[CS682] Final Project Report
Image Captioning using CNN-Transformer

Prahlad Das
University of Massachusetts Amherst
prahladdas@umass.edu

Quoc Anh (Alan) Bui
University of Massachusetts Amherst
qhbui@umass.edu

August 18, 2022

Abstract

Image captioning is a well studied field due to its wide range of application such as human-computer interaction, adding subtitles to video, visual question answering and image search by keywords to name a few. Different people have proposed different ideas to solve this problem optimized to a particular aspect. The goal of this work is to develop a plug and play model which can generate controlled caption. We were mostly relying on the image feature extraction and getting captions of similar featured images. We have used convolution neural network to extract features and transformer to generate summarized caption.

1 Introduction

Inspired by Passage Retrieval for Outside-Knowledge Visual Question Answering [4] future work, we decide to apply modern SOTA models to one of the classical problems in Machine Learning - Image Captioning. Currently, one of the approach that achieves SOTA result for the problem is using CNN-LSTM combined architecture. However, we were inspired that we could do better by taking different approach to the problem. Realizing similar images tend to have similar captions, thus given new input image, we will try to find similar images from the dataset and use their captions to generate a new caption for the input image.

Given an image, create and train (if required) a neural model which can generate a caption for that image. We are taking reference of images which have similar objects. After achieving our first goal we are planning to use object detection in the new image and use only portion of the reference images to find the similar captions which will improve the test time and also accuracy.

2 Background and/or Related Work

Image captioning has been quite a hot research area in the field of machine learning. In the survey paper [2] authors have summarized all the models and the datasets. A nice research paper "Show and tell: A neural Image Caption Generator"[6] by Google has taken similar approach as us. Authors have used two different models- one for capturing features by using deep CNN and another one is for generating text by using RNN. Though this approach can generate a better quality captions but training can be trickier as gradients are flowing through a single neuron between these two models. Another similar research paper[7] is based on attention. In this work authors have used CNN for feature extraction and then RNN with attention over image and then caption generation word by word. In another work [5] authors have used CNN and LSTM to generate captions in Myanmar language. In a research paper by Yahoo "Image Captioning: Transforming Objects into Words"[3] authors have used information about the spatial relationship between input detected objects through geometric attention.

3 Approach

The project consists of 4 main parts.

3.1 Feature extraction

In order to find similar images, we need to measure 'distances' between images. Traditionally, we define the distance between 2 images X and Y as $d(X, Y) = \|x - y\|$ where x and y are their corresponding flatten vectors of pixel values. However, we have seen its insufficiency when studying k-Nearest Neighbors - measuring differences (or

distances) based on only pixel values will give high similarity when X and Y are pixel-wise regardless of the images' contents. However, when we study Convolutional Neural Networks, we know that the outputs before Average Pooling and Fully Connected Layers are the features extracted from the input image. The features clearly represent the image better than its pixel values. Therefore, if we replace $d(X, Y) = \|x - y\|$ where x and y are pixel values vectors to their features vectors, this would give higher credibility to the similarity score (or lower $d(X, Y)$ distance value).

In this project, we will be using ResNet50 from `torchvision.models` with `pretrained=True` configuration to extract the features from the images. Despite that the model may be trained for different computer vision tasks, the output of its convolution layers still remains the features of the images. Thus, a model that had been trained on millions of images will the best tool for this task. In summary, during the training, a pretrained ResNet model will be used to extract the features of the entire dataset and the model will store these features. During testing, ResNet only needs to extract the feature of the input image.

3.2 Similar captions querying

After getting all the images all their features extracted. We can start calculating their distances $d(X, Y)$. For this project, we decide to use Cosine similarity to calculate the similarity score between 2 images. Thus given 2 images X and Y with their corresponding x and y features vectors, we define $d(X, Y)$ as following:

$$d(X, Y, x, y) = \frac{\langle x, y \rangle}{\|x\|_p \|y\|_p}$$

where $\langle \cdot, \cdot \rangle$ in this case is the dot product and $p = 2$. By normalizing x and y, the range of $d(\cdot) \in [0, 1]$ where value (score) of $\mathbf{0}$ indicates no similarity between 2 images and value (score) of $\mathbf{1}$ indicates 2 images are strongly related or similar.

