Image Captioning using Vision Transformer

Nagharjun M[1] (nm4074), Abirami S[1] (as16288), Rakshana BS[1](rb5118)

[1]New York University - Tandon School of Engineering
nm4074@nyu.edu, as16288@nyu.edu, rb5118@nyu.edu
https://github.com/Nagharjun17/ImageCaptioningUsingVisionTransformer.

Abstract

Our project aims to address the critical challenge of generating descriptive sentences from images, which lies at the intersection of computer vision and natural language processing. With over 12 million visually impaired individuals [1] above the age of 40 in the United States alone, our proposed solution has the potential to enhance accessibility. Our approach involves training a transformer-based deep learning model, to understand the image and generate linguistically accurate captions. The encoder part utilizes a pre-trained Vision Transformer to extract features from an image. After extracting the features, they are fed to the transformer decoder network to generate captions. We evaluated our model's performance using standard evaluation metrics such as BLEU on the Flickr8K dataset. The Vision Transformer model was able to extract features on par and higher against other traditional CNN architectures in terms of quality. Our approach has potential applications in enabling more efficient image retrieval systems and improving accessibility for visually impaired individuals.

Introduction

Our project focuses on addressing a critical challenge at the intersection of computer vision and natural language processing, namely, generating descriptive sentences from images. This task involves not only identifying objects within an image but also describing their relationships in a manner that sounds natural in human language. With a staggering population of over 12 million visually impaired individuals [1] aged 40 and above in the United States alone, our objective is to provide accurate image captioning that establishes the relationship between sequential images, thereby enhancing the mobility and independence of such individuals.

To achieve this goal, our solution revolves around training a deep learning model, specifically a Transformer-based neural network, capable of encoding and decoding visual features and linguistic structures simultaneously. This approach enables the generation of highly accurate and semantically meaningful captions. Moreover, our Transformer-based model integrates image captioning into a single stage, offering a comprehensive and powerful approach to handle the vast amounts of unstructured image data that dominate the digital landscape.

Traditionally, image captioning has been a challenging task in the field of computer vision, requiring a deep understanding of image content to generate descriptive text. The predominant approach involved utilizing deep convolutional neural network architectures like ResNet and Inception to extract features from images. However, the advent of Transformer Networks [2] in 2017, which have demonstrated significant improvements in various natural language processing tasks, has sparked growing interest in employing this architecture for computer vision tasks, including image captioning.

Transformer Networks have revolutionized deep learning by capturing relationships and semantic meaning in sequential data, surpassing traditional neural networks. They have shown remarkable performance in various domains like Neural Machine Translation, Text Summarization, and Language Understanding. Transformers have also proven effective in computer vision tasks, including Image Classification with Vision Transformers and Generative Networks. In Image Captioning, we can encode an input image into a sequence of features and generate a caption as a sequence of tokens. By leveraging Transformer Networks, we can capture relationships among image features and produce descriptive captions. This report explores the application of Vision Transformers in Image Captioning, leveraging their ability to understand sequential data, context, and semantics.

Literature Survey

Transformer-based models have sparked considerable interest due to their potential to revolutionize computer vision. These models have shown remarkable performance in tasks like image classification, object detection, and automatic image captioning. In this literature survey, we delve into five influential papers that specifically focus on the application of transformer-based models in computer vision. By leveraging attention or self-attention mechanisms, these models employ advanced mathematical techniques to understand the relationships and dependencies between distant data elements. The integration of transformer-based

NYU Tandon School of Engineering

models in computer vision holds great promise and presents new possibilities and challenges that we eagerly anticipate exploring further.

Salman Khan et al.'s survey [3] comprehensively reviews transformer-based models in computer vision, including the Vision Transformer (ViT), Dense Transformer Network (DeiT), and Swin Transformer. It compares their performance to traditional CNN-based models in accuracy, efficiency, and scalability. The survey identifies open problems such as developing more efficient models for larger datasets and exploring unsupervised learning and transferability across tasks. Improving interpretability and addressing real-world challenges like occlusions and lighting variations are highlighted.This survey provides valuable insights and research directions for enhancing transformer-based models in computer vision [3].

