Available online at website: https://jurnal.iaii.or.id/index.php/RESTI

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 10 No. 5 (2026) 1361 - 1370 e-ISSN: 2580-0760

Leveraging Assistive Image Captioning on VizWiz through Multi-Caption Supervision

Fauzan Firdaus[1*], Angel Metanosa Afinda[2], Edward Ferdian[3], Muhammad Maulana Yusuf[4], Ema Rachmawati[5]
[1]Software Engineering Study Program, School of Computing, Telkom University, Bandung, Indonesia
[2]Computer Engineering Study Program, School of Electrical Engineering, Telkom University, Bandung, Indonesia
[3,4,5]Informatics Study Program, School of Computing, Telkom University, Bandung, Indonesia

[*]*fauzanfirdausff@telkomuniversity.ac.id*

Abstract

Most image captioning datasets provide several human captions for the same image, yet model training is frequently carried out in a way that effectively treats each image as having only one caption. Our study revisits this mismatch by examining the impact of training with multiple captions per image on caption generation performance. For this purpose, we utilized the VizWiz-Captions dataset, which reflects a real assistive scenario with images taken by blind or visually impaired users with five reference captions per image. We evaluated two conventional CNN-based encoder–decoder models, namely CNN–LSTM and CNN–GRU. In both models, the visual encoder was fixed using InceptionV3 image features, while the recurrent decoder was varied between LSTM and GRU. We performed experiments by using training data by varying the number of captions used per image from one to five (k = 1–5), while keeping the overall pipeline unchanged. The models were evaluated using standard multi-reference metrics (BLEU-1 to BLEU-4, CIDEr, and SPICE). Across both baselines, performance generally improves when more captions are included in training, although the gains are not perfectly monotonic for every metric. The strongest results are observed at k = 5. For CNN–LSTM, the best setting achieved BLEU-4 = 0.175, CIDEr = 0.206, and SPICE = 0.120. On the other hand, CNN–GRU achieved BLEU-4 = 0.192, CIDEr = 0.226, and SPICE = 0.119. These findings suggest that, even without architectural changes, simply leveraging the available caption diversity during training can provide a practical boost for VizWiz-style captioning tasks.

Keywords: image captioning; multi-caption training; VizWiz dataset

How to Cite: F. Firdaus, E. Rachmawati, A. M. Afinda, E. Ferdian, and M. M. Yusuf, "Leveraging Assistive Image Captioning on VizWiz through Multi-Caption Supervision", *J. RESTI (Rekayasa Sist. Teknol. Inf.)* , vol. 10, no. 5, pp. 1361–1370, Oct. 2026, doi: 10.29207/resti.v10i5.7472.
Permalink/DOI: https://doi.org/10.29207/resti.v10i5.7472

Received: January 10, 2026
Accepted: September 5, 2026
Available Online: October 7, 2026

This is an open-access article under the CC BY 4.0 License
Published by Ikatan Ahli Informatika Indonesia

1. Introduction

In an ideal situation, image captioning is performed on a complete image or scene. However, for images that contain only partial information or incomplete scenes, image captioning is sometimes not very accurate. The VizWiz [1] ecosystem highlights this reality because its images are captured by blind and visually impaired users in uncontrolled conditions (blur, poor framing, occlusions, unusual viewpoints), which makes captioning substantially harder than clean web imagery, where existing vision-language models (VLMs) are forced to be robust to real-world noise [2]. Furthermore, this dataset provides multiple human captions per image, creating an opportunity: instead of treating captions as redundant labels, we can treat them as complementary descriptions that expose different valid perspectives, objects vs. actions, global context vs. local detail, or uncertainty vs. specificity [3], [4].

In parallel, the broader captioning landscape has been reshaped by large-scale vision-language pretraining and multimodal instruction tuning. Contrastive pretraining (e.g., CLIP-style objectives) has changed how visual features are learned and transferred across tasks, including captioning [5]. More recent systems pair strong image encoders with large language models (LLMs) to generate fluent captions (and more), often achieving impressive zero-shot or few-shot performance [6-8]. This shift has also triggered a wave of studies questioning whether common captioning metrics still reflect what we care about, especially when captions can be plausible yet mismatched, stylistically different, or overly generic [9], [10]. Newer

benchmarks and arena-style evaluations have emerged to stress-test caption quality under modern model behavior and to compare systems more directly [11], [12].

