Vision-to-Words: A Resource-Efficient Transformer-Based Approach for Image Captioning

Nagendra B Hanchinale[1], Parikshith Patil[2], Shridhar B Anigolkar[3], Karthik K Noolvi[4], Raghavendra V Vadavadagi[5], Uday Kulkarni[6], and Shashank Hegde[7]

School of Computer Science and Engineering,
KLE Technological University, Hubballi, India
{01fe22bcs041, 01fe22bcs168, 01fe22bcs215, 01fe21bcs163, 01fe22bcs202}@kletech.ac.in,
uday_kulkarni@kletech.ac.in, shashank.hegde@gmail.com

Abstract. Image captioning is an important problem of artificial intelligence that involves computer vision and natural language generation to provide textual descriptions of images. This paper proposes a new method of using EfficientNetB0 for image encoding and a Transformer for decoding to enhance the captioning process. Unlike other models which depend on ResNet or VGG structures, EfficientNetB0 is a lightweight yet very powerful feature extraction system that also increases the computational efficiency. The proposed model was trained using the Flickr8k dataset with the pre-processing to enhance the variability of the data. The performance evaluation based on the BLEU scores indicates the proposed approach is better than the baseline models with BLEU-1 of 0.5924 as compared to ResNet-50 with 0.4800, and VGG16 with 0.5680. Nevertheless, there are some issues which can be associated with creating detailed captions for intricate pictures. Future work will look into using larger datasets and also extend the use of vision-language pretrained models for better context. The experiment results also show that incorporating the EfficientNetB0 and transformers provide a highly efficient and feasible solution for real world image captioning.

Keywords: Image Captioning · EfficientNetB0 · Transformer · Multi-head Attention.

1 Introduction

Caption generation for images is one of the most important and difficult to solve problems which is based on two fields, Computer Vision and Natural Language Processing. It has many uses in the areas of helpful technologies for the blind and visually impaired people, automatic generation of contents for social media, and in intelligent systems that are able to interface with people like a human Karpathy and Fei-Fei [7] . The first problem is in understanding the meaning of the visual data and in giving natural and varied textual descriptions in the context of the images.

The preceding strategies for image captioning included rule-based methods or template-based models Zhang et al. [18]. These systems were rather rigid and could not handle variety of datasets and their flexibility in generating natural and context-sensitive descriptions was quite limited. As the deep learning progressed, the encoder-decoder architectures Sutskever et al. [14] gained more popularity for image captioning. These architectures employ CNNs to extract features from images and RNNs to produce captions word by word using the extracted features. The use of deep neural networks gave a boost to captioning by providing more descriptive and grammatically correct outputs as stated by Sudhakar et al. [13] .

However, there are some drawbacks of the CNN-RNN-based models, which are as follows: RNNs work in a sequential manner and hence it is challenging to establish long-term dependencies and associations between the visual content and textual descriptions. The sequential implementation also causes issues with training and inference since RNNs are step by step instead of parallel computation. These concerns were partially mitigated through attention mechanisms that enabled the models to direct their attention to certain parts of an image when captioning, which enhanced the performance by assigning importance

levels to various features of the image Ridoy et al. [12]). Although the attention-based RNN models have enhanced caption quality, the effectiveness of these models was limited by the serial nature of RNNs.

The key contributions of this paper lie in addressing these challenges by introducing a transformer-based image captioning model that eliminates sequential processing constraints and enhances both accuracy and efficiency. Unlike RNNs, transformers leverage self-attention mechanisms Vaswani et al. [16] to model both local and global dependencies effectively, enabling the model to process an entire input sequence in parallel. This improves both computational efficiency and the ability to capture intricate relationships between image features and textual representations. The proposed framework incorporates EfficientNetB0 Joshi et al. [6] as the feature extractor, ensuring a balance between computational efficiency and high-quality feature representation. Recent studies have demonstrated that transformer-based architectures outperform traditional sequence-based models in captioning tasks due to their ability to model contextual relationships more effectively and avoid issues such as vanishing gradients Joshi et al. [6].