The Vision Transformer (ViT) [4] has gained significant attention in computer vision due to its impressive performance. Proposed by Dosovitskiy et al., ViT employs a transformer architecture to encode images into vector features, removing image-specific biases. The paper introduces the Hybrid Vision Transformer (HVT), a combination of CNN and transformer layers, which enhances performance on small datasets. Dosovitskiy et al. conduct experiments comparing ViT with traditional CNN-based models in image recognition tasks such as classification and object detection. ViT achieves state-of-the-art results on benchmark datasets like ImageNet, CIFAR-100, and COCO object detection. Future research directions proposed include exploring different transformer architectures' impact, investigating transferability of pre-trained models across domains, and improving interpretability of transformer-based models.

A modified Transformer architecture [5] designed for automatic image captioning was proposed by He, Sen, et al. in their work, which was inspired by ViT. The model leverages spatial relationships between image regions to generate high-quality captions by dividing the image into a grid of non-overlapping regions and learning the relationships between them using self-attention mechanisms. The authors demonstrate that their model outperforms several state-of-the-art methods on the COCO dataset, achieving competitive results in both quantitative metrics and qualitative analysis. Future research could explore pre-trained models and investigate the relationship between the grid size and caption quality.

In a recent paper, researchers proposed a pure Transformer-based model [6] for image captioning that utilizes the Swin Transformer as the backbone encoder for end-to-end training. They introduced a refining encoder and pre-fusion process to enhance multi-modal interaction and capture intra-relationships between grid features. The proposed model consists of an encoder, a decoder, and a captioner module, and employs a combination of cross-entropy loss and reinforcement learning for training. Experimental results on the MSCOCO dataset demonstrate that the proposed model outperforms existing state-of-the-art methods in terms of both quantitative metrics and human evaluations.

Wang et al. proposed GIT, a Generative Image-to-text Transformer[7] that employs a Transformer-based architecture to generate image captions. GIT introduces novel techniques to enhance caption quality, including a multi-task learning strategy, a cross-modal transformer, and a gated positional self-attention mechanism. Evaluations on benchmark datasets demonstrate GIT's state-of-the-art performance, showcasing the potential of Transformer-based architectures for generative vision-and-language tasks.The success of GIT and similar image captioning models highlights the prospects of combining computer vision and NLP for advanced and accurate image understanding systems.

Dataset

The Flickr8K dataset is a widely used benchmark for sentence-based image description. It comprises 8092 images in JPEG format with varying shapes and sizes, of which 6000 are used for training, 1000 for validation, and 1000 for testing. The Flickr8K text folder contains text files that describe the train set and test set, while the Flickr8K.token.txt file contains five captions for each image, totaling 40460 captions. The dataset has become a standard benchmark for beginners in the field of sentence-based image description and has been used in various deep learning projects, including image caption generators that use CNN and LSTM units to recognize the context of an image and describe it in natural language. The Flickr8K dataset has also been used in studies exploring the possibility of textual and visual modalities sharing a common embedding space.

Transformer Network Architecture

The Transformer Network is an Encoder-Decoder architecture where the Encoder generates a vector representation of the input sequence, and the Decoder computes its relationship with the Encoder outputs to determine the relevant information for the corresponding target.

Attention Mechanism in Transformer Networks

The Transformer Network employs the attention mechanism to capture intricate relationships between sequences. The attention mechanism allows the model to focus on the most informative parts of the input and align them with the corresponding linguistic elements in the captioning process.In the context of image captioning, attention weights are dynamically assigned to individual image patches, indicating the regions of interest for the model. Through a learning process, these attention weights are optimized to highlight relevant visual features and suppress irrelevant information.