Despite these advances, there is still practical value in carefully controlled studies using classical encoder-decoder models. Transformer-based captioners and multimodal LLMs introduce many moving parts, including pretraining corpora, instruction data, decoding heuristics, and alignment objectives, which can obscure the specific contribution of training supervision design, including how multiple reference captions per image are used [13], [14]. In this context, a Convolutional Neural Network-Recurrent Neural Network (CNN-RNN) captioning model refers to an encoder-decoder architecture in which a Convolutional Neural Network (CNN) extracts visual features and a Recurrent Neural Network (RNN) generates the caption word by word. The RNN decoder can be implemented using Long Short-Term Memory (LSTM) or Gated Recurrent Unit (GRU) units. Previous multi-reference captioning research suggests that how references are sampled and smoothed during training can significantly influence learning dynamics and generalization [3]. For noisy assistive datasets like VizWiz, this issue becomes more important because captions can vary widely in completeness and correctness, and robustness to caption noise becomes a core requirement rather than an edge case [2], [15].

However, the specific effect of increasing the number of captions per image during training remains insufficiently isolated in VizWiz-style assistive captioning. Existing studies often focus on stronger architectures or broader evaluation issues, while less attention is given to whether simply using more of the available caption diversity during training can improve classical captioning baselines under the same pipeline. Therefore, this study examines the effect of increasing the number of captions per image used during training, from one to five, while keeping the model family deliberately classical. Classical CNN-RNN baselines are used in this study to keep the experimental setting controlled and to isolate the effect of multi-caption supervision from the additional complexity introduced by large-scale multimodal pretraining, instruction tuning, or advanced decoding strategies [16]. Specifically, we use CNN-RNN encoder-decoder baselines with fixed InceptionV3 visual features and either LSTM or GRU decoders, allowing the effect of multi-caption supervision to be examined without changing the visual encoder, decoding strategy, or overall training pipeline [17], [18]. Since evaluation is conducted using the full set of available references, this study also relies on standard multi-reference captioning metrics, while acknowledging prior findings on metric sensitivity, reference diversity, diminishing returns from additional references, instability under changing reference sets, and the distinction between caption correctness and helpfulness in accessibility contexts [9], [19-23].

This study contributes to the literature in three ways. First, it provides a controlled empirical analysis of multi-caption supervision for assistive image captioning on VizWiz-Captions. Second, it shows how captioning performance changes when the number of training captions per image is varied while the architecture, visual features, decoding strategy, and evaluation setting are kept consistent. Third, it compares two classical recurrent decoders, LSTM and GRU, under the same CNN-based visual feature pipeline, thereby clarifying whether the caption diversity already available in VizWiz-Captions can provide practical performance gains without requiring architectural changes or large-scale multimodal pretraining.

2. Methods

Our proposed method consists of an end-to-end experimental pipeline for evaluating the effect of multi-caption supervision in image captioning. Figure 1 summarizes the pipeline used in this study.

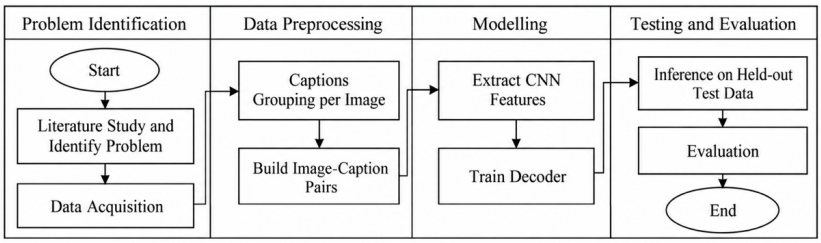


Figure 1. Proposed pipeline using multi-caption supervision for image captioning.

The process started with data preprocessing, where image-caption annotations were grouped per image according to the annotated VizWiz-Captions splits used in this study. The official training split was used for model training, while the official validation split was partitioned before training into a dev-validation subset and an internal held-out test subset. For each scenario $k \in \{1,2,3,4,5\}$, the training set was constructed by selecting k captions for each image and converting them into image-caption pairs. The modelling stage used InceptionV3 as a CNN feature extractor to obtain visual representations, while LSTM and GRU decoders were trained to generate captions. In the testing and evaluation stage, captions were generated for the internal held-out test images and evaluated using standard image captioning metrics, including BLEU-1 to BLEU-4, CIDEr, and SPICE. The results from different k values were then compared to measure how

the number of captions used during training affects captioning performance.

