Image Captioning

Nimisha Roy, Yash Lara, Ashutosh Baheti, Srikesh Srinivas
Georgia Institute of Technology

Abstract

The ability to recognize and provide descriptions of detected objects within images using deep learning has a wide range of applications in fields including biomedicine, commerce, military, education, digital libraries, and web searching. The main focus of this project is to implement the novel captions method of image caption generation that uses visual and multimodal space of the input for caption generation. A general approach of this category is to analyze the visual content of the images from the dataset first and then generate image captions from the visual content using a language model. This project seeks to replicate, build upon and experiment with three methods for generating image captions, a) overlaying encoder-decoder architecture with attention mechanism and beam search approaches; b) using LXMERT transformer architecture with mask sampling technique; and c) using a teacher-student model with LXMERT and Fairseq Seq2Seq implementation. We have successfully implemented approaches from 2 articles and were able to reproduce similar/slightly better results based on few modifications.

1. Introduction

In this project, we have implemented two papers, which have shown promise in Image captioning. Image Captioning describes the ability of a model to describe the contents of an image, and describe what is happening in the image. Automatically generating captions for an image is an important research area in the field of Computer vision, and is at the heart of scene understanding. It is an important challenge for both machine learning algorithms and practitioners alike since it needs the machine to mimic the remarkable human ability to convert visual information into descriptive language.

There are different approaches to image captioning methods. With advancements in training neural networks and in natural language processing, recent work has significantly improved image captioning techniques. Many of the methods are based on recurrent neural networks and encoder-decoder frameworks. Using Attention in CovNet model allows for salient features in the image to dynamically come up as needed. However, models that use attention so far suffer from loss of information. Using more low-level representation can overcome this problem, but it requires a more robust mechanism to direct the model to more relevant information. Image Captioning is a very important area in Deep Learning and Computer Vision, and teaches machines one of the most quintessential tasks which is understanding the content of an image and describing it. Although challenging, image captioning can have great impact, for example, allowing visually impaired and legally blind users to better navigate the web and any other digital artefact. Image captioning also has uses in self-driving cars, assistive education technologies and in general making machines understand and interpret visual instances. Our project is a step at confirming the research implemented in the research papers we are studying, and making sure that the results cited in the papers are achievable and replicable to a good extent.

For the first half of the project, we implemented the [6] paper. The paper introduces an attention base model that automatically learns to describe the content of images. The paper details how to train the model in standard deterministic technique using backpropagation and stochastically by maximizing a lower variation bond. The paper uses three datasets: Flick8k, Flick30k and MS COCO. Despite this paper being 5 years old, it is still considered to be exemplary in terms of introducing a robust attention-based mechanism to caption images with high accuracy. Since the paper, there have been developments in using Deep-Stacked LSTMs. Contextual word embeddings and data augmentation techniques ,along with CPTR (Caption Transformers) that allow for image captioning. The paper we implemented showcases a comparative BLUE-4 score as compared to the latest SOTA models, showing that the model proposed in the paper is very robust.

We also experiment with a pretrained transformer model as well as a combination of transformer and Seq2Seq in the effort to solve the problem of image captioning. It is currently done today with passing extracted features from a ResNet-101 or Faster R-CNN into VilBERT for caption retrieval and RNN/LSTM or Seq2Seq for generation tasks.

The final sentences are often evaluated with BLEU-4 scoring. The limitations of the current practice concern caption retrieval, which relies on a bank of sentences. For generation tasks, the LSTM/GRU accuracy is generally lower than that afforded by a transformer.

One paper details a teacher-student approach for text generation where the intermediate outputs of a pretrained teacher transformer can be transferred to the loss function of a student learner model when it trains, offering superior accuracy rates relative to traditional cross-entropy. One novelty of our approach is applying this framework to images rather than simply language translation. A co-opted combination of the cross-entropy loss with this new loss in the fine-tuning process as opposed to the training process may afford superior results, contributing a new loss apparatus to the overall objective of caption generation. Should a derivative of the masking strategies work, it would signify that a model can simple be fine-tuned on lesser data. We utilized MS-COCO dataset with over 120K train images and 560K captions jointly pre-trained on COCO itself paired with Visual Genome dataset. The 6GB COCO 2014 validation images contained a variety of high-resolution images of all categories (e.g. food, automobiles, etc.) and became our final accuracy test bed on which we attempted to create sentence descriptions.