These are the major contributions of this paper in that this paper proposes a transformer-based image captioning model which is free from the sequential processing issue and also improves the accuracy and efficiency of the image captioning model. While RNNs have the capability to capture the dependencies of a sequence by using recurrence, transformers utilize self-attention Vaswani et al. [16] to capture both local as well as global dependencies, allowing the model to process the whole sequence at once. This enhances efficiency in computations and the likelihood of correlating multiple characteristics of the image to the textual description. The proposed framework utilizes EfficientNetB0 Joshi et al. [6] for feature extraction because it provides a good trade-off between efficiency and feature quality. Current research has shown that transformer-based architectures are superior to sequence-based models in captioning tasks because of their superior contextual relationship modeling and avoiding problems such as gradient vanishing Joshi et al. [6].

The choice of EfficientNetB0 as the backbone CNN is based on the recent works on state of the art deep learning methods for feature extraction. EfficientNet based architectures have been adopted in numerous computer vision problems due to their efficiency and effectiveness. For instance, EfficientNetB7 has been used in medical imaging for fine-grained classification like detection of brain tumor in which advanced fine-tuning feature extraction has enhanced the classification performance Ghosh et al. [5]. Inspired by this, we utilize EfficientNetB0, a computationally lighter variant, to ensure high-quality feature extraction while maintaining real-time processing capabilities, making it suitable for large-scale image captioning tasks. Additionally, CNN-based encoder-decoder frameworks have been explored for compressed image captioning Ridoy et al. [12], emphasizing the need for optimized architectures that maintain high accuracy while reducing computational costs.

Furthermore, by leveraging the parallel computation capabilities of transformers, the proposed model significantly improves training speed and inference efficiency. Unlike RNN-based approaches that generate words sequentially, transformer decoders utilize multi-head self-attention mechanisms to generate captions more effectively by capturing dependencies across the entire sequence simultaneously. Experimental results demonstrate that the transformer-based model, combined with EfficientNetB0 feature extraction, outperforms conventional CNN-RNN architectures in terms of caption quality, computational efficiency, and robustness across diverse datasets. This research contributes to advancing image captioning systems by addressing the limitations of sequential processing while aligning with the growing demand for scalable and high-performance AI solutions.

The remainder of this paper discusses related work in Section 2, the proposed methodology in Section 3, experimental results and analysis in Section 4, and conclusions with future research directions in Section 5.

2 Background

Convolutional Neural Networks (CNNs) have transformed computer vision by providing an organized method for learning spatial feature hierarchies from images. moving beyond traditional techniques that

rely heavily on handcrafted features. CNNs use convolutional layers to automatically discover patterns such as edges, shapes, and textures, progressively building abstract representations of input data.

This parallelism not only accelerates training and inference but also enables transformers These capabilities have made CNNs as the go-to solution for most tasks including image classification, object detection, and image captioning. Krizhevsky et al. [8], Ren et al. [11], Sutskever et al. [14].

Advanced architectures like ResNet,InceptionAtliha and Šešok [2], Degadwala et al. [4] and EfficientNetRidoy et al. [12] reveal that EfficientNet has the unique ability to learn semantic and spatial features so that it can interpret high level complicated image. However, deeper and wider CNN architectures that are developed to enhance accuracy result in high computational cost and hence present the issue of real-time applicability in devices such as mobile gadgets Karpathy and Fei-Fei [7]. EfficientNet Tan and Le [15] offers the solutions on compound scaling to achieve the state-of-the-art accuracy with high computational efficiency in terms of depth, width, and resolution. The EfficientNet family starts with B0 which is designed for lower computational complexity to the higher versions such as B5 Ghosh et al. [5] and B7 Ghosh et al. [5] which come with higher accuracy in terms of computational resources.

Specifically, EfficientNetB0 performs feature extraction quite well and is lightweight, thus suitable for real-time use. On the other hand, models like EfficientNetB5 and B7**?**]**?**] are more suitable for more complicated tasks such as fine grained classification and multi object detection which require high accuracy and high computational resources.

