https://doi.org/10.65091/icicset.v2i1.10

Generating Sanskrit Captions for Images using Transformers and Long Short-Term Memory

Smriti KC
Dept. of CSIT
Padmakanya Multiple Campus
Kathmandu, Nepal
kcsmriti80@gmail.com

Iliya Fathma
Dept. of CSIT
Padmakanya Multiple Campus
Kathmandu, Nepal
iliyafathma@gmail.com

Sudip Raj Khadka
Dept. of CSIT
Samriddhi College
Bhaktapur, Nepal
khadkarajsudip@gmail.com

Mohan Bhandari
Dept. of ICT
SIIT, Thammasat University
Bangkok, Thailand
mail2mohanbhandari@gmail.com

***Abstract*—This study addresses the challenge of automated image captioning for Sanskrit, a low-resource language that lacks dedicated models and datasets for vision-language tasks. We propose a encoder-decoder architecture that integrates a Vision Transformer (ViT) for visual feature extraction with a Long Short-Term Memory (LSTM) for generating syntactically coherent Sanskrit captions. We curated and utilized a dataset of 40,000 image-caption pairs, with English captions from Flickr manually translated into Sanskrit. In validation data, trained model achieved BLEU-1, BELU-2, BELU-3, BELU-4, ROUGE-L scores of 0.3082, 0.1843, 0.1115, 0.0639, and 0.3472 respectively. This work represents a significant advancement in the processing of Sanskrit language within computer vision, with applications in multimedia retrieval for digital archives, automated content analysis of cultural heritage materials, and the development of assistive accessibility tools.**

***Index Terms*—Deep Learning, Image Captioning, Long Short-Term Memory, Vision Transformer**

I. INTRODUCTION

The generation of descriptive textual interpretations of images is the combined result of image processing and natural language processing (NLP) to generate [1]. Generating captions is critical for a wide range of applications, including providing accessibility for visually impaired individuals, enhancing multimedia retrieval systems by enabling image search using natural language queries, and supporting automated content analysis.

Image captioning architectures have gone through significant paradigm changes. The most prevalent models are convolution neural networks (CNNs) for feature extraction and recurrent networks, in particular, long short term memory (LSTM) networks, for sequentially generating text [2]. CNNs capture hierarchical features of the image using localized convolutional operations, while LSTMs add the much-required sequential modeling capability needed to generate text. However, CNNs inherently emphasize the capture of local spatial patterns with limited capability in capturing long-range dependencies from the entire image, hence the limited feature learning in complex global contexts, where large-scene understanding and relationships between far-away visual elements are concerned. By segmenting images into non-overlapping patch sequences and analyzing them with self-attention, ViTs have become the benchmark architecture for feature extraction [3]. ViT features interactions between spatially distant regions, capture contextual information and generate structured visual representations. This study uses ViT as an encoder to extract image features with global contextual awareness and the LSTM network as the decoder to generate sequential captions. The encoder converts raw image data to structured, high-dimensional feature representations. The resultant features condition an LSTM decoder to generate text sequentially, maintaining contextual coherence through autoregressive token prediction.

A large number of existing image captioning models support only high-resource languages such as english, chinese, and a few European languages, while low-resource and classical languages are underrepresented in AI research [4]. Sanskrit remains underserved in modern NLP applications, despite being a classical language with a rich literary heritage and an important part of Nepali culture, philosophy and religious texts.

This study aims to generate Sanskrit captions for daily activity images extracted from the Flickr dataset [5], in an effort to address the significant gap in linguistic resources and technological accessibility. Sanskrit image captioning advances both cultural preservation through digitized access and multilingual NLP through deep learning adaptation for low-resource, morphologically complex languages. The objectives of this study are :

1) To generate a specialized Sanskrit dataset for image captioning, leveraging existing visual resources.
2) To develop a Sanskrit image captioning model utilizing a ViT-LSTM hybrid network.

