PROJECT - 1: IMAGE CAPTIONING

G. Aravind

23^{rd} September 2021

ABSTRACT

I have worked on image-captioning, where we automatically generate a textual description of an image, based on objects and actions detected in the image. First, the problem is formulated using a basic encoder-decoder architecture, where I tried various architectures and novel techniques inspired from recent research in other areas of Natural Language Processing. Then I have performed image-captioning using Transformers, and compared it with previous architectures.

1. INTRODUCTION

Image captioning is most naturally viewed as an encoder-decoder problem, where we extract and encode the features of images using ConvNets, and further decoded as sequences of words using RNN-based architectures. I have tried many approaches to this problem - the most basic approach is to divide the problem into two parts - using the Inception-v3 model to first extract the image features and use various LSTM architectures to decode the image features into captions.

Then I implemented the whole architecture as a single model - consisting both CNNs for feature extraction and sequential layers for caption generation, where parts are trained simultaneously. This approach resulted in lower performance, using the BLEU score as performance metric. I implemented both these architectures by myself, without using any external codes. Here, I also tried novel approaches to further improve the model, like the use of *Dynamic Meta-Embeddings* [1], incorporated *self-attention* at the decoder while generating captions, and more.

I also trained *multi-modal transformer* [2] models for generating image captions, which resulted in significantly improved performance over the previous models. I trained these models on the flickr8k dataset.

2. TECHNICAL DETAILS

2.1. Encoder-Decoder I

- Extracted encoded features from the images using hidden layers of Inception-V3.
- Encoded words using word embeddings like Glove, Fasttext, and Word2Vec. Also implemented Dynamic Meta-Embeddings here, which allows us to use all three of these embeddings simultaneously by concatenating them, and uses attention to give appropriate weights to each embedding.
- Used LSTMs at the decoder side for processing the text sequence followed by fully connected layers. Also experimented with self-attention layer after LSTM layer.

2.2. Encoder-Decoder II

- Similar to previous encoder-decoder architecture, but instead of Inception-V3, I used my custom image features encoder using trainable CNN layers. Also used Dynamic Meta-Embeddings here.
- This model did not perform as well as previous one, probably due to the fact that this CNN encoder has been trained on lesser data compared to Inception-V3.

2.3. Transformer based model

- In this architecture, used transformers for image captioning, in which the image encoder learns the deep image representation in a self-attention manner, then the caption decoder it to generate captions. [2]
- Compared to above models, this model captures both intra-modal and inter-modal (between image and text features) interactions in a unified attention block. [2]
- This model resulted in increase in performance in image captioning. For this particular model only, I took the original author's code as reference and modified it according to my needs.

3. RESULTS

Architecture	BLEU Score [3]
Encoder - Decoder I	0.23
Encoder - Decoder II	0.15
Transformer model	0.29

4. CONTRIBUTIONS

I implemented the first two models entirely from scratch. For transformer model, I took reference from author's code and I modified it to my requirement. Models were trained on the flickr8k dataset, because the COCO dataset originally used in the paper requires too heavy computational resources.

My novelty in this project includes implementing Dynamic-Meta-Embeddings [1] from scratch (which learns the importance of each of the 3 embeddings - Word2Vec, Fasttext, Glove using Attention for each word), and also includes implementing self-attention at various parts of the decoder. I also experimented with different modifications to the architecture, if it can be considered as novelty.

Originally, for novelty I also planned to replace the encoder-decoder in the Transformer model with Vision-Language pre-trained model like ViLBERT, but that proved to be very difficult to actually implement and integrate it with the original author code, hence I could not complete the implementation in time inspite of putting effort for it.

5. RESOURCES

- Flickr8k dataset consisting of 8k images and captions: *https://www.kaggle.com/adityajn105/flickr8k/activity*
- Toolkits used: Keras (primary) and PyTorch
- Reference implementation for the third architecture (transformer model): *https://github.com/MILVLG/mt-captioning*

6. REFERENCES

[1] Douwe Kiela, Changhan Wang, and Kyunghyun Cho, "Dynamic meta-embeddings for improved sentence representations," 2018.

[2] Jun Yu, Jing Li, Zhou Yu, and Qingming Huang, "Multimodal transformer with multi-view visual representation for image captioning," 2019.

[3] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, "Bleu: A method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, USA, 2002, ACL '02, p. 311–318, Association for Computational Linguistics.