2.1 Dataset

Table 1 summarizes the VizWiz-Captions [1] split statistics. The dataset contains 39,181 images, each paired with 5 captions, and is distributed into standard splits. Specifically, it provides 23,431 training images with 117,155 captions, 7,750 validation images with 38,750 captions, and 8,000 test images with 40,000 captions.

Table 1. Annotated VizWiz-captions data split used in this study

Split	Images	Captions	Notes
Official train	23,431	117,155	Training
Dev-validation	6,190	30,950	Early stopping, model checkpointing, and learning-rate scheduling
Internal held-out test	1,560	7,800	Final BLEU, CIDEr, and SPICE evaluation
Official test	8,000	Not locally available	Not used for metric computation

Because the locally available VizWiz-Captions test split does not include ground-truth annotations, BLEU, CIDEr, and SPICE could not be computed on the official test split. Therefore, this study used the official training split for training and partitioned the official validation split before training into a dev-validation subset and an internal held-out test subset. Specifically, 23,431 official training images were used for training, while the 7,750 official validation images were divided into 6,190 dev-validation images and 1,560 internal held-out test images. The dev-validation subset was used for early stopping, model checkpointing, and learning-rate scheduling, whereas the internal held-out test subset was used only for final evaluation. This produced an approximate 75:20:5 ratio over the annotated data used in this study.

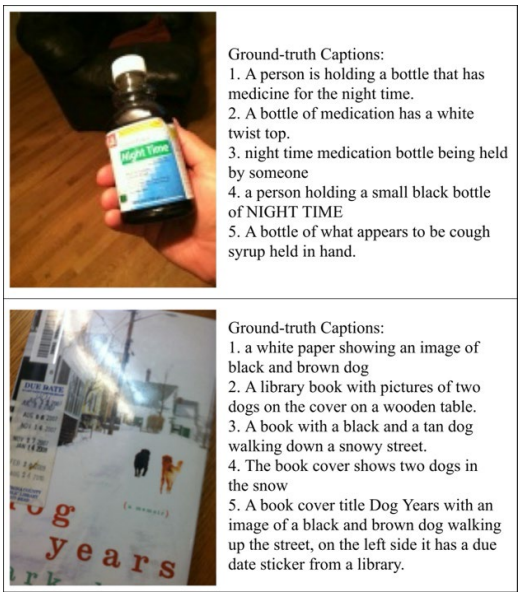


Figure 2. Representative examples from VizWiz-Captions with five ground-truth captions per image

Firstly, we grouped the annotations so that each image was paired with a list of up to five captions, forming the basis for multi-caption training. Figure 2 shows representative examples from VizWiz-Captions along with their five ground-truth captions per image.

The examples illustrate two important characteristics of the dataset. First, the image reflects real-world assistive capture conditions, such as blur, poor framing, occlusion, or challenging lighting. Second, the human-written captions for the same image often differ in wording, focus, and level of detail. For example, one caption may describe the main object, while another may mention the surrounding context or express uncertainty about the scene. This caption diversity supports our investigation into whether using multiple captions per image during training can improve captioning performance under noisy and assistive visual conditions.

2.2 Convolutional Neural Network

To represent the visual content of each image, we utilized a Convolutional Neural Network (CNN) as a fixed feature extractor [24]. Specifically, we used InceptionV3, which was pretrained on ImageNet and removed its classification head, while applying global average pooling to obtain a compact, fixed-length embedding (d = 2048). Input images were resized to 299×299 pixels, followed by standard InceptionV3 preprocessing, resulting in a 2048-dimensional feature vector. Figure 3 depicts the visual feature extraction stage, where each input image is encoded by a frozen InceptionV3 backbone with global average pooling to produce a fixed 2048-dimensional representation that conditions the subsequent caption decoder.