II. LITERATURE REVIEW

A. Foundational CNN-RNN Architectures

Vinyals et al. [1] proposed the encoder-decoder architecture for image captioning in the “Show and Tell” model. This model combined a CNN encoder for visual feature extraction with an LSTM decoder for natural language generation. With a BLEU-1 score of 59 in the Pascal dataset, the prior state-of-the-art score of 25, while human-level performance reached 69. Xu et al. [6] introduced the “Show, Attend and Tell” model using visual attention mechanisms where the decoder focus on image region to generate caption. This modification significantly improved caption relevance and interpretability, demonstrating the importance of context-dependent feature selection. In another work, Vinyals et al. [7] claimed to refine their approach by adopting stronger CNN backbones (e.g., Inception-v3 and ResNet-based encoders) and optimizing decoder structures in their follow-up work on the MSCOCO Image Captioning Challenge. Gupta [8] experiments a diagnostic study to upgrade visual backbones in CNN-LSTM models without corresponding attention mechanisms. His results show that performance degradation can occur because of information bottlenecks, highlighting the importance of balanced architectural improvements.

B. Vision Transformer-Based Approaches

Dosovitskiy et al. [9] proposed ViT, a major shift in visual representation learning by applying self-attention directly to image patches, removing the need for convolutional operations. The model divides images into fixed-size patches and processes them through transformer encoders, enabling global context modeling across the entire image. Cornia et al. [10] integrated ViT-based encoders into image captioning frameworks using transformer decoders and provided an empirical analysis of explaining transformer-based image captioning models. Liu et al. [11] proposed the Swin Transformer, which uses shifted window attention mechanisms to capture multi-scale features while maintaining computational efficiency. In another work, Liu et al. [12] enhanced the Swin Transformer approach with feature enhancement and multi-stage fusion, demonstrating improved performance on standard benchmarks. Li et al. [13] introduced Object-Semantics Aligned Pre-training, a fully-attentive transformer model that achieved BLEU-4 scores exceeding 40 on the MS COCO benchmark. Zhang et al. [14] proposed VinVL, which revisited visual representations in vision-language models and reported a BLEU-4 of 41.0 on MS COCO, demonstrating the effectiveness of improved visual representations. These studies consistently show that transformer-based architectures surpass traditional CNN-LSTM models.

C. Low-Resource Language Image Captioning

Mishra et al. [15] developed one of the first Hindi image captioning frameworks using deep learning on a translated MS-COCO corpus, representing early attempts to extend image captioning beyond English. Afzal et al. [16] published the first open-access generative image captioning system for Urdu, reporting BLEU-1, BLEU-2, BLEU-3, and BLEU-4 scores of 72.5, 56.9, 42.8 and 31.6, respectively. These works demonstrate that languages with limited resources can still benefit from CNN-RNN or attention-based captioning pipelines. However, due to small datasets and morphological complexity, performance remains substantially behind state-of-the-art English systems, highlighting a clear research gap for under-resourced languages such as Sanskrit.

III. METHODOLOGY

The general flow of the study work is shown in Figure 1. The workflow explains the overall implementation model of the study. The proposed framework follows a systematic approach: a) Image Input, b) Preprocessing, c) ViT Feature Extraction, d) Linear Projection, e) LSTM decoder and f) Caption Generation. The approach takes advantage of the ViT architecture as the encoder [9] , which processes images as sequences of non-overlapping patches through a self-attention mechanism [17]. The complete pipeline processes input images through preprocessing, feature extraction using ViT, linear projection for dimension adjustment, and sequential caption generation through an LSTM-based decoder.