CNNs are used for feature extraction, while RNNs for the generation of sequential captions, thus improving the field of image captioning according to**?**]. LSTMsSutskever et al. [14] and GRUs [17] are specifically designed to handle sequences of visual features and translate them into textual descriptions. However, the Seq2Seq models are not without their drawbacks, and some of them include; generalization problem, scalability problem, and computational complexity problem. The sequential nature of RNNs hampers their ability to learn long dependencies and capture the context in general and especially in the large-scale datasets like MSCOCO and Flickr8k containing a variety of objects, relations, and contexts. Furthermore, processing captions word by word makes training and inference time-consuming and thus hampers real-time use. Some of these issues have been somewhat alleviated with the introduction of attention mechanisms like Bahdanau attentionBahdanau et al. [3] as well as the multi-head attention Vaswani et al. [16] that helps the models to focus on the necessary areas of the image while generating the caption, thus providing more accurate and semantically rich captions. However, these mechanisms are not very effective in capturing the global context of the situation. Self-attention mechanisms of transformer architectures Liu et al. [9] have introduced revolutionary changes as it is capable of processing all the elements in parallel. to capture both local and global dependencies effectively. Multi-head attention Vaswani et al. [16]in transformers enhances their ability to focus on multiple aspects of an image simultaneously, producing captions that are syntactically accurate, semantically rich, and contextually coherent.

3 Proposed Work

This section describes the dataset used, the pre-processing steps applied, the CNN for feature extraction, the Transformer architecture for caption generation, training and optimization details, hyperparameter settings, and the evaluation methodology.

Figure 1 presents the process of an image captioning model that takes both visual and text inputs. First, raw image data is fed into EfficientNetB0, an efficient and lightweight convolutional neural network employed for the extraction of high-level visual features. These features are then inputted into an encoder that converts them into an appropriate sequence form that can be used for sequence generation. At the same time, a text input (e.g., a start token) is supplied to steer the generation process. The decoder is given both the encoded image features and the initial text input to produce coherent word sequences. Last but not least, the model produces a caption that captures the content of the image in natural language.

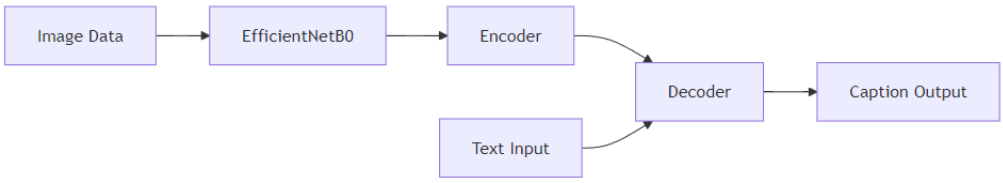


Fig. 1: Encoder-Decoder Architecture for Caption Generation

3.1 Feature Extraction using EfficientNetB0

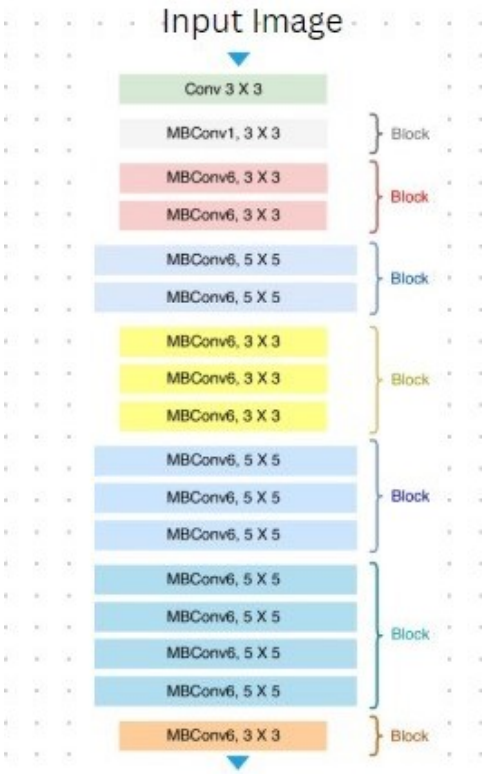


Fig. 2: EfficientNetB0 Architecture

The image2 represents the EfficientNet backbone, which is used as the encoder in the proposed image captioning system. It takes an input image and outputs deep visual features, which are then decoded into meaningful captions by a language model.Aneja et al. [1]. The weights of the network were locked during training in order to avoid forgetting pre-learned features of the image. The input images were of size $299 \times 299 \times 3$, and through EfficientNetB0, the feature map of size $7 \times 7 \times 1280$ was obtained which represented spatial and channel-wise features of the images. These features were used as input to the Transformer encoder as described in the previous section.