For this study, we have used the MS COCO'14 data set [1]. The COCO (Common Objects in Context) dataset is a large scale object detection, segmenting and captioning dataset. The COCO dataset was created by Microsoft in collaboration with Facebook, CVDF and Mighty AI. The dataset contains 91 common object categories with 82 of them having more than 5000 labeled instances. In total, the dataset contains 2500000 labeled instances of 328000 images. The data set can be downloaded in a pre-divided training and validation data set through the MS COCO website. The 2014 release contains 82,783 training, 40,504 validation, and 40,775 testing images. The split was done taking care of the fact that no near-duplicate images are present in the splits, which was done using gist descriptors. The MS COCO dataset has shown promise in the past for image detection, object segmentation, recognition in context and other computer vision and related-NLP tasks.

2. Approaches

2.1. Encoder-Decoder Model with Visual Attention

In this approach, the model comprises of an encoder that encodes an input image with 3 color channels into a smaller image with learned channels using CNN. This smaller encoded image is a summary representation of all that's useful in the original image. The decoder then looks at the encoded image and generates a caption word by word using LSTM. Additionally, Attention mechanism is used which allows the decoder to be able to look at different parts of the image at different points in the sequence generation task. We have used soft attention wherein the weight of pixels in the image that indicate its importance add upto 1. We have also used beam search approach to transform the Decoder's output into a score for each word in the vocabulary, wherein for every decode step, the top 5 candidates are considered to generate the next set of words. After 5 sequences terminate, the sequence with best overall score is selected as the output. The combined network used in this approach is implemented from [6] and is depicted in Figure 1.

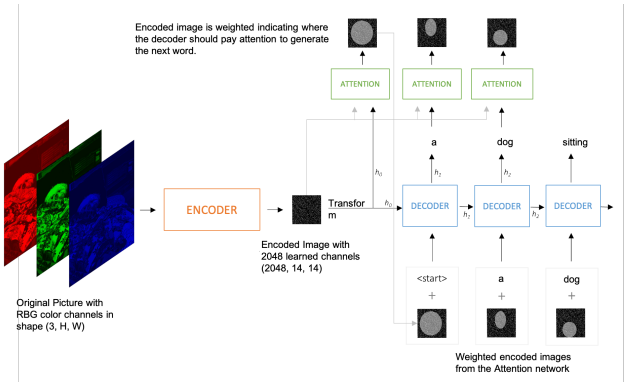

Figure 1: Overview of the Encoder-Decoder network

2.2. Masked prediction with LXMERT

The next approach for caption generation uses a pretrained multi-modal transformer LXMERT [4]. We experiment with masked sampling sequence prediction [5] on LXMERT transformer architecture. Dockerized Faster R-CNN feature extraction was used for image encoding. LXMERT has been pre-trained on 180K image over mix of both MS-COCO 2014 and Visual Genome Datasets. We predicted captions using sequential sampling whereby a set of all mask tokens and extracted features are sent as model inputs. In each iteration a token is sampled from the masked word and inserted into the input for the next iteration, and the process is repeated over several times for the entire caption. This test was inconclusive, although superseding the entire set of mask tokens with partial masking produced somewhat reasonable estimates of the provided images.

2.3. Transformer caption generation model trained with teacher LXMERT

The limited scope and efficacy of the stand-alone transformer model led to the creation of a teacher-student model with LXMERT and Fairseq Seq2Seq implementation[2]. We

[1] https://cocodataset.org/#home

[2] https://github.com/krasserm/fairseq-image-captioning

modified the preprocessing code of FairSeq model by re-tokenized the captions with LXMERT's imported custom BertTokenizer implementation. We use Knowledge-Distillation to transfer probability distributions from pre-trained LXMERT model into Fairseq model [1]. Over MSCOCO captions, LXMERT (i.e. the teacher) generates language scores for masked tokens within the caption. We employ a circular masking scheme whereby every 7th token in the caption is masked. Then mask is shifted one space to the right 7 times (i.e. circulant shift) and we get word probabilities for all words within the caption. These distributions were then used in a modified loss function by the FairSeq (i.e. the student) and defined as such:

$$\mathcal{L}(\theta) = \alpha \mathcal{L}_{bidi}(\theta) + \mathcal{L}_{xe}(\theta) \tag{1}$$

where $\mathcal{L}_{bidi}(\theta)$ measures the loss from the probability distributions and $\mathcal{L}_{xe}(\theta)$ measures the loss of the ground truth labels.

Here, α is the hyperparameter to tweak the weights of different components of the final loss function.