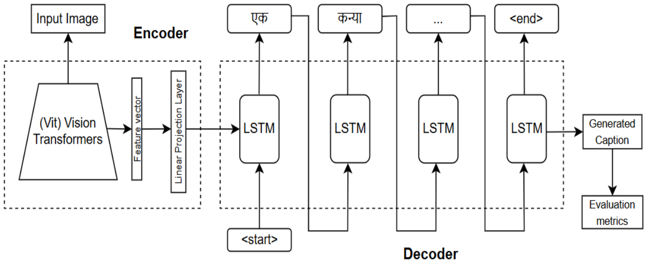


Fig. 1: Methodology of Sanskrit Image Captioning System

A. Dataset Description

The data set is based on Flickr8k [5] , comprising 8,000 images with five English captions each. These captions were translated into Sanskrit and manually verified, resulting in 40,000 pair of image-captions. The images are resized to fixed dimensions and normalized to the [0,1] range. Feature extraction is performed for Vision Transformer compatibility. Captions undergo tokenization and vocabulary building with `<start>` and `<end>` tokens appended to mark sentence boundaries. The data set is partitioned into training, validation, and test sets in the ratio of 80:10:10. Sample image, english caption ad sanskrit captions are shown in Table I. The dataset we translated is publicly available in https://www.kaggle.com/datasets/mohanbhandari/sanskrit-image-captioning

B. Vision Transformer (ViT) Architecture

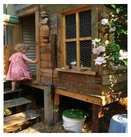
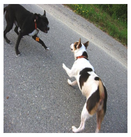
ViT operates on fixed-size image patches, allowing the model to capture long-range dependencies and spatial relationships effectively. An input image $I \in \mathbb{R}^{H\times W\times C}$ is divided into non-overlapping patches of size $P \times P$, producing

$$N = \frac{H \times W}{P^2} \tag{1}$$

patches. Each patch is flattened and linearly projected. Positional embeddings are added to preserve spatial ordering.

The encoder is composed of multiple Transformer blocks, each containing a Multi-Head Self-Attention (MHSA) module

TABLE I: FLICKR-8K: Representative images with their respective english and sanskrit caption

Image	English Caption	Sanskrit Caption
	A child in a pink dress is climbing stairs.	गुलाबि-वस्त्रधारिणी बालिका सीढ्याः आरोहति।
	A girl going into a wooden building.	एका बालिका काष्ठभवनं प्रविशति।
	A little girl climbing a wooden playhouse.	एका बालिका काष्ठक्रीडागृहं प्रति आरोहति।
	A girl climbing stairs to her playhouse.	बालिका क्रीडागृहं प्रति सोपानम् आरोहति।
	A girl in a pink dress entering a wooden cabin.	गुलाबि-वस्त्रधारिणी बालिका काष्ठगृहं प्रविशति।
	A black dog and a spotted dog are fighting.	कृष्णः कुक्कुरः स्पाटड्-कुक्कुरेण सह युध्यति।
	A black dog and a tri-colored dog playing.	मार्गे द्वौ कुक्कुरौ क्रीडन्ति।
	A black dog and a white-brown spotted dog staring at each other.	द्वौ कुक्कुरौ परस्परं पश्यन्ते।
	Two dogs of different breeds looking at each other.	भिन्नजातीयौ द्वौ कुत्रौ परस्परं पश्यतः।
	Two dogs on pavement moving toward each other.	पादमार्गे द्वौ कुत्रौ परस्परं प्रति गच्छतः।

and a Feed-Forward Network (FFN). Self-attention is computed from the matrices Query (Q), Key (K) and Value (V) obtained through learned linear projections.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \tag{2}$$

The output of multiple attention heads is concatenated to form a complete feature representation. Each Transformer block incorporates Layer Normalization and residual connections to stabilize training. Finally, the extracted features are projected to the required dimensionality through a linear projection layer to ensure compatibility with the decoder.

C. LSTM Decoder Architecture

Following feature extraction, an LSTM-based decoder generates captions sequentially. The decoder receives the embedding of the previously generated word as its recurrent input. The vector of visual characteristics v from the ViT encoder is used to initialize the initial hidden state h_0 and the cell state c_0 (as shown in Section 3.2). At each time step t, the LSTM updates its hidden state h_t and cell state c_t according to the standard formulation [18]:

$$h_t, c_t = \text{LSTM}(E(y_{t-1}), h_{t-1}, c_{t-1}), \tag{3}$$

where $E(y_{t-1})$ represents the embedding of the previous word. The decoder computes a probability distribution over the vocabulary through a fully connected output layer:

$$p(y_t \mid h_t) = \text{softmax}(W_o h_t + b_o). \tag{4}$$

The word with the highest probability is selected as the next predicted token and fed back into the decoder, enabling autoregressive caption generation. This process continues until the end-of-sequence token is produced or the maximum caption length is reached. Training is performed using the Cross-Entropy Loss function, which is minimized over the sequence length T:

$$\mathcal{L} = -\sum_{t=1}^{T} \mathbb{I}(y_t \neq \langle\text{pad}\rangle) \log p(y_t^*), \tag{5}$$

where y_t^* denotes the ground-truth word in the time step t, and $\mathbb{I}(\cdot)$ is the indicator function that ensures that the loss is not calculated for the padding token $\langle\text{pad}\rangle$. The model parameters are optimized through backpropagation to align the generated captions with the reference descriptions.