3.2 Transformer Architecture

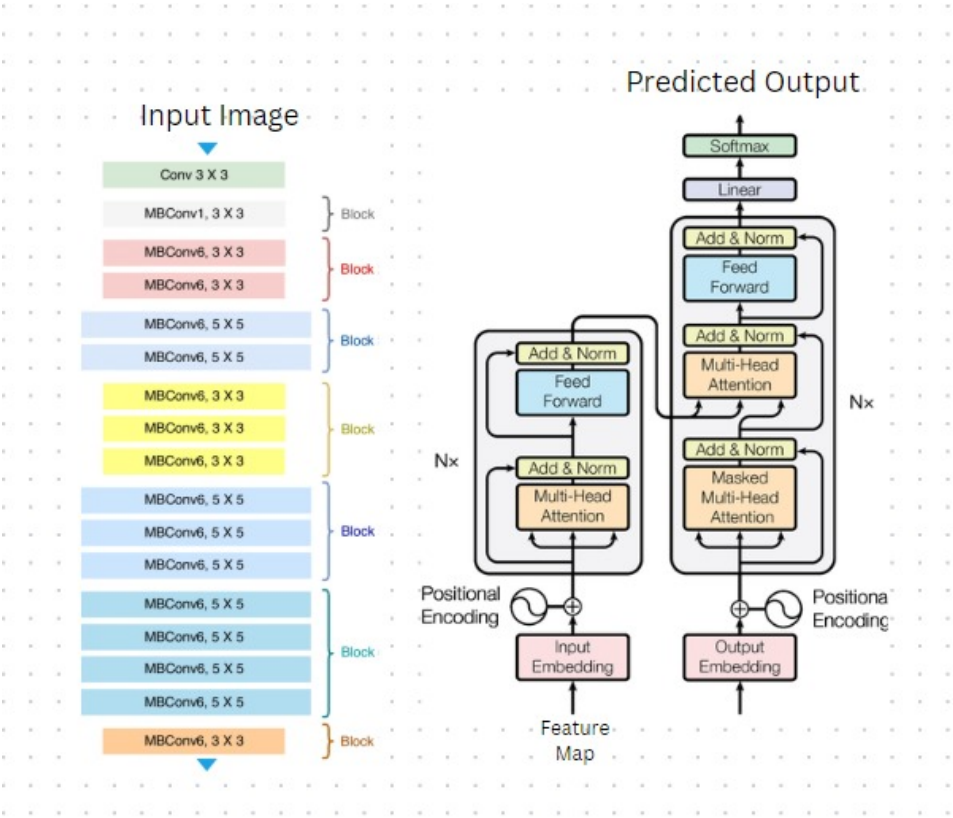


Fig. 3: Transformer Architecture

Transformer architecture as shown in Figure3 Vaswani et al. [16] is an encoder-decoder model Xiao et al. [17], the encoder applies multi-head attention mechanism to the image feature vectors and output contextual feature vectors. This mechanism helps the model to attend to multiple regions of the image at the same time and the attention function is defined by 1:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \tag{1}$$

Here,$Q = XW_Q$, $K = XW_K$, and $V = XW_V$ where W_Q, W_K, W_V are learnable weight matrices and d_k is the dimensionality of the key vectors. The attention output is then passed through another feed-forward network for further processing as shown in 2:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{2}$$

where W_1 and W_2 are weight matrices, and b_1 and b_2 are biases. Additionally, layer normalization is used to stabilize training3:

$$\text{LayerNorm}(x + \text{Sublayer}(x)) \tag{3}$$

The decoder generates captions token-by-token. It uses causal attention to prevent future token access during prediction given by 4:

$$\text{Causal Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \tag{4}$$

Cross-attention aligns the decoder outputs with image features, and the soft-max layer outputs token probabilities shown in 5:

$$\hat{y} = \text{softmax}(z) = \frac{e^{z_i}}{\sum_j e^{z_j}} \tag{5}$$

3.3 Hyperparameter Settings

In this section, we define the hyperparameters used for training the model. The learning rate is set to 1×10^{-4}, a standard starting value for many deep learning models, and may be adjusted based on model performance during training. The batch size is set to 64 images, meaning 64 images are processed together in each training iteration. The number of epochs typically ranges from 10 to 50, depending on when the model reaches convergence. The Adam optimizer is utilized due to its capability to dynamically adapt the learning rate, making it highly suitable for intricate deep learning architectures. The loss function used is sparse categorical cross-entropy, effective for multi-class classification tasks like caption generation. Early stopping is applied to prevent overfitting, terminating training if the validation loss does not improve for three consecutive epochs.