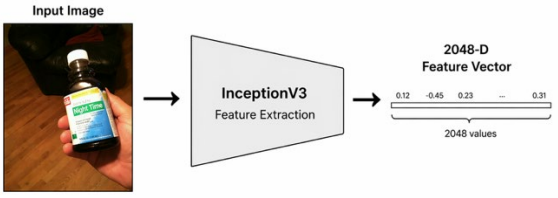


Figure 3. Feature Extraction with an input image of 299x299 pixels

In our experiments, the CNN weights were kept frozen to reduce computational cost and to ensure that performance differences across scenarios primarily reflect the effect of multi-caption supervision rather than changes in visual feature learning. The extracted features were computed once per image before training the necessary components and reused across all training scenarios to keep the experimental pipeline consistent and efficient.

2.3 Long Short-Term Memory (LSTM)

In the CNN-LSTM captioning model, sequence generation is handled by a Long Short-Term Memory (LSTM) network that models word dependencies over time while being conditioned on the image representation [17], [25]. Given an image feature vector extracted by the CNN, we first apply dropout and a fully

connected layer to project the visual feature into a lower-dimensional embedding.

In parallel, a partial caption prefix is converted to token indices and passed through an embedding layer (with masking for padded tokens), followed by dropout and an LSTM layer to produce a language-context vector. To fuse visual and textual information, we use an element-wise additive combination [26], [27] between the projected image embedding and the LSTM output, as defined in Equation 1.

$$\mathrm{z}_t = \mathrm{v}_i' + \mathrm{h}_t \quad (1)$$

In Equation 1, z_t denotes the fused vector, v_i' denotes the projected visual embedding, and h_t denotes the LSTM language context. The fused vector is then passed through an additional fully connected layer with ReLU activation and mapped to a vocabulary distribution using a softmax layer.

Training follows a next-word prediction objective with teacher forcing, implemented by generating supervised prefix-target pairs from each caption sequence. The model is optimized using categorical cross-entropy over the vocabulary. To mitigate overfitting and stabilize learning, we apply dropout in both the visual and language branches and L2 regularization on the dense and recurrent layers. During inference, captions are produced with greedy decoding.

2.4 Gated Recurrent Units (GRU)

The CNN-GRU model follows the same overall encoder-decoder structure as the CNN-LSTM setup, but replaces the LSTM decoder with a Gated Recurrent Unit (GRU). Dropout and a linear projection applied to obtain a visual embedding. In parallel, the input caption prefix is tokenized, padded to a fixed length, and mapped to a sequence of word embeddings. The GRU processes this embedded sequence and produces a language-context vector [28], [29].

Compared to LSTM, GRU uses a simpler gating mechanism, consisting mainly of update and reset gates, to regulate information flow across time steps. This architectural simplicity makes GRU a relevant decoder alternative for comparison with LSTM under the same CNN-based captioning pipeline. In our implementation, the GRU output is fused with the projected visual embedding using element-wise addition, same as the CNN-LSTM setup. The fused representation is then transformed by a fully connected layer with ReLU activation and mapped to a probability distribution over the vocabulary through a softmax layer.

2.5 Multi-caption Training

To study the effect of the number of captions during model training, we define $k \in \{1,2,3,4,5\}$ as the number of captions used per image. For a given image I_i with a reference set of captions c_{ik}, the training supervision for scenario (k) is defined in Equation 2.

$$\mathcal{D}_k = \{(I_i, c_{i1}), (I_i, c_{i2}), \dots, (I_i, c_{ik})\} \quad (2)$$

In Equation 2, $\mathcal{D}_k$ denotes the image-caption pairs constructed by using the first (k) captions available for image I_i. This first-(k) strategy supports reproducibility, but it is also a limitation because it does not evaluate all possible caption combinations for each image. Therefore, the results should be interpreted as a controlled prefix-based comparison rather than an average performance over randomly sampled caption subsets.

Each selected caption was converted into a token sequence and wrapped with special boundary tokens at the start and end of the sequence. Training was performed using a standard next-word prediction objective with teacher forcing. For a tokenized caption $c_1, c_2, \dots, c_T$, the model learns to predict the next token given the image representation and a partial prefix, as defined in Equation 3.

$$p(\, c_t \mid I, c_{1:t-1}\,),\ t = 2, \dots, T \quad (3)$$

In Equation 3, c_t denotes the target token, I denotes the image representation, and $c_{1:t-1}$ denotes the preceding caption tokens used as the prefix. Concretely, from each caption, we generate multiple supervised samples by pairing a padded prefix sequence $c_{1:t-1}$ as input with the target token c_t as output. This formulation allows the decoder, either LSTM or GRU, to learn language dynamics while being conditioned on the fixed CNN feature vector extracted from the corresponding image.