D. Architecture Implementation

The ViT encoder provides efficient feature extraction capabilities, while the LSTM decoder enables fine-tuned sequential caption generation in Sanskrit. The combination leverages complementary strengths of vision processing through transformers and language modeling through recurrent networks. The motivation behind selecting the ViT-LSTM architecture lies in addressing the need for capturing long-range dependencies in images while maintaining sequential coherence in generated Sanskrit captions. ViT achieves comprehensive visual understanding through self-attention mechanisms that process global image context, unlike CNNs which focus on local features. The LSTM component retains contextual information from previously generated words while predicting subsequent words, ensuring grammatical coherence and semantic meaningfulness in generated Sanskrit captions.

E. Experimental Setup

The proposed ViT–LSTM framework is implemented using Python and the PyTorch deep learning library [19], using TorchVision for preprocessing steps such as resizing and normalization. Dataset management, including image-caption pairing and partitioning, was handled using Pandas. The data set, consisting of Sanskrit captions paired with images from Flickr8k, was split into training , validation and test sets. The model was optimized using the Cross-Entropy Loss function, with the validation set used to monitor learning trends and implement early stopping to mitigate overfitting. Model performance was quantitatively evaluated using standard natural language generation metrics, including BLEU and ROUGE scores, which compare generated captions against ground-truth descriptions.

F. Evaluation Metrics

To quantitatively assess the performance of the proposed ViT–LSTM model for Sanskrit image captioning, standard natural language generation metrics are used

1) Bilingual Evaluation Understudy: Bilingual valuation under study (BLEU) [20] is a precision-based metric that evaluates the similarity between a generated caption C and a set of reference captions $\{R_1, R_2, \ldots, R_k\}$ based on the overlap of n-grams. BLEU-4, which can accommodate up to 4-grams, is commonly used for image captioning. The BLEU score is computed as:

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right), \tag{6}$$

where p_n is the modified precision for n-grams, w_n are positive weights summing to one (typically $w_n = 1/N$), and BP is the brevity penalty:

$$\text{BP} = \begin{cases} 1, & |C| > |R| \\ \exp(1 - |R|/|C|), & |C| \leq |R| \end{cases} \tag{7}$$

with $|C|$ and $|R|$ representing the lengths of the candidate and reference captions, respectively.

2) Recall-Oriented Understudy for Gisting Evaluation: Recall-Oriented Understudy for Gisting Evaluation (ROUGE-L) [21] is a recall-based metric that measures the overlap between the candidate and reference captions using the longest common subsequence (LCS). ROUGE-L considers both precision (P_{LCS}) and recall (R_{LCS}), and reports an F-measure:

$$R_{LCS} = \frac{\text{LCS}(C, R)}{|R|}, \quad P_{LCS} = \frac{\text{LCS}(C, R)}{|C|}, \tag{8}$$

$$\text{ROUGE-L} = \frac{(1 + \beta^2) R_{LCS} P_{LCS}}{R_{LCS} + \beta^2 P_{LCS}}, \tag{9}$$

where $\text{LCS}(C, R)$ is the length of the longest common subsequence between candidate C and reference R, and β is typically set to 1 to give equal weight to precision and recall.

IV. Analysis and Discussion

A. Training and Validation Performance

The training and validation performance of the proposed model is shown in Figure 1, which displays the loss curves for the training epochs. The training loss shows a consistent decreasing trend from approximately 5.07 to 4.12, while the validation loss decreases from 4.45 to 3.84.