Using this formula, given the test images set, we can compute the similarity scores across all pairs of training and testing images. The result would be a matrix of $D \in \mathbb{R}^{M \times N}$ where M is the number of testing images and N is number of training images. In order to generate k similar images, we can take the k images that have top similarity scores respectively to the queried (testing) image. Figure 1 and 2 show the $k = 4$ images from the training set that have highest top scores to the queried images. After successfully achieve k similar images, we can thus query the according k captions. These k captions will be the input for our next step. Figure 3 and 4 shows top $k = 4$ images that are similar to the queried images along with their captions. Despite we have around

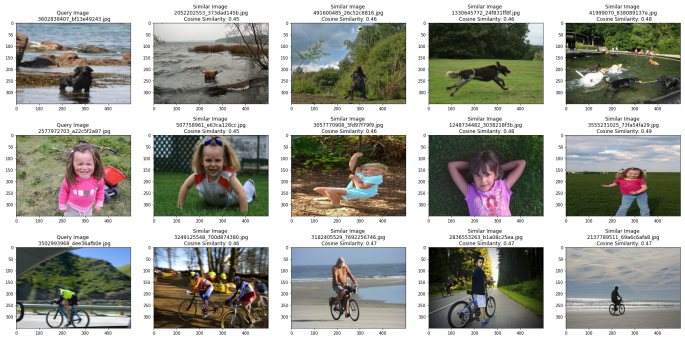


Figure 1: Top $k = 4$ cosine similarity score training images w.r.t to queried images

Figure 2: Top $k = 4$ cosine similarity score training images w.r.t to queried images

5 captions per images, we will just randomly pick 1 instead.

Figure 3: Top $k = 4$ cosine similarity score training images w.r.t to queried images along with captions

Figure 4: Top $k = 4$ cosine similarity score training images w.r.t to queried images along with captions

3.3 Summarize and generate new caption

After acquring k captions in the previous step, we need to consolidate a single caption out of these. For this we are using 't5' encoder-decoder based transformer model which can summarize the k captions.

3.4 Compare against existing SOTA model

In order to test out the performance of our new method, we compare it against the results from existing SOTA model. We train a CNN-LSTM model[1] on the same Flickr8k dataset so that the comparisons are relative. Figure 5 shows some results generated by the CNN-LSTM model. Overall, Figure 6

Figure 5: CNN-LSTM results

describes the full flow of the project.

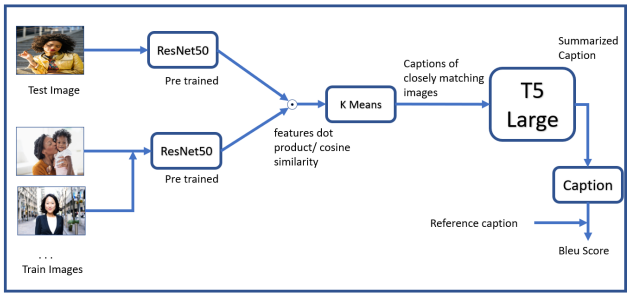


Figure 6: Project flow

4 Experiment

4.1 Dataset

In this project, we decided to use Flickr8k dataset - one of the most common dataset used for image-captioning task as it contains 8000 images along with their corresponding 5 captions. Specifically, there are in total 6000 images for training, 1000 images for validation and the rest 1000 are testing images; each training, validation and test images have 5 captions.

4.2 Metrics

After gathering captions of the corresponding k similar images, we summarize these captions and generate a new caption using T5-Large transformer model. Average BLEU score will be used to determine how well the generated caption using CNN-Transform describes the image compared to the reference (true) caption. Similarly, Average BLEU score will also be used for the caption generated using CNN-LSTM model. We have used only first caption per test image for calculation of BLEU score.

4.3 Result

We achieve the average BLEU score of ~ 0.37 for CNN-LSTM and ~ 0.35 for CNN-Transformer. Our method performs slightly worse, this can be explained by the number of training images that we are currently have is relative small (~ 6000 images). Thus, it is not surprised that some queried images do not have greatly similar images in training dataset, thus affecting the performance of the method. This can also happen due to the fact that we were controlling the length of the caption to be between 5 to 20. And if there is difference between lengths of reference and generated caption, it can affect the score.

5 Conclusion

5.1 Discussion

After comparing the results with reference model and human annotated captions. We can say that generated captions are good in quality. We have added first few test images with generated captions below. Rest of the comparison results are stored in prepossed_files/test_img_captions.xlsx. Time taken to generate caption for each image was $\sim 30s/image$. This time can be reduced if we detect object he image and compare only to those images as explained in the future work. Overall this is a nice method to generate captions without any training and caption will never be very short or repeated for different images.