While the training supervision varies across k, the dev-validation and internal held-out test subsets remain unchanged across scenarios. During final evaluation, generated captions are compared against the full set of reference captions available for each internal held-out test image.

2.6 Evaluation

The evaluation in this study used standard automatic image captioning metrics, namely BLEU-1 to BLEU-4, CIDEr, and SPICE [30-32]. Since image captioning is a sentence generation task, the generated captions were evaluated by comparing them with the available human-written reference captions. These metrics were selected because they capture different aspects of caption quality: BLEU measures n-gram overlap, CIDEr measures consensus with reference captions, and SPICE measures semantic content based on objects, attributes, and relations.

BLEU evaluates the precision of n-gram overlap between a generated caption and reference captions. BLEU-N is computed as defined in Equation 4.

$$BLEU_N = BP \cdot \exp(\textstyle\sum_{n=1}^{N} w_n log p_n) \quad (4)$$

In Equation 4, p_n is the modified precision for n-grams, w_n is the weight for each n-gram order, and BP is the brevity penalty used to penalize captions that are too short. BLEU-1 to BLEU-4 were used to evaluate overlap from unigram-level similarity to longer phrase-level similarity. BLEU-1 captures word-level overlap,

while BLEU-4 provides a stricter evaluation of longer phrase consistency.

CIDEr measures how well a generated caption agrees with the consensus of human reference captions. Unlike BLEU, CIDEr applies TF-IDF weighting to n-grams so that more informative words have greater influence than common words. The CIDEr score can be expressed as defined in Equation 5.

$$CIDEr(c_i, S_i) = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{m}\sum_{j=1}^{m}\frac{g^n(c_i)\cdot g^n(s_{ij})}{\|g^n(c_i)\|\|g^n(s_{ij})\|} \quad (5)$$

In Equation 5, c_i is the generated caption, S_i is the set of reference captions, s_{ij} is the j-th reference caption, m is the number of reference captions, and $g^n(\cdot)$ represents the TF-IDF vector of n-grams. CIDEr was used because the VizWiz-Captions dataset provides five human captions per image, making consensus-based evaluation highly relevant.

SPICE evaluates caption quality by converting both generated and reference captions into semantic scene graphs. These graphs represent semantic propositions such as objects, attributes, and relations. In this study, SPICE is computed using the F-score between semantic tuples extracted from the generated caption and the reference captions, as defined in Equation 6. The corresponding precision and recall terms are defined in Equations 7 and 8, respectively.

$$SPICE = F_1 = \frac{2\cdot Precision\cdot Recall}{Precision+Recall} \quad (6)$$

$$Precision = \frac{|T(c)\cap T(S)|}{|T(c)|} \quad (7)$$

$$Recall = \frac{|T(c)\cap T(S)|}{|T(S)|} \quad (8)$$

In Equation 7, $T(c)$ is the set of semantic tuples extracted from the generated caption and $T(S)$ is the set of semantic tuples extracted from the reference captions. SPICE was included because it focuses more on semantic correctness than surface-level word overlap, which is important for assistive image captioning, where different valid captions may use different wording.

Final evaluation was performed on the internal held-out test subset of 1,560 images. This subset was separated from the official validation split before training and was not used for early stopping, model checkpointing, learning-rate scheduling, or model selection. All generated captions from the internal held-out test images were evaluated against the full set of available ground-truth captions. The use of multiple metrics is important because no single automatic metric fully captures caption quality. Therefore, BLEU, CIDEr, and SPICE were used together to provide complementary evidence of captioning performance across different values of k.

3. Results and Discussions

The following section reports and interprets the experimental findings of our multi-caption training study on VizWiz-Captions.

3.1 Experimental Setup Summary

Table 2 summarizes the experimental configuration and key hyperparameters used across all scenarios. For the captioning architecture, we implemented two classical baselines: CNN+LSTM and CNN+GRU.