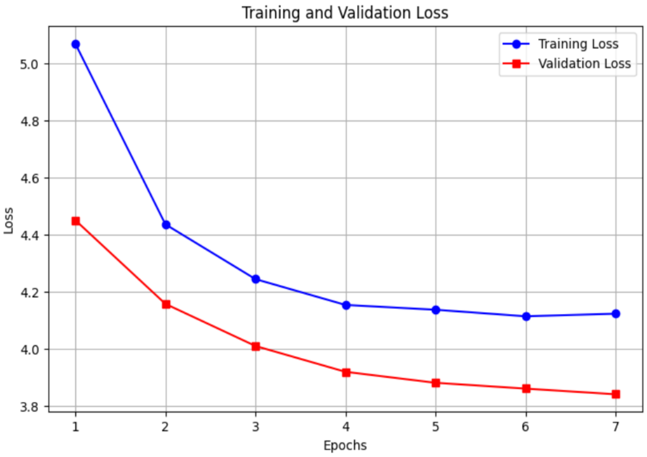


Fig. 1: Training and Validation Loss

B. BLEU Score Performance

The evaluation of the BLEU score, Figure 2, measures the quality of the Sanskrit captions generated by comparing them with reference translations. The BLEU-4 score demonstrates remarkable and consistent improvement throughout the seven training epochs, starting from 0.0361 and reaching 0.0639, representing a 77.0% relative improvement. The detailed analysis of the n-grams reveals progressive improvements at all levels: BLEU-1 scores increased from 0.2652 to 0.3082, BLEU-2 from 0.1430 to 0.1843, BLEU-3 from 0.0768 to 0.1115, and BLEU-4 from 0.0361 to 0.0639. While the absolute BLEU-4 score of 6.39% remains modest. Results are in Table II.

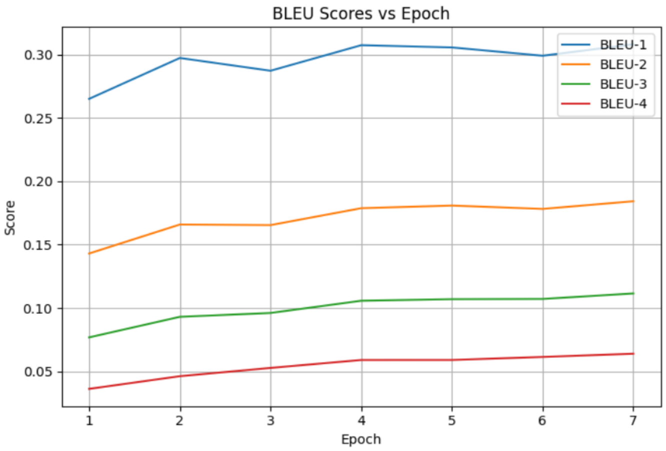


Fig. 2: BLEU score progression

C. ROUGE-L Score Performance

The ROUGE-L score, shown in Figure 3, measures the longest common subsequence between the generated and reference captions, evaluating the structural and semantic similarity. The score shows a progressive and substantial improvement from 0.2967 to 0.3472, representing an improvement of 17.0%. Additional ROUGE metrics (Table II) further validate this improvement: ROUGE-1 increased from 0.3095 to 0.3623, and ROUGE-2 improved from 0.0946 to 0.1290, demonstrating improved n-gram overlap at multiple levels. The ROUGE-L score of 34.72% indicates that the model achieves reasonable structural alignment with reference captions, successfully capturing approximately one-third of the sequence patterns in ground truth descriptions.