3.4 Training and Optimization

The training process utilized the Adam optimizer with an initial learning rate of 1×10^{-4}, which progressively diminished over the course of epochs. The batch size was set to 64 images, and the loss function used was sparse categorical cross-entropy as shown in equation 6 and 7:

$$\text{CIoU} = 1 - \text{IoU} + \frac{\rho^2(b, b_{\text{gt}})}{c^2} + \alpha v \tag{6}$$

$$\mathcal{L} = -\sum_{i} y_i \log(\hat{y}_i) \tag{7}$$

where y_i is the true label, and $\hat{y}_i$ is the predicted probability for token i. Early stopping was applied, terminating training if the validation loss did not improve for three consecutive epochs.

3.5 Evaluation

Model performance was evaluated using token-level accuracy to measure the proportion of correctly predicted tokens and sparse categorical cross-entropy over the validation set.we have used standard bleu metric to evaluated the generated captions from the model. BLEU scores are calculated across multiple levels, from BLEU-1 (unigram overlap) to BLEU-4 (four-gram overlap). Higher BLEU-1 scores reflect the model's ability to capture relevant keywords, while the BLEU-2 to BLEU-4 scores indicate its ability to form grammatically correct and contextually coherent phrases.

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{8}$$

In the BLEU8 metric, p_n represents the precision for n-grams, which measures how many n-grams in the generated caption match those in the reference captions. w_n is the weight assigned to each n-gram level, balancing the contributions of unigrams, bigrams, and higher-order n-grams. N defines the maximum n-gram order considered, typically ranging from 1 to 4. The Brevity Penalty (BP) is applied to penalize captions that are shorter than the reference, encouraging outputs of appropriate length. Together, these components ensure that BLEU evaluates captions for relevance, fluency, and structural adequacy.

3.6 Caption Generation

The trained model generated captions for validation images by first extracting visual features using EfficientNetB0, then processing these features through the encoder. Captions were generated token-by-token until an end token was predicted or the sequence length reached 24 tokens. Start and end tokens were removed to produce the final captions.

4 Results and Analysis

The primary goal of our image captioning approach is to produce a descriptive sentence that accurately captures the visual content of an image. Our model was trained on the Flickr8k dataset, and its performance was assessed using the standard BLEU score metric to evaluate the quality and relevance of the generated captions.

4.1 Dataset and Pre-processing

The Flickr8k dataset, comprising 8,000 images, each paired with five captions, was utilized. Pre-processing involved resizing all images to 299×299 pixels to match the input size of EfficientNetB0 and normalizing pixel values to the range $[0, 1]$ for uniformity. Data augmentation techniques were applied to enhance diversity in the dataset, including random horizontal flips, rotations within the range of $[-20°, +20°]$, and contrast adjustments varying from 70% to 130% of the original. In caption pre-processing, captions were also converted to lowercase and special characters were also eliminated to obtain the clean text data. The start and end tokens were preserved to delimit the sentences, and only the captions with the length ranging from 5 to 25 tokens were used. Hence, using a lexical analysis of a text containing a word list of the 10,000 most frequent words in the English language was and captions were made of the same length of 25 tokens by padding or truncating the text.The data was then divided into 80:20 split for training and validation respectively and the shuffling was done to ensure that the splits are diverse. Papineni et al. [10]

4.2 Evaluating the model

In the process of testing the proposed image captioning model, one of the most popular tests, the Bilingual Evaluation Under-study (BLEU), was used. The main aspects that are considered to calculate this evaluation metric are the relevance, flow and structure, and it is based on the comparison between the quality of generated captions and reference captions. BLEU is a counting method in terms of counting how many times the n-gram occurs in the generated captions and the ground truth captions and is a measure of how different the generated captions are from the reference in the word and phrase.