Table 2. Experimental Setup

Component	Setting
GPU	NVIDIA A100
CPU	Google Colab virtual CPU
Python	3.11.13
TensorFlow/Keras	TensorFlow 2.18.0 / tf.keras
Dataset	VizWiz-Captions
Used data	Official train, dev-validation, and internal held-out test
Multi-caption setting	$k = 1..5$, first-k caption per image
Visual encoder	InceptionV3 pretrained ImageNet, frozen, pooling=avg
Image input	Resize 299x299
Decoder	LSTM/GRU (hidden size 256)
Tokenization	Keras Tokenizer
Training objective	Next-word prediction (teacher forcing), categorical cross-entropy
Optimizer	Adam
Epochs	Up to 100 + EarlyStopping (patience=10)
Batch size	128
Regularization	Dropout 0.5 + L2=1e-4
LR schedule	ReduceLROnPlateau (factor 0.5, patience 4)
Decoding	Greedy
Metrics	BLEU-1, BLEU-2, BLEU-3, BLEU-4, CIDEr, SPICE

In both models, visual features were extracted using a pretrained InceptionV3 backbone (frozen) with global average pooling, producing a fixed 2048-dimensional representation per image. The language decoder was trained with teacher forcing using a next-word prediction objective, and captions were generated at inference time via greedy decoding. Performance was measured on the internal held-out test subset of 1,560 images using standard captioning metrics: BLEU-1 to BLEU-4, CIDEr, and SPICE.

The comparison between CNN-LSTM and CNN-GRU is therefore treated as a controlled decoder comparison rather than a full efficiency benchmark. Both models used the same frozen InceptionV3 visual features, hidden size, tokenization, training objective, and greedy decoding strategy. Although GRU has a simpler recurrent gating structure than LSTM, training and inference time were not used as primary comparison metrics because the experiments were conducted in a managed Google Colab environment where wall-clock runtime can vary across sessions. The main comparison in this study is consequently based on captioning quality metrics, while decoder simplicity is discussed only as architectural context.

3.2 Experimental Results

This section reports the quantitative performance of the proposed multi-caption training strategy under two classical image captioning baselines: CNN+LSTM and CNN+GRU.

The CNN+LSTM baseline shows a generally positive response to increasing the number of captions per image used during training. Performance improves from $k = 1$ to $k = 2$ across all metrics, indicating that even a modest increase in caption supervision provides more informative learning signals. While there was a slight fluctuation in intermediate settings (e.g., small non-monotonic changes around $k = 3$ and $k = 4$ for some metrics), the overall trend remains upward when moving toward the full multi-caption setting.

