Image Captioning using Multi Head Attention

Aaditya Gupta 2020552
Harjeet Singh Yadav 2020561
Harsh Vardhan Singh 2020202

Abstract

Image captioning, the task of generating a natural language description of an image, has been an active research area in computer vision and natural language processing. In recent years, deep learning models have achieved remarkable success in this task. This paper provides a review of the recent advances in image captioning using deep learning, including the dataset used, evaluation metrics, and various techniques used in the models. We also discuss the challenges faced in this task, such as handling rare words and long-term dependencies. Finally, we provide insights into the current state-of-the-art techniques and future research directions.

1 Introduction

Let's understand the problem statement first. The task is to generate a textual description of an input image. The whole project can be broken down into two sub-tasks, mainly

- Image Feature Extraction using CNN
- Textual caption Generation using either RNN's or Transformers.

,

2 Related Work

Image captioning in deep learning has been a widely researched area, with various models proposed to generate natural language descriptions of images. Some of the early models used in this area include the neural language models, Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) models. These models were trained to learn the conditional probability of the next word given the previous words and the image features.

Later on, with the advancements in deep learning, more sophisticated models were proposed, including Convolutional Neural Networks (CNNs) for image feature extraction and attention-based models to improve the captioning performance. One of the popular models in this category is the Show and Tell model proposed by Vinyals et al., which uses a CNN for image feature extraction and an LSTM for caption generation.

Recently, the Transformer model, initially proposed for natural language processing tasks, has also been applied to image captioning with promising results. The Transformer model uses self-attention mechanisms to capture long-range dependencies in the input and effectively generates coherent and fluent captions.

There are various existing baselines for image captioning in deep learning, including but not limited to the following:

- Show and Tell model.
- Show, Attend, and Tell model.
- Neural Image Captioning with Attention to Fine-Grained Details.
- Bottom-Up and Top-Down Attention for Image Captioning.

3 Dataset Details

Flickr 8k Dataset (used for both baseline models): The Flickr8k dataset is a popular benchmark dataset for image captioning. It comprises 8,000 images, each with five captions written by human annotators. The dataset was created using Flickr's images, covering various subjects such as people, animals, nature, and objects. The images have varying resolutions, but they are mostly around 500 x 500 pixels in size.

ImageNet (used on InceptionV3 and MobileNet, which are the pre-trained model used for the models we use): ImageNet is a large-scale image database designed for visual object recognition research. It contains over 14 million images belonging to more

than 21,000 categories and has been widely used as a benchmark dataset in computer vision research. The images in ImageNet are annotated with 1,000 object categories that cover a wide range of everyday objects, such as animals, plants, vehicles, and household items.

4 Methodology

4.1 Main model

The model uses an EfficientNetB0 CNN model as the encoder and a transformer decoder as the decoder. The encoder is used to extract features from the input image, and the decoder generates the captions for the input image.

The decoder consists of a stack of TransformerDecoderBlock layers, which are responsible for generating the output caption. Each block in the decoder consists of a multi-head self-attention layer, followed by another multi-head attention layer that takes the output from the encoder and the output from the previous attention layer as inputs. Finally, the output is passed through a feedforward neural network (FFN).

The PositionalEmbedding layer is used to add positional information to the input sequence, and the LayerNormalization layer is used to normalize the input before passing it through each layer. The Dropout layer is used for regularization to prevent overfitting.

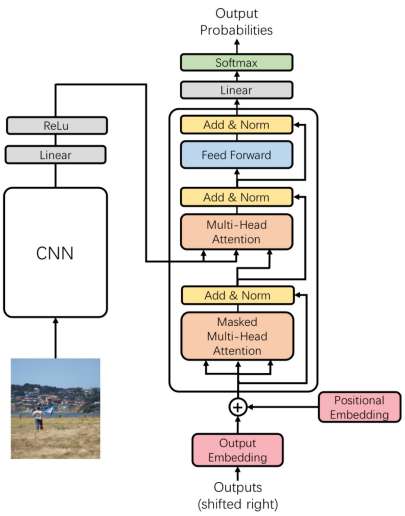

Figure 1: A basic illustration of the architecture of the main model used.

4.2 First Baseline

Our first baseline, "Image captioning with visual attention" demonstrates how to generate captions for images using a deep learning model with an attention mechanism. The model consists of an encoder-decoder architecture with a visual attention mechanism that allows the decoder to focus on different image regions in varying decoding steps.

The model architecture in the tutorial first extracts image features using a pre-trained CNN (MobileNet), which is then passed through 2 self-attention layers to encode the features. Based on the Transformer architecture, the decoder uses cross-attention to attend to the image features and the previously generated words when generating the following word in the caption. The model is trained using teacher forcing, where the ground-truth captions are used as inputs during training.

This model assumes that the pre-trained image encoder is sufficient and focuses on building the text decoder. This model uses a 2-layer Transformer-decoder. The model is inspired by 'Show, Attend and Tell: Neural Image Caption Generation with Visual Attention' but uses an updated 2-layer Transformer-decoder as mentioned above. The model architecture built works in the following fashion. Features are extracted from the image and passed to the cross-attention layers of the Transformer-decoder.

4.3 Second Baseline

"Image captioning" demonstrates how to generate captions for images using a deep-learning model. The model uses a convolutional neural network (CNN) to extract image features, which are then passed to a recurrent neural network (RNN) to generate captions word by word. The RNN is trained using teacher forcing, where the ground-truth captions are used as inputs during training.

The use of an encoder-Decoder model is being done. Here encoder model will combine the encoded form of the image and the encoded form of the text caption and feed it to the decoder. The model will treat CNN as the 'image model' and the RNN/LSTM as the 'language model' to encode the text sequences of varying lengths. The vectors resulting from both encodings are then merged and processed by a Dense layer to make a final prediction.