Also,

$$\mathcal{L}_{bidi}(\theta) = -\sum_{\mathbf{w} \in \mathcal{V}} [\log P_\phi(y_t = w | Y_u, X) \cdot log P_\phi(y_t = w | Y_u, X)] \tag{2}$$

and

$$\begin{aligned} \mathcal{L}_{xe}(\theta) &= -\log P_\theta(y_t | y_{1:t-1}, X) \\ &= -\sum_{n=i}^{N} \log P_\theta(y_t | y_{1:t-1}, X) \end{aligned} \tag{3}$$

Note, P_θ represents the word probability distribution of the student and P_ϕ are the word probabilities learned from the teacher (i.e. LXMERT model) over the output vocabulary $\mathcal{V}$.

3. Experiments and Results

For all the three approaches implemented in this paper, the cross-entropy loss function is used. The third approach (teacher-student framework) compares CE loss with KD (Knowledge Distillation) loss results as well. The evaluation of the model's performance on the validation set, however, is done based on the automated Bilingual Evaluation Understudy (BLEU) score [3], due to its ease and popularity in evaluating the quality of machine translated text. We have used the BLEU-4 score to compare the performances among the different approaches and with the state-of-the-art results. [6] observed that model loss stops correlating with the BLEU score after a certain point. So, we have stopped training when we observed BLEU score to start decreasing, irrespective of decreasing loss.

3.1. Encoder-Decoder Model with Visual Attention

The following experiments were performed with this approach.

3.1.1 Encoder Pre-trained Model

For the encoder, we first tested with the VGGNet model that was used in [6]. Thereafter, we also experimented with a different pre-trained model, ResNet- 101 trained on ImageNet Classification task. This was done to experiment with a model that has a better error rate.The last two linear layers of the ResNet model that are responsible for the classification task were stripped away for this purpose. We also fine tuned the ResNet model for better performance. The original paper [6] did not fine tune the VGGNet model, so we wanted to test the performance with a tuned pre-trained model.

3.1.2 Hyper-parameter Tuning

For the hyper-parameter tuning, we experimented with the learning rates of the encoder and decoder as well as the beam size for the beam search operation in each decoding step. One cycle of training took about 90 hours of time while using GPU. So, we were constrained in the amount of hyper-parameter tuning that could be performed.

3.1.3 Use of teacher forcing

In the original paper, teacher forcing was used during the validation process, which means that ground truth values were supplied regardless of the word last generated. Although this is a commonly used approach during training, the validation scores using teacher-forcing would likely not reflect real performance [3]. Hence, our implementation incorporated computing the final test scores with and without teacher forcing, to understand its effect on the performance of the model.

The experiments performed with this approach along with the corresponding BLEU-4 scores obtained for the validation set in the difference scenarios are presented in Table 1.

3.1.4 Training and Results

We first trained the decoder without fine-tuning the encoder for 20 epochs. We observed a peak in BLEU-4 score at

[3] https://https://github.com/sgrvinod/a-PyTorch-Tutorial-to-Image-Captioning

Category	Experimented Cases	Best Case
Encoder Pre-trained Model	[VGGNet, ResNet-101, ResNet-101 with fine tuning]	ResNet-101 with fine tuning
Encoder Learning Rate	$[10^{-4}, 4 \times 10^{-4}, 10^{-5}]$	10^{-4}
Decoder Learning Rate	$[10^{-4}, 4 \times 10^{-4}, 10^{-5}]$	4×10^{-4}
Decoder Beam Search Sizes	[1, 3 , 5, 7]	3
Teacher training for test set scores	[Yes, No]	No

Table 1: Experiments performed on the Encoder-Decoder Model

epoch 11 with a value of 23.19. Thereafter, we continued training the decoder along-with fine tuning the encoder from epoch 11 to epoch 30. The final BLEU-4 score on the test set was 33.04 with beam size 3 using the fine-tuned ResNet-101 model for the encoder. The results are shown in Table 2. We got a better BLEU-4 score than the original paper [3], probably due to a different pre-trained model, fine tuning of the model, as well as different method of computing test score without teach forcing. The results of this approach can also be visualized in Figure 2, where the attention mechanism is highlighted. The parts of the image with higher weights at different time-steps is highlighted to show the effectiveness of the method.

Table 3 shows the BLEU-4 from the implementation of all methods in this study and compares it with the state-of-the-art results.

3.2. Masked prediction with LXMERT

This initial experiment utilized a full sentence of `[MASK]` tokens in caption length, providing the hex feature extractions as well as the supplementary input. Analysis of resultant sentences showed that all mask failed to produce relevant output, however masking all token but the first generated relevant final tokens within the sentence approximately 30% the time. The result is likely explained by the fact that the pre-training of LXMERT on VQA data produced captions as questions as well as introduced a set of arbitrary words to the LXMERT word-sampling. Figure 3 highlights a success case of caption generation in the case of donuts on a table and a failure case. Notice in the case of

Figure 2: Output of the Encoder-Decoder network depicted with focus on the attention mechanism

the airplane, the caption still manage to accurately identify an airplane alongside stairs.

