

Enhanced Image Captioning with Deep Learning: Leveraging Attention Mechanism for Improved Image Description

Prof. Pravin Patil¹, Bhupesh Suryawanshi², Om Nikhade³, Nayan Magare⁴ and Kamlesh Mali⁵

¹⁻⁵SVKM's Institute of Technology/Information Technology, Dhule, India

Email: patilpravin555@gmail.com, bhupeshsuryavanshi2@gmail.com, om.nikhade777@gmail.com, magarenayan9@gmail.com, kamleshmali1681@gmail.com

Abstract—With the unprecedented growth in visual content around the internet, it's important to give images relevant captions. The automation of image captioning is beneficial across various areas starting from making images more accessible to blind people to making images easily searchable in social media and e-commerce sites. Over time, this technology has transitioned from basic, rule-driven methods to advanced deep learning techniques that can generate complex and precise descriptions. Previous methods, which were based on command-designed features, frequently produced ambiguous or general captions. A comparison of older technologies (including the one being referred here) vs. modern technology such as deep learning, which can generate meaningful reports about images. With the use of CNNs to learn visual features and RNNs to generate relevant and coherent sentences, this project tries to improve image captioning techniques. Attention mechanisms further improve the effectiveness of the task, allowing the attention system to target only on the meaningful parts of the image. Which results in less accurate (textually) but more visual context-based captions. This paper advances the field of image captioning by proposing a system that produces captions that closely aligned with human-generated captions. The outcomes of this project will improve user experience and serve as helpful solutions in these cases where visual data understanding is required.

Index Terms—Attention Mechanism, CNN, Computer Vision, Deep Learning, Image Captioning, Image Recognition, LSTM, RNN, Transfer Learning.

I. INTRODUCTION

As the amount of image content shared and indexed online continues to rise, so does the importance of generating useful captions for images. This technology has been widely described as image captioning and is used to help visually challenged users to understand the image contents; facilitate image retrieval and categorization; and generally, provide more engaging experiences for users in such diverse applications as social networks and online shops. Previous methods in image captioning, where the features were manually engineered and the dependencies on the parameters were deterministic, were usually unable to come with quite accurate or rich descriptions [1]. This is especially the case as there is a push for more accurate and contextually aligned captions, due to the improvements in deep learning as a method of developing highly complex, natural descriptions.

This calls for the project, “Advanced Image Captioning with Deep Learning” as it seeks to build a reliable image captioning by harnessing the deep learning to overcome the aforementioned challenges. The system effectively overcomes these challenges, utilizing both computer vision and natural language processing approaches by extracting efficient and discriminant visual features with CNNs for feature extraction and encoding coherent textual descriptions employing RNNs such as LSTMs or GRUs[2]. Furthermore, a combination of attention mechanisms helps the model pay more attention to the significant areas within the images and, therefore, generate captions that are both descriptive and contextually correct.

This project uses images of large dataset such as the Flickr8k for training as well as testing the model, thus overcoming the problem of content diversity. Data pre-processing involves augmenting image and caption data with improving quality. Techniques will involve data augmentation which improves the model performance and evaluates performance from BLEU scales. The system’s practical use is even more beneficial when the application is integrated into API so that it can be easily utilized in different applications, such as, social networks, e-shops, and tools for the visually impaired.

Future developments for this project include expanding to more diverse architectures such as Vision-Language Models, coming up with multilingual captioning to reach a more extensive audience and applying a feedback loop mechanism that says enhance captions based on the users’ experience.[3] With an aim to make gestures that link the domain of visual data and natural language more meaningful as well as beneficial for users and organizations across various domains this work targets envisioning effective solutions towards image understanding as a ground-breaking field.

II. LITERATURE SURVEY

All models using attention mechanisms for captioning images have opened up ways that allows models on focussing on specifically selective regions from image for generating the description. Xu et al. (2015) introduced the concept of "Show, Attend, and Tell," a model that applies visual attention mechanisms to focus on key areas during image captioning, creating a stronger link between vision and language. Subsequently, variants like spatial and channel-wise attention have added more detail to this solution, allowing models to produce more pinpoint and contextual descriptions[4].