TABLE II: Performance Metrics Across Training Epochs

Epoch	Train Loss	Val Loss	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L
1	5.0691	4.4509	0.2652	0.1430	0.0768	0.0361	0.3095	0.0946	0.2967
2	4.4366	4.1579	0.2974	0.1659	0.0930	0.0461	0.3264	0.1062	0.3133
3	4.2448	4.0104	0.2873	0.1654	0.0960	0.0527	0.3379	0.1138	0.3240
4	4.1543	3.9199	0.3075	0.1788	0.1057	0.0589	0.3455	0.1197	0.3310
5	4.1374	3.8815	0.3058	0.1809	0.1070	0.0589	0.3539	0.1256	0.3390
6	4.1145	3.8610	0.2992	0.1782	0.1071	0.0613	0.3543	0.1257	0.3402
7	4.1234	3.8413	0.3082	0.1843	0.1115	0.0639	0.3623	0.1290	0.3472

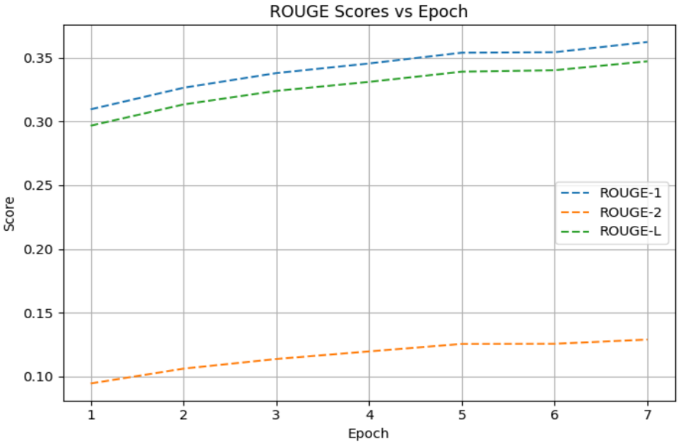
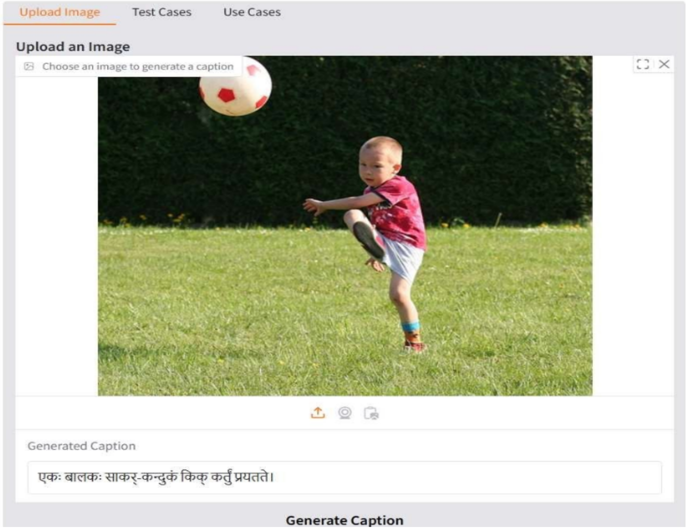


Fig. 3: ROUGE Score Progression across Training Epochs

V. MODEL DEVEOPMENT

The trained ViT-LSTM model is serialized and saved in .pth file. A web application is developed using the Gradio framework, providing an interactive interface where users can upload images and view the generated Sanskrit captions in real-time. Upon image upload, the backend processes the input through the image preprocessing module, which resizes and normalizes the image. Figure 4 shows the interface.



Generated Caption

एकः बालकः साकर्-कन्दुकं किक् कर्तुं प्रयतते।

Fig. 4: Model Deployment

VI. CONCLUSION AND IMPLICATION

This study demonstrated the capability of deep learning in generating Sanskrit image descriptions through a ViT-LSTM based image captioning system. The model effectively relates visual features to meaningful textual representations, providing scope for research in multimodal AI applications for low-resource languages. Although the system performs effectively, certain limitations were identified. Dataset quality significantly impacts caption accuracy, and the model occasionally fails on complex or less frequent Sanskrit words. Computational constraints also affect overall performance. Despite these challenges, the results demonstrate that transformer-based encoders combined with sequence-generating decoders can effectively bridge the gap between vision and language in Sanskrit NLP applications. Further improvements are possible through data set expansion with more diverse images and captions to enhance vocabulary coverage and reduce generic predictions. Architectural modifications, such as replacing the LSTM with transformer-based decoders or fine-tuning larger ViT models, could improve the quality of caption generation. Hyperparameter optimization presents additional opportunities for performance enhancement. The deployment through web-based or mobile applications would enable real-world usage and broader accessibility. Future research directions include real-time caption generation, multilingual support, and integration with assistive technologies, contributing to the preservation and digital integration of Sanskrit language in modern AI applications.