5.2 Future Work

The first improvement that we can quickly make in the next iteration is that we can acquire more training images. This will help us get better results when querying similar images. Additionally, we can set a threshold value ϵ so that out of k similar images, we can pick a subset that all of its elements have cosine similarity score $\geq \epsilon$. A reasonable value for ϵ is the average cosine similarity score among k images.

Another interesting step that we could take in the next iteration is that we could embed object detection into the current flow. We can use object detection to detect objects inside the input image and then querying similar images that contain the same objects. As the result, this will significantly reduce the test time and potentially improve the quality of caption as we are computing the cosine similarity between the input image with a set of images that initially share lots of similarities, i.e contain same objects (Figure 7).

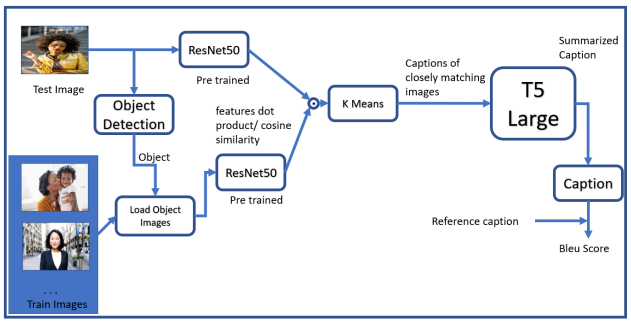
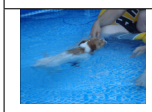


Figure 7: Improved project flow

Image	Reference Model (CNN-LSTM)	Our model (CNN-Transformer)	Human Annotated captions
	dog running through the grass	brown dog playing play wrestling with black dog on snow covered land	the dogs are in the snow in front of fence
	the dog is wearing red shirt and has his tongue sticking out	two dogs play together in the snow	brown and white dog swimming towards some in the pool
	the corner of the man in the red shirt is sitting on the sidewalk	colourfully dressed and painted woman working hula hoop in crowd	man and woman in festive costumes dancing
	fire	santa clause with his white beard and red suit is riding in carriage	couple of people sit outdoors at table with an umbrella and talk
	football player in white uniform is running	football player in red gets ready to throw football	man is wearing sooners red football shirt and helmet
	the dog is running through the grass	small tan dog running in the grass with stick in its mouth	brown dog running
	man in red shirt and glasses is sitting on bench	girl with long dark hair wearing yellow and white shirt is smiling	girl with dark brown hair and eyes in blue scarf is standing next to girl in fur edged coat
	two dogs are playing in the grass	two boston terriers biting at each other	dog with its mouth opened
	dog running through the water	brown dog swimming through lake with stick in his mouth	black dog emerges from the water onto the sand holding white object in its mouth

	basketball player in white and white uniform	four atheletes and man with crowded stadium in the background	player from the white and green highschool team dribbles down court defended by player from the other team
	logo and black and white and black and white dog are playing in the sand	three individuals are posing on skis behind no skiing sign	group of tourists stand around as lady puts her hand near the mouth of statue
	man in red shirt is holding up his hand in the air	couple in funny wigs pose for the camera	man in yellow grimaces
	two dogs are playing in the grass	white dog with black spots catches bright green and pink frisbee in field	dog lays on mattress on the porch
	woman in white shirt and black shirt and black and white and black and white and black and white and black and white and black and white and black and white	asian kids are playing with very large tree branch and small child hanging from that same branch	man is standing on sidewalk in the background with blurry image of another man in the foreground
	the man is wearing red shirt and walking on the side of the side of the hill	boy stands on rocky mountain	girl wearing brown cap red sneakers and dark green coat sits on rock bench

References

[1] Learn to build image caption generator with cnn lstm. In *https://data-flair.training/blogs/python-based-project-image-caption-generator-cnn/*. 3

[2] Ziwei Zhou Chaoyang Wang and Liang Xu. An integrative review of image captionng research. *Journal of Physics: Conference Series*, 32, 2020. 1

[3] Simao Herdade, Armin Kappeler, Kofi Boakye, and Joao Soares. Image captioning: Transforming objects into words. *Advances in Neural Information Processing Systems*, 32, 2019. 1

[4] Chen Qu, Hamed Zamani, Liu Yang, W Bruce Croft, and Erik Learned-Miller. Passage retrieval for outside-knowledge visual question answering. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1753–1757, 2021. 1

[5] Win Pa Pa San Pa Pa Aung and Tin Lay Nwe. Automatic myanmar image captioning using cnn and lstm based language model. In *Proceedings of the 1st Joint SLTU and CCURL workshop*. European Language Resources Association, 2020. 1

[6] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015. 1

[7] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015. 1