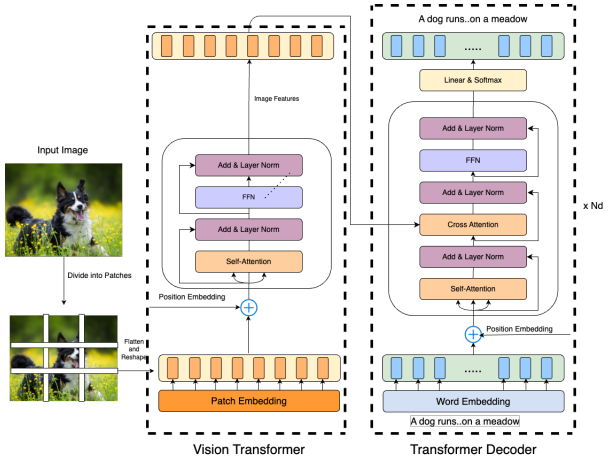


Figure 1: Transformer Network Architecture

Multi-Headed Attention Mechanism

In the Transformer model, the attention module is extended to multiple parallel computations known as attention heads. Each attention head consists of separate learnable weight matrices for queries, keys, and values, enabling the model to capture different patterns and relationships within the data. By employing multiple heads, the Transformer network gains the ability to capture diverse and complementary information in the input sequence. The results from all attention heads are then concatenated and transformed, providing a rich and comprehensive representation that encodes a nuanced understanding of the data.
This multi-head attention mechanism enhances the model's capacity to handle complex dependencies and capture fine-grained contextual information, contributing to improved performance in tasks such as image captioning. The Transformer model revolutionized the implementation of attention by dispensing with recurrence and convolutions and, alternatively, relying solely on a self-attention mechanism.

Encoder

In the Encoder, we utilize the powerful Vision Transformer (ViT) model to process our input images. The input to the Encoder is a batch of images with dimensions (batch size, image size, number of channels in each image). These images are first divided into fixed-size patches using the Patches class, as demonstrated in the example provided. Positional embeddings are then added to each patch to incorporate spatial information. The resulting patches with positional embeddings are passed through the Encoder layer, which consists of a self-attention mechanism with residual connections. This attention layer attends to the patches, producing a context vector that captures important visual features. The context vector is subsequently fed into a point-wise feed-forward neural network. Finally, the Encoder outputs a tensor of shape (batch size, sequence length, d model), representing the encoded information extracted from the input images. By employing the Vision Transformer as the backbone of our Encoder, we effectively capture and encode the visual content of the images for further processing and caption generation.

Decoder

Playing a vital role in the Transformer model, the Decoder facilitates the generation of target captions based on encoded image representations. It takes the target captions as input and applies positional encoding to incorporate sequential information. To ensure effective learning, the Decoder employs a masked self-attention layer that restricts its attention to previous outputs during training. This contextual understanding allows the Decoder to effectively generate captions.

Furthermore, the Decoder utilizes a Multi-Head Attention layer where the target captions serve as queries, and the encoder outputs act as keys and values. This enables the network to establish relationships between the captions and relevant image regions. During training, attention weights are learned and backpropagated, facilitating the learning process. The Multi-Head Attention layer produces a context vector, which is transformed through a softmax function to generate prediction probabilities.

The Decoder is composed of multiple DecoderLayer instances, incorporating self-attention, point-wise feed-forward networks, and layer normalization. Self-attention mechanisms attend to the input sequence, refining representations using attention weights. Point-wise feed-forward networks further enhance the expressive power of the representations. Layer normalization ensures stable training, while dropout regularization is applied within the DecoderLayer.

Combining the Encoder and Decoder, the Transformer model is created. The vision transformer, obtained from the TensorFlow hub, plays a crucial role in encoding image patches. The Encoder processes the encoded patches, producing a context vector that is fed into the Decoder. The Decoder generates the final output, which is passed through a dense layer to produce predicted target captions. The entire model is jointly trained, with attention weights and gradients backpropagated.

The incorporation of the vision transformer in the Transformer model proves beneficial, enabling effective encoding of visual information from image patches and learning of intricate relationships between captions and relevant image regions. Leveraging the power of the vision transformer, the Decoder successfully generates accurate and contextually meaningful target captions.