4.3 Model Inference and accuracy

In this section, we also perform a quantitative comparison of the proposed model, namely, the CNN EfficientNet-B0 and decoder, with other models that include ResNet-50Atliha and Šešok [2], and other variant of EfficientNet. In this regard, the research concludes that these are the values of BLEU scores, model size and model complexity in relation to the proposed CNN EfficientNet-B0.

Model Performance Comparison The comparison of the model's performance in Table 1 also indicates that the CNN EfficientNet-B0 and decoder performs better than the models based on ResNet-50 and VGG16 in terms of BLEU-1, BLEU-2, BLEU-3 and BLEU-4. Moreover, the table also shows that EfficientNet-B0 is more efficient and smaller in size and computation requirement compared to many of the models, which is suitable for image captioning.

Table 1: Comparison of Model Efficiency and Accuracy

Model	**Size (MB)**	**FLOPs (billion)**	**BLEU-1**
CNN EfficientNet-B0	15	0.39	0.5924
ResNet-50	98	4.10	0.4800
VGG16	528	15.3	0.5680

It is also evident that the integration of the CNN EfficientNet-B0 and the decoder results to a BLEU-1 score of 0.5924 which is higher than the ResNet-50 and VGG16 by a margin of 0.1124 and 0.0244 respectively. This puts more emphasis on the fact that the EfficientNet- B0 has a balance of accuracy and computation cost.

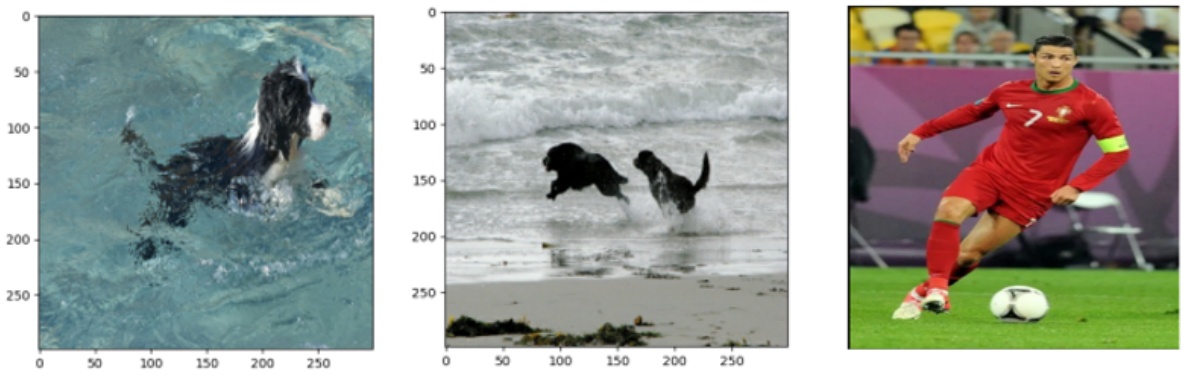
Performance of EfficientNet Variants Comparison of BLEU scores of EfficientNet Variants A comparison of the BLEU scores for various EfficientNet variants including B0, B5 and B7 is as shown below in Table 2. These results prove that EfficientNet-B0 has better BLEU scores for all metrics, and therefore, it is suitable to be used for image captioning.

Table 2: BLEU Score Comparison Across EfficientNet Variants

Model	**BLEU-1**	**BLEU-2**	**BLEU-3**	**BLEU-4**
EfficientNet-B0	0.5924	0.3987	0.2308	0.1123
EfficientNet-B5	0.5788	0.3837	0.2112	0.1018
EfficientNet-B7	0.5822	0.3804	0.2032	0.0903

EfficientNet-B0 achieves the overall BLEU-1 score of 0.5924, which is higher than that of EfficientNet-B5 and EfficientNet-B7. This is evident from the analysis of the different BLEU scores (BLEU-2, BLEU-3, and BLEU-4) where B0 maintains high performances across the board but with less computation time. This makes EfficientNet-B0 the best model to use when there is a constraint on computational resources but accuracy is paramount.

In addition to the quantitative metrics, accuracy analysis was performed to evaluate the model's ability to generate relevant and fluent captions for images. This analysis highlights how well the model understands the visual content and maps it to meaningful textual descriptions. Below are some examples of the generated captions compared to their corresponding images.