Figure 3: Images with stand-alone LXMERT. (above) A successful caption generation. **Reference Caption: a pile of brown donuts on top of a table. Predicted Caption: A small bowl with two dessert balls with sprinkles**. (below) An unsuccessful caption generation. **Reference Output: A bunch of airplanes are parked on the runway. Predicted Output: the stairs the same plane one will propelled the plane together.** Notice several keywords were predicted correctly.

The overall limited capacity of this model prompted the more successful teacher-student framework.

3.3. Transformer caption generation model trained with teacher LXMERT

For this model we fine-tune the fairseq Image to Caption generation model pretrained with cross-entorpy loss for additional 2 epochs with knowledge distillation loss as given in equation 1.

The results of our experiment with respect to the CE loss benchmark are documented in Table 2. Notice the slightly higher BLEU scores for KD model, fine-tuned from checkpoint 14, and lower perplexity (PPL).

Caption sizes were found to be smaller for Fairseq dataset, this could possibly have caused minor distortion. Rarely, some characters were incorrectly processed as (for instance, the exclamation mark "!"), but these were outliers. warmup-updates parameter was set to 8000 for pretraining with CE loss and zero during fine-tuning with KD loss.

Score	CE Loss	KD Loss
BLEU 1	0.671	0.678
BLEU 2	0.511	0.519
BLEU 3	0.386	0.395
BLEU 4	0.293	0.300
METEOR	0.266	0.268
ROUGE_L	0.534	0.538
CIDEr	0.935	0.968
SPICE	0.194	0.197
PPL	7.2	7.09

Table 2: Final Results for CE Loss vs. KD Loss in LXMERT model

Approach	BLEU-4 Score on Test Set
Encoder-Decoder	0.330
Original Paper of encoder-decoder approach [6]	0.243
Transformer with CE loss	0.293
Transformer with KD loss	0.300
State-of-the-art [2]	0.417

Table 3: Final Results from the approaches implemented in this study compared against the state-of-the-art results

4. Work Division

The work division among the team members are highlighted in table 4

5. Conclusion

In this project, we were successfully able to implement two state-of-the-art papers spanning three approaches in generating captions for images. We produced results similar to the results stated in the research papers. In the case of the encoder-decoder approach, we were able to obtain slightly better results due to fine tuning the pretrained model and interpreting teacher-forcing in generating results differently. In the case of using transformer models , fine tuning the model and using KD loss resulted in better results as compared to using CE loss. the Our project confirmed the implementation and reproducability of the results presented in these papers.

add heightStudent Name	Contributed Aspects	Details
Nimisha	Implementation and Analysis of Encoder-Decoder Model	Data Preprocessing Trained the CNN of the encoder and LSTM of the decoder using GPU. Did Hyperparameter Tuning and some analysis
Yash	Analysis and Documentation of Encoder-Decoder Model	Generated Visualizations. Did Hyperparameter Tuning and some analysis Model Evaluation and Documentation
Ashutosh	Implementation, Analysis, Documentation of Transformer Models	Trained the LXMERT and FairSeq models and generated final captions. Coding for Masking and KD Loss. Documentation updates
Srikesh	Implementation, Analysis, Documentation of Transformer Models	Initial Data Scraping Code contributions for masking and loss. Model Evaluation Documentation

Table 4: Contributions of team members.

Through the project, we were able to deepen our own understanding of neural networks, Encoder-Decoder networks, Transformers and image captioning. We got the opportunity to apply first-hand the many concepts we had learnt in the class. This was a great opportunity for us to strengthen our understanding and foundations in Deep Learning. This project was also chance for us to work in a group setting. We gained valuable experience in working as a team in implementing a Machine Learning project.

This project was a great culmination to the Deep Learning course in Georgia Tech. It was a chance for us to apply all the coursework we studied, and we feel even more confident now in tackling industry and research problems in the domain of Machine Learning and Deep Learning.

References

[1] Yen-Chun Chen, Zhe Gan, Yu Cheng, Jingzhou Liu, and Jingjing Liu. Distilling knowledge learned in BERT for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7893–7905, Online, July 2020. Association for Computational Linguistics. 3

[2] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, and Y. Choi. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137, 2020. 5

[3] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002. 3, 4

[4] Hao Tan and Mohit Bansal. LXMERT: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. 2

[5] Alex Wang and Kyunghyun Cho. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 30–36, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 2

[6] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural

image caption generation with visual attention. *International conference on machine learning*. 1, 2, 3, 5