These two have also contributed significantly to improving Image captioning through both Object detection and feature extraction. Using Dense Cap & GANs, the situated captions tend to focus on particular target objects through object detection frameworks like Faster R-CNN & YOLO. Research shows that recognizing different photometric features through objects enhances the density and semantic truthfulness of the description[5].

CNNs as well as RNNs, specifically LSTMs are commonly used together to generate image captions. Vinyals et al. (2015) showed that it is possible to achieve success using CNNs to extract features while using LSTMs to generate the next output in the sequence and produce coherent, contextually appropriate descriptions[6]. The feature extraction has been beneficially improved with pre-trained CNN architectures such as ResNet and Inception, thus generating dense and semantically related descriptions.

Caption quality assessment needs certain parameters such as Bilingual Evaluation Understudy, METeor, Recycled Understanding for summarization-based evaluation, and collaborative image description assessment. COCO, Flickr30k and Flickr8k datasets are usually employed as reference datasets, so that performances obtained by various methods can be readily compared and evaluated[7].

Recent developments have also looked at techniques that establish how to incorporate text and vision in compilation for improved captioning. Although, Lu et al. (2016) and Anderson et al. (2019) showed that attention mechanisms to CNN enhances the correspondence between image descriptors and textual descriptions, generating meaningful and semantically correct captions[8][9].

In fact, transfer learning has been crucial in extending functionalities of pre-trained models to captioning. By connecting captioning to models learned from object detection and classification it was demonstrated by Karpathy and Fei-Fei (2015) that object detection and classification improves the descriptive quality and versatility of captioning[10].

The transformer architectures used in natural language processing have given a new platform to image captioning. Vision Transformers (ViTs), as presented by Dosovitskiy et al. (2021) relies on Swin Transformer modules that convey self-attention which in turn helps to express elaborate relations across distinct regions of an image. In contrast to the CNN-RNN approaches, ViTs by principle work on the image-level instead of local patches thus providing more rich descriptions[11].

Apart from the CNN-LSTM models, researchers have considered other formations in the recent past. Other works including Li et al. (2020) and tan et al. (2021) analyzed models such as MergeNet with more effective

multimodal fusion approaches for enhancing connection between vision and Language for enhancing the captioning quality and contextual comprehension[12].

Model performance has also been increasing through the increased use of multiple datasets. Wang et al. (2018) also elaborated the effect of the training models on different data sets, which include COCO, Flickr8k & others; results indicated higher exposure to image-caption pairs sufficiently enhances a model's ability to generalize[4]. This results in captions that are more versatile and can fulfil various roles of captioning in various settings and real-life usage.

III. METHODOLOGY

A. Data Collection and Preparation

In this work we employed Flickr8k dataset where each image is associated with multiple captions written by different people. Before using the images, each of them was reshaped to the 224×224 format and transformed into NumPy arrays for the further quick analysis. Bearing in mind that deep learning pre-trained models are developed, this work utilized the VGG16 model to extract features of images. We cut out the last layer of VGG16 that makes predictions of the classes in order to make it output 4096 dimensions instead[14].

For caption pre-processing, captions of images were read from a text file and each caption assigned with the image ID it belongs to. The text was then pre-processed for cleaning by converting it to lower case, text without using alphabet characters and most importantly adding the tokens 'startseq' and 'endseq' to delineate the captions. To add to this, in an attempt to enhance the standardization of the text we made use of Keras's Tokenizer to tokenize the captions alongside padding sequences such that the sequences `bx&` represented matched the longest caption.

B. Feature Extraction

For extracting useful feature representation from the chosen image, we used the pre-trained VGG16 without its last layers, which generates 4096 dimensions from the model's final fully connected layer. These extracted features were saved with the help of the python's pickle module so that use of these features will be easier during both the training as well as the time of doing testing[15].