REFERENCES

[1] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, “Show and Tell: A Neural Image Caption Generator,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3156-3164.

[2] G. Keren and B. Schüller, “Convolutional RNN: an enhanced model for extracting features from sequential data,” in *2016 International Joint Conference on Neural Networks (IJCNN)*, Vancouver, BC, Canada, 2016, pp. 3412–3419.

[3] L. Scabini et al., “A comparative survey of vision transformers for feature extraction in texture analysis,” *Journal of Imaging*, vol. 11, no. 9, p. 304, 2025.

[4] T. Zhong et al., “Opportunities and challenges of large language models for low-resource languages in humanities research,” *arXiv preprint arXiv:2412.04497*, 2024.

[5] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, “From Image Descriptions to Visual Denotations,” 2014. [Online]. Available: https://www.kaggle.com/datasets/adityajn105/flickr8k

[6] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention,” in *Proc. 32nd Int. Conf. Mach. Learn. (ICML)*, Lille, France, Jul. 2015, pp. 2048–2057.

[7] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, “Show and Tell: Lessons learned from the 2015 MSCOCO Image Captioning Challenge,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 652–663, Apr. 2017.

[8] R. K. Gupta, “When better eyes lead to blindness: A diagnostic study of the information bottleneck in CNN-LSTM image captioning models,” *arXiv preprint arXiv:2507.18788*, Jul. 2025.

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021.

[10] M. Cornia, L. Baraldi, and R. Cucchiara, “Explaining transformer-based image captioning models: An empirical analysis,” *AI Commun.*, vol. 35, no. 2, pp. 111–129, 2022.

[11] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, “Swin transformer: Hierarchical vision transformer using shifted windows,” in *Proc. IEEE/CVF Int. Conf. Comput. Vision (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 9992–10002.

[12] L. Liu, Y. Jiao, X. Li, J. Li, H. Wang, and X. Cao, “Swin transformer-based image captioning with feature enhancement and multi-stage fusion,” in *Proc. 19th Int. Conf. Natural Comput., Fuzzy Syst. Knowledge Discovery (ICNC-FSKD)*, Harbin, China, Jul. 2023, pp. 1–7.

[13] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, Y. Choi, and J. Gao, “Oscar: Object-semantics aligned pre-training for vision-language tasks,” in *Proc. 16th Eur. Conf. Comput. Vision (ECCV)*, Glasgow, UK, Aug. 2020, pp. 121–137.

[14] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, and J. Gao, “VinVL: Revisiting visual representations in vision-language models,” in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit. (CVPR)*, Nashville, TN, USA, Jun. 2021, pp. 5575–5584.

[15] S. K. Mishra, R. Dhir, S. Saha, and P. Bhattacharyya, “A Hindi image caption generation framework using deep learning,” *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 20, no. 2, pp. 1–19, Mar. 2021, doi: 10.1145/3432246.

[16] M. K. Afzal, M. Shardlow, S. Tuarob, F. Zaman, R. Sarwar, M. Ali, N. R. Aljohani, M. D. Lytras, R. Nawaz, and S. Hassan, “Generative image captioning in Urdu using deep learning,” *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 6, pp. 7719–7731, Jun. 2023, doi: 10.1007/s12652-023-04584-y.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances Neural Inf. Process. Syst.*, vol. 30, 2017.

[18] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[19] A. Paszke et al., ”PyTorch: An Imperative Style, High-Performance Deep Learning Library,” in Adv. Neural Inf. Process. Syst., 2019.

[20] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “BLEU: A Method for Automatic Evaluation of Machine Translation,” in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, Philadelphia, PA, USA, 2002, pp. 311–318.

[21] C.-Y. Lin, “ROUGE: A Package for Automatic Evaluation of Summaries,” in *Text Summarization Branches Out, Post-Conference Workshop of ACL 2004*, Barcelona, Spain, 2004, pp. 74–81.