Parameters	Values
Loss Function	Cross Entropy
Learning Rate	0.00001
Batch Size	32
Optimizer	Adam
Regularization	L2 Penalty
Gradient Regularization	Gradient Clipping

Table 1: Model Configuration

Training

Loss Function and Optimizer

In our project, the cross entropy loss function was employed to quantify the dissimilarity between the predicted sequences and the ground truth target sequences. By calculating the cross entropy loss, we aimed to train the model to minimize the discrepancy between the predicted and actual translations. The specific value of the L2 penalty, which was set to 0.5, influenced the regularization of the model during training. The L2 penalty adds a penalty term to the loss function, discouraging large weights in the model and promoting smoother and more generalizable solutions. As for the optimizer, we chose to use the AdamW optimizer. It is an extension of the Adam optimizer that incorporates weight decay regularization. The learning rate, set to 0.00001, determines the step size taken during parameter updates. By adjusting the learning rate, we can control the rate of convergence and optimize the model's performance.

Hyperparameter Selection

To ensure effective training and robust model performance, several key hyperparameters were thoughtfully selected. The number of epochs, set to 40, determined the total number of complete iterations over the training dataset. This value ensures sufficient exposure of the model to the training examples, allowing it to gradually learn and refine its translation capabilities. However, in addition to the number of epochs, it is crucial to address potential issues that can arise during the training process.

One such issue is the risk of exploding gradients, which can lead to unstable optimization and hinder convergence. To mitigate this problem, gradient clipping was implemented with a threshold of 2.0. Gradient clipping acts as a safeguard by constraining the magnitude of the gradients during backpropagation. If the gradient norm exceeds the specified threshold, it is rescaled to ensure stable and controlled updates to the model's parameters. By effectively preventing the gradients from growing too large, gradient clipping enables smoother optimization and helps the model converge more reliably.

Furthermore, the evaluation period, set to 10 steps, played a critical role in monitoring the model's performance during training. At regular intervals, every 10 steps, the model's progress was evaluated to assess its translation quality and observe its learning trajectory. This periodic evaluation serves as a valuable insight into the model's performance evolution over time and aids in identifying potential issues or areas for improvement. By closely monitoring the model's progress, we can make informed decisions about adjustments in hyperparameters or training strategies, ultimately leading to improved translation capabilities.

Additionally, the choice of a batch size of 32 also contributes to the effectiveness of the training process. The batch size determines the number of training instances processed in parallel before updating the model's parameters. A batch size of 32 strikes a balance between computational efficiency and gradient noise reduction. By processing multiple instances simultaneously, computational resources are utilized efficiently. Moreover, it helps in reducing the noise caused by individual instances, leading to a smoother convergence during training.

Evaluation - BLEU Scores

The quality of the generated sequences was assessed using BLEU scores, a widely adopted metric for evaluating machine translation performance. The chosen hyperparameters, such as the learning rate of 0.00001 and the L2 penalty of 0.5, significantly influenced the training dynamics of the model and subsequently impacted the quality of the generated translations. BLEU scores were employed to measure the similarity between the generated translations and the reference translations based on n-gram matches. By computing 1-gram, 2-gram, 3-gram, and 4-gram BLEU scores, a comprehensive evaluation of the model's translation accuracy was obtained. Higher BLEU scores indicate a greater degree of similarity and alignment between the generated translations and the ground truth references. The utilization of BLEU scores enables objective comparisons between different model configurations, facilitating the identification of optimal hyperparameter settings and training strategies for machine translation tasks.