C. Model Architecture

This model adopts an architecture known as encoder-decoder for generating captions, where:

The encoder part takes image features and provides them to dense layer equipped with dropout and ReLU activation to helping to reduce overfitting. The decoder part employs a bidirectional LSTM whereby the word sequences of captions are tokenized consequently, thus the decoder can learn from past and future words of caption in the sequence. One of the significant improvements made to the model is the inclusion of attention mechanism that allows model to select parts of the image to pay attention to when generating captions. This attention mechanism calculates attention scores by taking a dot product between image feature vectors and LSTM's hidden states. This attention-weighted context vector is then concatenated with the features of the images to solve the discrepancy between the image features and captions.

The final layer is composed of dense layers with softmax activation function, which largely predicts the next word sequence for captions.

D. Training Strategy

To develop an efficient approach to training the model, we defined the data generator function that takes image features, input sequences, and target words and loads them in the small subsets. This approach effectively reduces the memory usage during training without compromising the efficiency of the process.

With Adam selected as the optimizer and loss function selected was categorical cross-entropy, the model was implemented and trained for total 100 epochs using a 32 as batch size. To achieve more balanced exposure to data the dataset was divided into the validating set which was at 10% and the training set at 90%. The number of steps per epoch was determined continuously by the size of accessed data and the defined batches.

E. Model Evaluation

Since it is difficult to evaluate the captions generated by the system since they will have been written in a different language, we had to use a metric that could evaluate the quality of captions generated by system, BLEU score, which we obtained through the Natural Language Toolkit (NLTK) that uses the following formula: To assess the performance, the results were calculated based on the validation dataset and the quantities of training loss and validation loss were plotted.

F. Inference Process and Caption Formation

For inference, the pretrained VGG16 model was utilized again to embed the images of the test images into the respective feature space. Caption generation was initiated with a start sequence token and the model successively predicted the next token till the maximum number of tokens specifying the caption length was crossed or it output the end sequence token or. Last, the automatically generated captions were displayed below the images so as to facilitate understanding in case of complex images.

G. Model Saving and Implementation Tools

Given the substantial size of the trained model and the extensive time required for training, we saved it using Keras's `model.save()` function, while the tokenizer was stored using pickle module of python. This project was done in Python; TensorFlow and Keras were used in deep learning; Matplotlib was used for representations and NLTK for BLEU score. This makes the process very efficient from data acquisition, data transformation, model training, model evaluation and deploying the model into an operational environment. Processing done by model was illustrated in the Fig. 1 below.

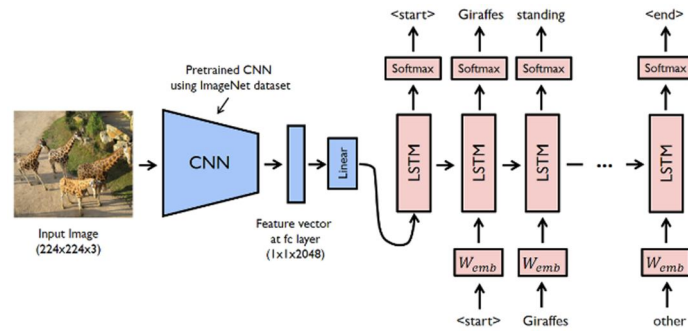


Figure. 1. Model Processing

IV. EXPERIMENT ANALYSIS AND RESULTS

This section outlines the necessary setup for conducting the experiments, with the results analyzed in the following discussion.

A. Experimental setup

Due to flexibility, this project was written in Python with excellent deep learning frameworks such as TensorFlow and Keras. There were other tools like NumPy, Pickle, Matplotlib, PIL, and NLTK for data manipulation, assessing and functions respectively. The entire development process of the solutions described in the given paper was performed in Jupyter Notebook, which allowed for an effective write-and-test approach to code development. Thus for faster computations the project was carried out on a system with NVIDIA GPU (4 GB memory) having support for CUDA.

As a test set for the image captioning, we used Flickr8K set, which consist of images , each of which has its description. They were further preprocessed by being resized to dimensions 224x224 pixels and then transformed into NumPy arrays. The `preprocess_input` function that was from `keras.applications.vgg16` was used with a view of standardizing the pictures. For feature extraction we used a pre-trained VGG16 model with the final classification layer removed its output is a 4096 feature vector. Caption preprocessing therefore entailed, among others, converting all letters to lowercase as well as stripping off all nonalphabetic characters while standardizing for formats. Furthermore, the special tokens at the start and end of each caption were also used to enhance training. The extracted image features and tokenizer were saved using Pickle to be used later on as shall be seen in the next section.