(i) (ii) (iii)

Fig. 4: (i)A black dog is running through the water.
(ii) A black and white dog is swimming in a pool.
(iii) A man in a red shirt and blue shorts is playing soccer.

examples from Figure4 demonstrate that the model is capable of generating captions that accurately describe the content of the images. The captions are linguistically fluent and semantically relevant, showcasing the model's ability to identify key visual elements and their relationships within the images. However, there is still room for improvement in capturing finer details or more complex scenes, which could further enhance the model's overall performance.

5 Conclusion

In this study, we explored the integration of EfficientNetB0 and Transformer architectures for the task of image captioning. The model demonstrated comparable performance in generating accurate, fluent, and contextually relevant captions. The use of EfficientNetB0 for feature extraction ensured computational efficiency while maintaining high-quality representations of semantic and spatial features. The Transformer architecture, with its attention mechanisms, addressed several limitations of basic RNN-based models, achieving improved accuracy and contextual coherence. The model achieved notable results on the Flickr8k dataset, with significant improvements in BLEU scores compared to baseline architectures such as ResNet-50 and VGG16. Furthermore, the results indicated that captioning models perform better with EfficientNetB0 due to its ability to reduce computational overhead, making it a suitable option for less powerful systems. While the model performs well in the current setup, there are still areas for improvement. Limitations include challenges in handling more complex scenes or fine-grained details. However, challenges remain in generating highly descriptive captions for complex images. Future work will explore integrating larger-scale datasets and incorporating vision-language pretraining models to enhance contextual understanding. Additionally, evaluating the model on larger and more diverse datasets, such as MSCOCO, will provide deeper insights into its scalability and generalization capabilities. The findings suggest that the combination of EfficientNetB0 and transformers offers an effective and scalable solution for real-world image captioning applications. The model's potential real-world applications, such as in assistive technologies, content generation, and image-based search systems, could benefit from further exploration. Its scalability to larger datasets and deployment on resource-constrained devices remains an important avenue for future research.

References

1. Aneja, J., Deshpande, A., and Schwing, A. G. (2018). Convolutional image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5561–5570.
2. Atliha, V. and Šešok, D. (2020). Comparison of vgg and resnet used as encoders for image captioning. In *2020 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream)*, pages 1–4. IEEE.
3. Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.
4. Degadwala, S., Vyas, D., Biswas, H., Chakraborty, U., and Saha, S. (2021). Image captioning using inception v3 transfer learning model. In *2021 6th International Conference on Communication and Electronics Systems (ICCES)*, pages 1103–1108. IEEE.
5. Ghosh, A., Soni, B., and Baruah, U. (2024). Transfer learning-based deep feature extraction framework using fine-tuned efficientnet b7 for multiclass brain tumor classification. *Arabian Journal for Science and Engineering*, 49(9):12027–12048.
6. Joshi, A., Alkhayyat, A., Gunwant, H., Tripathi, A., and Sharma, M. (2024). Enhancing image captioning performance based on efficientnet b0 model and transformer encoder-decoder. In *AIP Conference Proceedings*, volume 2919. AIP Publishing.
7. Karpathy, A. and Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137.
8. Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.
9. Liu, W., Chen, S., Guo, L., Zhu, X., and Liu, J. (2021). Cptr: Full transformer network for image captioning. *arXiv preprint arXiv:2101.10804*.
10. Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). 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.
11. Ren, S., He, K., Girshick, R., and Sun, J. (2016). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149.
12. Ridoy, M. A. R., Hasan, M. M., and Bhowmick, S. (2024). Compressed image captioning using cnn-based encoder-decoder framework. *arXiv preprint arXiv:2404.18062*.
13. Sudhakar, J., Iyer, V. V., and Sharmila, S. T. (2022). Image caption generation using deep neural networks. In *2022 International Conference for Advancement in Technology (ICONAT)*, pages 1–3. IEEE.
14. Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. *arXiv preprint arXiv:1409.3215*.
15. Tan, M. and Le, Q. V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR.
16. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.
17. Xiao, X., Wang, L., Ding, K., Xiang, S., and Pan, C. (2019). Deep hierarchical encoder–decoder network for image captioning. *IEEE Transactions on Multimedia*, 21(11):2942–2956.
18. Zhang, D., Islam, M. M., and Lu, G. (2012). A review on automatic image annotation techniques. *Pattern Recognition*, 45(1):346–362.