Evaluation

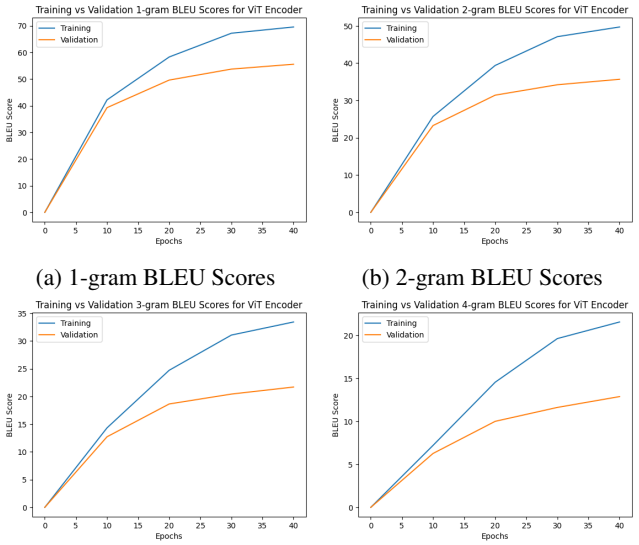
ViT Encoder - Training				
Epochs	1-gram BLEU	2-gram BLEU	3-gram BLEU	4-gram BLEU
10	42.177	25.6924	14.33078	7.1955
20	58.279	39.3787	24.7216	14.5401
30	67.186	47.0968	31.0677	19.6122
40	69.532	49.6889	33.4294	21.5386

Table 2: Training BLEU Scores for various epochs.

In Table 1, we have experimented with our model on different epochs and evaluated it with 1-gram, 2-gram, 3-gram, and 4-gram BLEU Scores. One common trend we can notice is that the model is training in the right direction as the BLEU Scores increases as the number of epochs increases.

ViT Encoder - Validation				
Epochs	1-gram BLEU	2-gram BLEU	3-gram BLEU	4-gram BLEU
10	39.2688	23.2618	12.7236	6.2591
20	49.6166	31.413	18.643	9.995
30	53.7354	34.2057	20.4285	11.6105
40	55.5569	35.6686	21.698	12.869

Table 3: Validation BLEU Scores for various epochs.



(a) 1-gram BLEU Scores (b) 2-gram BLEU Scores

(c) 3-gram BLEU Scores (d) 4-gram BLEU Scores

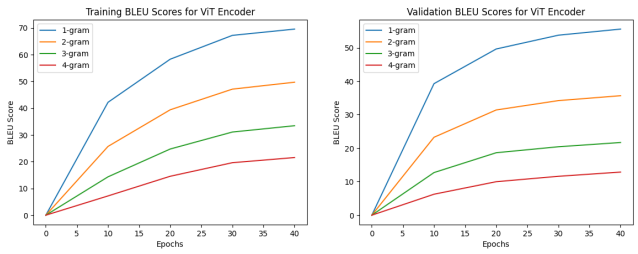
Figure 2: BLEU Scores

In Table 2, we have 1-gram, 2-gram, 3-gram, and 4-gram BLEU Scores on the Validation data. The validation metrics are not very far behind the training metrics.

Figure 2 represents the BLEU Scores for different combinations of n-grams. We can notice the BLEU Scores increase every epoch. There is a steady increase in the start but as the epochs rise, the learning becomes slow. This may be due to the higher learning rate at the start. The model does not converge and maybe decreasing the learning rate further and training can make the model converge faster. Methods such as learning rate decay can be used to experiment with the model.

Figure 3(a) represents the BLEU Scores for the training data. We can see that the model is able to predict 1-grams with increasing confidence as the number of epochs increases. It is then followed by 2-grams, 3-grams, and 4-grams.

Figure 3(b) represents the BLEU Scores for the validation data. It follows the same trend as training with predicting 1-grams with the highest confidence.



(a) Training BLEU Scores (b) Validation BLEU Scores

Figure 3: Training and Validation BLEU Scores

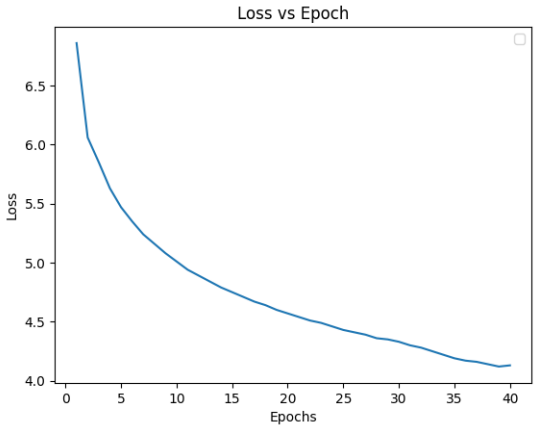


Figure 4: Loss vs Epoch Curve

Figure 4 represents the loss with respect to the epoch. The loss is decreasing as the number of epochs increases.