In this work, we utilized VGG16 model as our encoder for extracting features from images in order to develop the model architecture. Video captions were pre-processed using bidirectional LSTM layer while word vectors were used in embedding layer. This work proposed a specific attention map to enhance the attentiveness of the model to various parts of the image. This mechanism performed dot products for encoded features of images and caption embedding and applied softmax activation in order to obtain attention scores. These scores were used to generate a context vector for the decoder with which the image feature vectors were fused. The word embeddings were fed to the decoder which utilised dense layer for generating subsequent word for the caption sequence with a problem of word selection being tackled by use of softmax layer.

The model was trained for 100 epochs by having 32 as batch size, utilizing optimizer Adam and categorical cross-entropy as loss function. A self-define data generator was used to generate image features and caption sequences during training in batches. The data was then laid out into 70% for training and 30% for testing. Apart from the above, precision, recall, F1 score and accuracy has been computed with the help of BLEU score on processed set for test.

Throughout training, model was continuously validated using the test set to monitor loss and learning progress. The trained model was saved in HDF5 format (.h5), allowing for easy reloading and evaluation in future experiments. A function was developed to create captions for images used in testing, where the model iteratively predicted the next word until the end token was reached. BLEU scores were computed at different n-gram levels to assess performance. To gain a deeper insight into the performance of model, additional evaluation metrics such as F1 score, recall, accuracy, and precision were analysed.

To visually analyze the model's structure, we used `plot_model` from Keras. Training and validation losses were tracked throughout the process, and sample predictions were generated to qualitatively evaluate the model's captioning performance.

B. Results and discussion

To assess how well the system performs image captioning model, several metrics were used. Each metric offers distinct insights into various aspects of the model's performance. Fig. 2. graphically presents the results of these metrics.

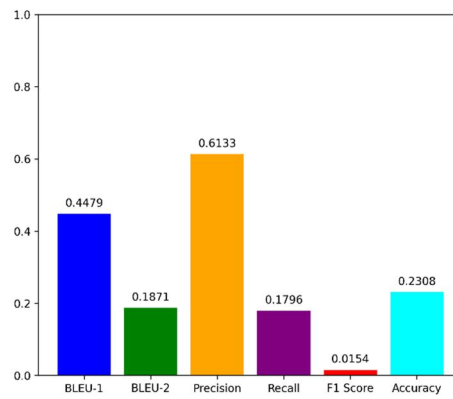


Figure. 2. Performance Metrics

The Bilingual Evaluation Understudy metric known as BLEU is commonly employed to analyse the quality of generated text by measuring its similarity to reference captions. They measure n-gram overlap, with higher scores signifying higher similarity.

The BLEU-1 score evaluates the overlap of unigrams among the generated caption and the guide captions. It helps determine how well the model identifies individual words that are relevant to the image. 0.447888 of BLEU-1 score indicates that model recognizes about 44.8% of the important words, showing basic word matching while leaving scope for improvement.

BLEU-2 score evaluates the overlap of bigrams among the generated and guide captions. It assesses the model's capability to generate meaningful word pairs, contributing to sentence fluency. 0.187083 of BLEU-2 score shows that only 18.7% bigrams align with the captions in the source.

Precision measures the correctly predicted relevant word's ratio to the entire count of words produced by model. It shows how many words generated by model are accurate in context. The precision score of 0.6133 implies that 61.3% of the generated words are relevant, indicating moderate accuracy.

The recall metric evaluates the number of relevant words from the reference captions that the model successfully predicts. 0.1796 of recall score reveals that the model captures only 17.96% of the relevant words, indicating that a large portion of the caption content is missed.

The balanced average of recall and precision, offering a well-rounded evaluation of performance of model is shown by F1 score. It reflects the ability of model to generate correct and comprehensive captions. A low F1 score of 0.0154 indicates an imbalance where the model captures some correct words but fails to cover a complete range of relevant terms effectively.

