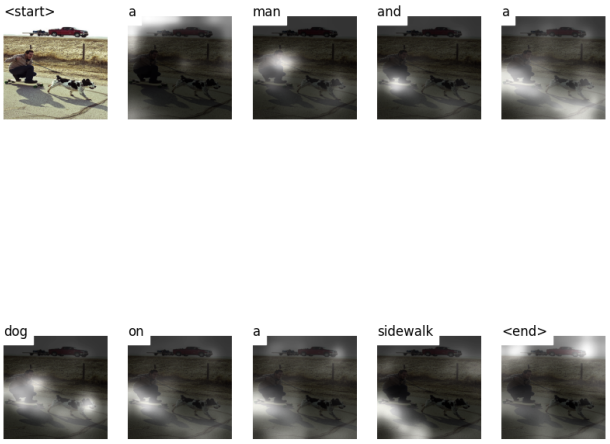

Figure 12: Raw Attention Map

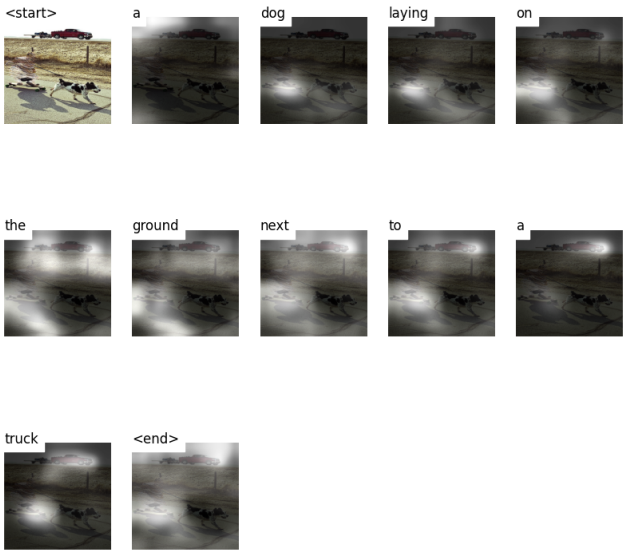

Figure 13: Attention Map after Perturbation

We produce a new image, caption, and attention map for each annotation in the image, then compute our BERT distances between the counterfactual caption and raw image caption, and compare the results. For reference, a BERT distance of 0 means the sentences are identical, and a BERT distance of 1 means the sentences are conceptually the opposite as deemed by the word embedding produced by the pre-trained BERT model.

Table 2: BERT Distance Results

Annotation ID	Counterfactual Caption	BERT Distance from Original
Skateboard	A man and a dog playing with a frisbee	0.10
Dog	A man kneeling down next to a dog on a sidewalk	0.07
Truck	A man sitting on the ground with a skateboard	0.04
Human	A dog laying on the ground next to a truck	0.08

The table above allows us to create a visualization of the relative importances of each object to the caption, with the assumption that the difference between the counterfactual image's caption and the original image's caption speak to the importance of that object to the image captioning model.

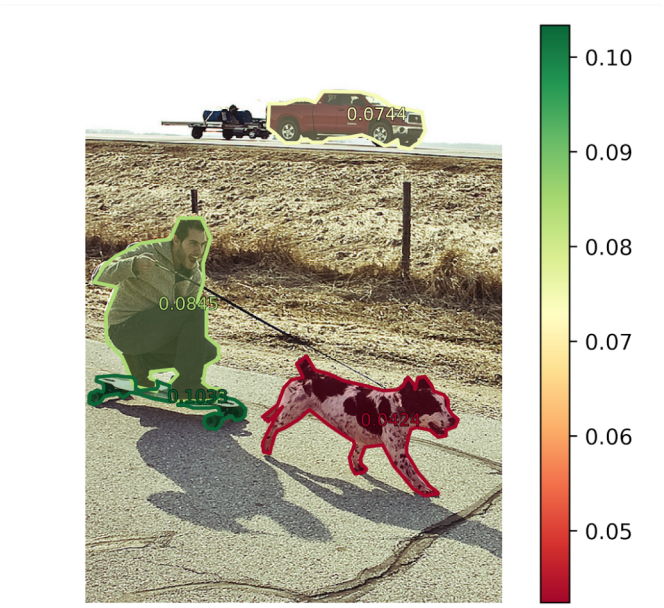

Figure 14: Object Importance Visualization

In Figure 14, we shade each object with it's corresponding importance as valued by how much its corresponding counterfactual affected the original caption. This example alone brings up many interesting questions and insights about how our image captioning model functions, and areas where it may improve based on our intuition. Firstly, we find that, in this image, the size of the annotation doesn't correlate to the amount with which the caption is affected. In-fact, the smallest object, the skateboard, produced the greatest difference in caption. This suggests that not all parts of the image are treated equally and that our model is encoding more relevant parts of the image to produce its caption. Second, we may question why certain objects are more important than others, and whether or not this matches our intuition. For example, one may expect the objects in the foreground to be most relevant to the caption, yet in Figure 14, the truck in the background has a higher importance score than the dog in the foreground. This can lead to many interesting questions about the biases our model contains, such as how prevalent each object is in the training dataset, or how often that object is the focus of the image's caption. Lastly, using BERT distance as an importance score is only one way of thinking about how important an object is to the caption produced by an image captioning model; there are numerous other methods with which one can determine this importance. Another method could, for example, compare the attention maps produced by each counterfactual, either the map for the same word or with all the attention maps added together, to determine what parts of the image the model sees as important before and after the inpainting. Overall, we hope that this tool for visualizing object importance can provide a framework of knowledge about image captioning models that will lead to further research and questions about possible improvements to image captioning.

5. Related Work

For the image captioning section, we implemented the model based on the paper: Show, Attend and Tell: Neural Image Caption Generation with Visual Attention.

For the image perturbation section, we implemented the model with Contextual Attention and Gated Convolution (Link to the repo we based on: https://github.com/JiahuiYu/generativeinpainting).

For the caption similarity comparison part, we use the pre-trained BERT word embedding model to embed the captions into vectors.

6. Conclusion

We trained our own image captioning model with the COCO dataset and evaluated the model with BLEU metric. With the image captioning model, we generated attention maps to visualize and explain to the audience how a caption is generated by our model step by step. We also implemented our image perturbation model and trained it with the COCO dataset. It has decent performance on removing an object from an image and refilling it. With the image perturbation model, we investigated how an caption can be changed if an object is removed from the raw image. Furthermore, we investigated and visualized the object importance by assigning an importance score to each object in an image. In the future, we want to make our model detect adversarial and see how much the caption

changes. We also plan to add a segmentation prediction model to eliminate the need for pre-defined annotations. We hope our work will let more people know about Explainable AI.

7. Reference

[1] Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention", https://arxiv.org/pdf/1502.03044.pdf, 2016

[2] Yu, Jiahui and Lin, Zhe and Yang, Jimei and Shen, "Generative Image Inpainting with Contextual Attention", https://arxiv.org/abs/1801.07892, 2018