Encoder	1-gram BLEU	2-gram BLEU	3-gram BLEU	4-gram BLEU
ViT	67.186	47.0968	31.0677	19.6122
ResNet50	65.765	45.423	29.892	18.236
Inception	63.825	44.055	28.74	17.813
DenseNet	61.758	42.0103	26.814	16.268

Table 4: Training BLEU Scores for various epochs.

From Table 4, we can see that using Vision Transformer as the encoder is producing the same calibre of results compared with popular CNN architectures.

Figure 5 shows an accurate caption generated by the model. Figure 6 shows an inaccurate caption generated by the model. It can be understood that the model is not perfect in nature. The model needs to be trained for more epochs to increase its accuracy of caption generation. We could not train it for more than 40 epochs due to limited resources. A lot more computing resources are required to further train the model.

Figure 5: Correct Caption

Figure 6: Incorrect Caption

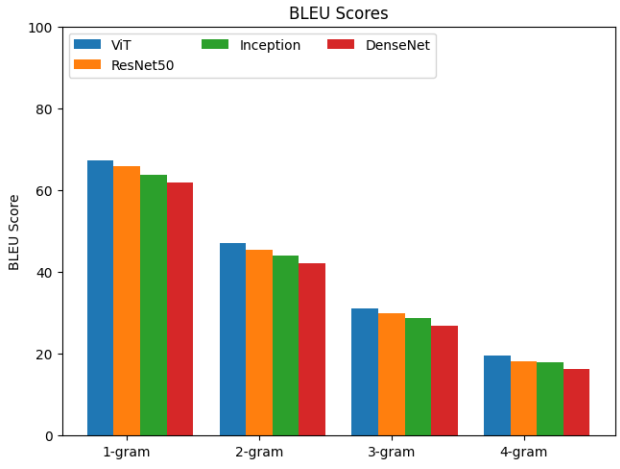


Figure 7: Performance Comparison - BLEU Scores

Conclusion

In summary, our project tackles the task of generating descriptive sentences from images, with a specific emphasis on improving accessibility for people with visual impairments. Our main goal is to leverage the power of attention mechanisms and transformers in Computer Vision instead of the usage of traditional CNN architectures such as ResNet, DenseNets, and Inception to extract essential features and feed them to the traditional transformer decoder to generate captions. For the same reason, we have utilized the Vision transformer network as the encoder. Through training a Transformer-based neural network to simultaneously encode visual features and linguistic structures, we have created a robust model for generating accurate and semantically meaningful captions. After training our model for 40 epochs, we were able to find that the transformer mechanisms perform on par and higher against traditional CNN architectures. By evaluating the models using BLEU Scores, our Vision Transformer approach achieves a 1-gram BLEU Score of 69.532, 2-gram BLEU Score of 49.6889, 3-gram BLEU Score of 33.4294, 4-gram BLEU Score of 21.5386.

References

[1] Fast facts of common eye disorders. Centers for Disease Control and Prevention. (2022, December 19)

[2] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. ”Attention is all you need.” Advances in neural information processing systems 30 (2017).

[3] Khan, Salman, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. ”Transformers in vision: A survey.” ACM computing surveys (CSUR) 54, no. 10s (2022): 1-41.

[4] Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani et al. ”An image is worth 16x16 words: Transformers for image recognition at scale.” arXiv preprint arXiv:2010.11929 (2020).

[5] He, Sen, Wentong Liao, Hamed R. Tavakoli, Michael Yang, Bodo Rosenhahn, and Nicolas Pugeault. ”Image captioning through image transformer.” In Proceedings of the Asian conference on computer vision. 2020.

[6] Wang, Yiyu, Jungang Xu, and Yingfei Sun. ”End-to-end transformer based model for image captioning.” In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, no. 3, pp. 2585-2594. 2022.

[7] Wang, Jianfeng, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. ”Git: A generative image-to-text transformer for vision and language.” arXiv preprint arXiv:2205.14100 (2022).