To encode the image features use of transfer learning is done. The inceptionV3 model is used as it has fewer training parameters and outperforms many other models.

5 Experimental Setup

5.1 First Baseline

Hyperparameters: Adam, loss=masked loss, metrics=[masked acc].

Further experimentation was done by using a GAN in which the captions generated by this model, were used to produce images on a GAN model. The images generated by the GAN were then captioned using this baseline to evaluate the captions. Few examples are shown here.

Figure 2: Caption generated on a GAN made image: "A little boy is jumping over a blue slide"

Figure 3: Orignial caption by the model:"A shirtless man in a blue shirt is standing in front of a crowd"

As we see, this introduces more loss, but it is a good way to check the soundness of captions.

5.2 Second Baseline

The setup of the second baseline is as follows: Every word will be mapped to a 200-dimensional vector to encode the text sequence. For this will use a pre-trained Glove model. This mapping will be done in a separate layer after the input layer, called the embedding layer. To generate the caption two popular methods, Greedy Search and Beam Search, are used to generate the caption. These methods will help pick the best words to define the image accurately. Hyperparameters: Epochs = 15, Batch size = 3, Loss = cross entropy loss, Optimizer = Adam, Activations = relu, softmax Loss = 0.985632

5.3 Main model

Experimental setup of our main model is as following: Hyperparameters used:

- Crossentropy is the loss function that will be used to measure the difference between the predicted captions and the ground truth captions.
- SparseCategoricalCrossentropy is a common loss function used in natural language processing tasks.
- LRSchedule is a learning rate scheduler that will be used to adjust the learning rate during training. The learning rate will start at post warmup learning rate and gradually increase to its final value over warmup steps steps. After the warmup period, the learning rate will remain constant.
- cnn model used is efficientnet
- encoder:TransformerEncoderBlock
- decoder:TransformerDecoderBlock
- caption model:ImageCaptioningModel
- Early stopping patience=3

A sample example for image captioning is shown here, namely Figure 1 and Figure 2.

6 Results

Index	BLUE1	BLEU2	ROGUE-L
Baseline	0.55	0.36	0.53
VisualAttention	0.54	NA	NA
MultiHead	**0.569**	**0.384**	**0.580**

Figure 4: A group of people are standing in front of a large crowd

Figure 5: A boy in a blue shirt is standing on a bench

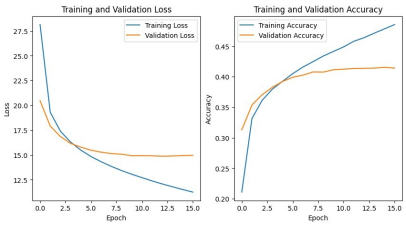

Figure 6: Loss and accuracy curves for the main model

7 Error Analysis

- Model 1 uses a simple CNN (Inception-v3) and LSTM. Inception-v3 is a popular CNN architecture that is widely used for image classification. LSTM is a type of recurrent neural network that is commonly used for natural language processing tasks. The model uses Glove embeddings, which are pre-trained word embeddings.

- Model 2 uses a more advanced CNN architecture (EfficientNetB0) and a transformer decoder. EfficientNetB0 is a state-of-the-art CNN architecture that is designed to be computationally efficient while achieving high accuracy. The transformer decoder consists of a multi-head self-attention layer, which allows the model to focus on different parts of the input image and generate captions more effectively.

- Model 3 also uses a CNN architecture (MobileNet) and a transformer decoder, but it includes visual attention. Visual attention is a mechanism that allows the model to selectively focus on different parts of the input image when generating captions. This can help the model generate more accurate captions by focusing on the most important features in the image. Overall, the main differences between these models are in the CNN architecture used and the type of decoder used for generating captions. Additionally, Model 3 includes visual attention, which can help improve caption quality.

8 Contributions

- **Harjeet Singh Yadav**: Baseline 1
- **Aaditya Gupta**: Baseline 2
- **Harsh Vardhan Singh**: Main model

9 References

- Gautam, T. (2021, January 20). A guide to using transformers using TensorFlow for caption generation. Analytics Vidhya. Retrieved March 19, 2023, from https://www.analyticsvidhya.com/blog/2021/01/implementation-of-attention-mechanism-for-caption-generation-on-transformers-using-tensorflow/

- The team, D. F. (2020, August 6). Python-based project - learn to build Image Caption Generator with CNN LSTM. DataFlair. Retrieved March 19, 2023, from https://data-flair.training/blogs/python-based-project-image-caption-generator-cnn

- Image captioning with visual attention:Tensorflow Core. TensorFlow. (n.d.). Retrieved March 19, 2023, from https://www.tensorflow.org/tutorials/text/imagecaptioning.
- Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R.Zemel, R., Bengio, Y. (2016, April 19). Show, attend and tell Neural image caption generation with visual attention. arXiv.org. Retrieved March 19, 2023, from https://arxiv.org/abs/1502.03044
- Z. Zhang, Q. Wu, Y. Wang and F. Chen, "High-Quality Image Captioning With Fine-Grained and Semantic-Guided Visual Attention," in IEEE Transactions on Multimedia, vol. 21, no. 7, pp. 1681-1693, July 2019, doi: 10.1109/TMM.2018.2888822.
- C. Yan et al., "Task-Adaptive Attention for Image Captioning," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 32, no. 1, pp. 43-51, Jan. 2022, doi: 10.1109/TCSVT.2021.3067449.
- R. Castro, I. Pineda, W. Lim and M. E. Morocho-Cayamcela, "Deep Learning Approaches Based on Transformer Architectures for Image Captioning Tasks," in IEEE Access, vol. 10, pp. 33679-33694, 2022, doi: 10.1109/ACCESS.2022.3161428.