Accuracy reflects how well the predictions done by model align with the guide captions, offering a broad assessment of its overall performance. An accuracy score of 0.2308 highlights that only 23.08% of the generated

words match the actual reference captions. While recall and precision centers on how well model selects relevant words, general measure of correctness is offered by accuracy, emphasizing the importance of refining the model to enhance both its completeness and precision.

Fig. 3. Was presenting some results of testing images provided to our model.



Figure. 3. Testing Images

V. CONCLUSION

This work has explored the advancement of an advanced image caption generating system, highlighting key techniques and tools that enhance its performance. By incorporating attention mechanisms, object feature integration, and CNN-LSTM fusion, the model can generate more precise and contextually relevant captions. The use of multimodal approaches ensures seamless integration between image data and textual descriptions, improving overall caption coherence. To ensure that model training and deployment must be efficient, the project leverages modern hardware, including multi-core processors, GPUs, and ample RAM, alongside a robust software stack featuring Python and TensorFlow. The integration of pre-trained models, transfer learning, and Transformer architectures helps tackle challenges related to contextual understanding and caption fluency. Evaluation on benchmark dataset such as Flickr, using metrics like BLEU, provides a quantifiable measure of performance of model. This comprehensive approach establishes a good foundation for developing a sophisticated, context-aware image captioning system that balances technical efficiency with innovative deep learning strategies.

REFERENCES

- [1] C. Liu, J. Mao, F. Sha, and A. Yuille, "Attention Correctness in Neural Image Captioning," In AAAI, 2017. [Online]. Available: www.aaai.org
- [2] K. Wang, X. Zhang, F. Wang, T. Y. Wu, and C. M. Chen, "Multilayer Dense Attention Model for Image Caption," IEEE Access, vol. 7, pp. 66358–66368, 2019, doi: 10.1109/ACCESS.2019.2917771.
- [3] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," Dec. 2016, [Online]. Available: <http://arxiv.org/abs/1612.03144>
- [4] S. Zagoruyko and N. Komodakis, "Learning to Compare Image Patches via Convolutional Neural Networks," Apr. 2015, [Online]. Available: <http://arxiv.org/abs/1504.03641>
- [5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," Mar. 2017, [Online]. Available: <http://arxiv.org/abs/1703.06870>
- [6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," Dec. 2015, [Online]. Available: <http://arxiv.org/abs/1512.03385>
- [7] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," Jun. 2016, [Online]. Available: <http://arxiv.org/abs/1606.00915>
- [8] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," Aug. 2016, [Online]. Available: <http://arxiv.org/abs/1608.06993>

- [9] A. Oluwasanmi, M. U. Aftab, E. Alabdulkreem, B. Kumeda, E. Y. Baagyere, and Z. Qin, "CaptionNet: Automatic End-to-End Siamese Difference Captioning Model with Attention," *IEEE Access*, vol. 7, pp. 106773–106783, 2019, doi: 10.1109/ACCESS.2019.2931223.
- [10] A. Gupta, H. Goyal, and I. Kohli, "Synthesis of Vision and Language: Multifaceted Image Captioning Application," *INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, vol. 07, no. 12, pp. 1–10, Dec. 2023, doi: 10.55041/ijsem27770.
- [11] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," Nov. 2016, [Online]. Available: <http://arxiv.org/abs/1611.08050>
- [12] S. Yan, Y. Xie, F. Wu, J. S. Smith, W. Lu, and B. Zhang, "Image captioning via hierarchical attention mechanism and policy gradient optimization," *Signal Processing*, vol. 167, Feb. 2020, doi: 10.1016/j.sigpro.2019.107329.
- [13] T. Bai, S. Zhou, Y. Pang, J. Luo, H. Wang, and Y. Du, "An image caption model based on attention mechanism and deep reinforcement learning," *Front Neurosci*, vol. 17, 2023, doi: 10.3389/fnins.2023.1270850.
- [14] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," 2004.
- [15] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," Dec. 2016, [Online]. Available: <http://arxiv.org/abs/1612.08242>