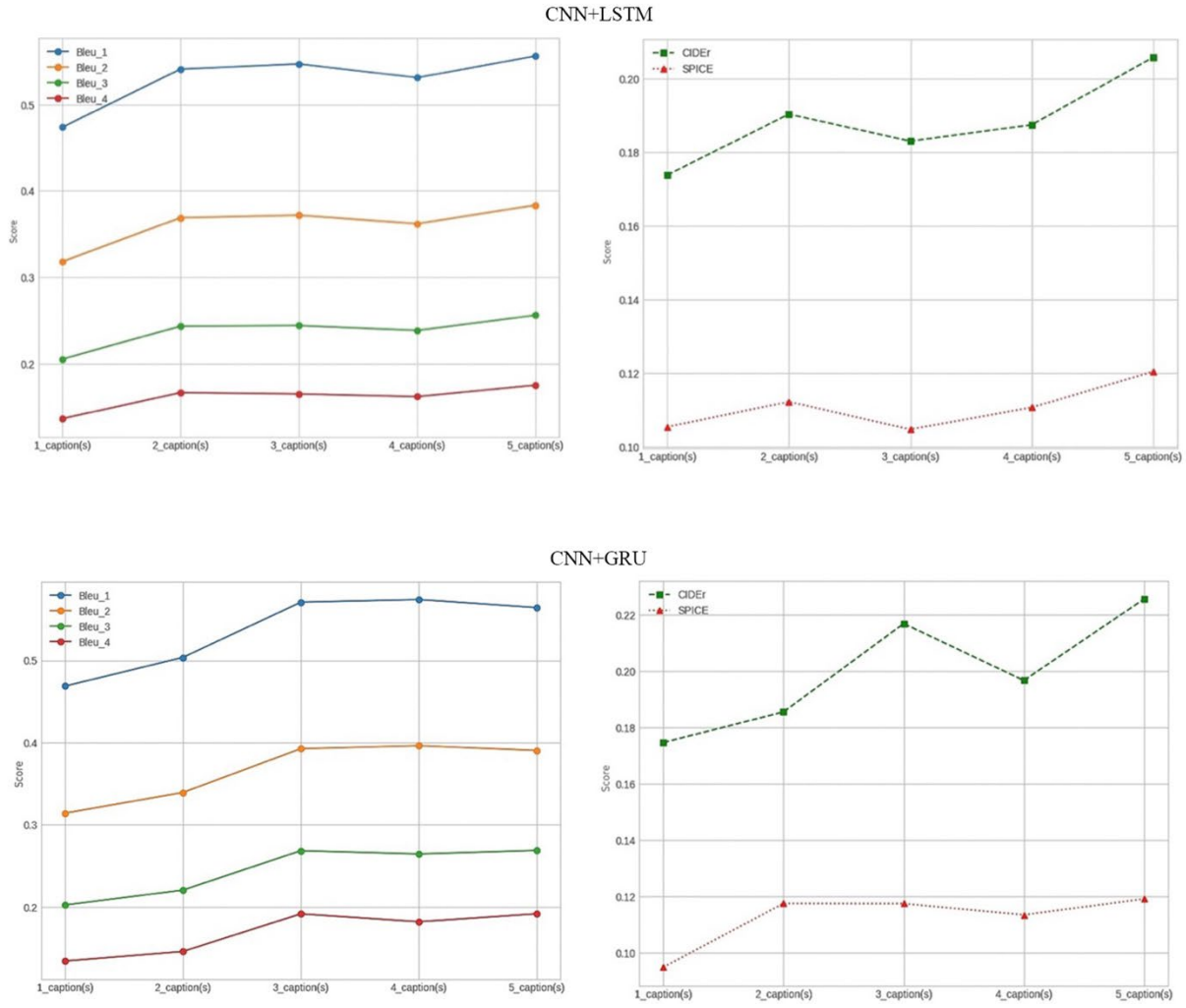


Figure 4. CNN+LSTM and CNN+GRU result comparison for $k = 1, 2, 3, 4$ and 5 in the multi-caption experiments on the internal held-out test set (n = 1,560), where k corresponds to the number of captions used per image during training, covering BLEU-1 to BLEU-4, CIDEr, and SPICE.

Figure 4 summarizes the CNN+LSTM results across the five multi-caption settings on the internal held-out test set (n = 1,560). BLEU-1 to BLEU-4 generally improved as the number of captions increased, with the most noticeable improvement occurring from the single-caption baseline ($k = 1$) to $k = 2$. After that, the gains became more incremental and occasionally fluctuated, but the strongest BLEU profile was still observed at $k = 5$. A similar pattern was also shown by CIDEr and SPICE. CIDEr increased overall and reached its highest value at $k = 5$, while SPICE also peaked at $k = 5$, suggesting improvement not only in wording overlap but also in semantic content captured by scene-graph style evaluation.

Table 3 reports the CNN+LSTM results across multi-caption training scenarios ($k = 1$ to $k = 5$) on the internal held-out test set, measured using BLEU-1 to BLEU-4, CIDEr, and SPICE. The best performance for CNN+LSTM is obtained at $k = 5$, where BLEU-4 reaches 0.175 and CIDEr reaches 0.206, with the highest SPICE score (0.120). Compared to the single-caption setting ($k = 1$), this reflects an improvement from BLEU-4 0.136 to 0.175, CIDEr 0.174 to 0.206, and SPICE 0.105 to 0.120.

A similar pattern is observed for CNN+GRU, where performance improves when additional captions per image are incorporated during training. The

improvements are particularly visible when moving from $k = 1$ to $k = 3$, after which the metrics remain competitive with small variations across $k = 4$ and $k = 5$.

Table 3. Ablation study of single and multi-caption training in the CNN+LSTM model on the internal held-out test set (n = 1,560).

k	BLEU-1	BLEU-2	BLEU-3	BLEU-4	CIDEr	SPICE
1 caption	0.474	0.318	0.205	0.136	0.174	0.105
2 captions	0.541	0.369	0.243	0.166	0.190	0.112
3 captions	0.547	0.372	0.244	0.165	0.183	0.105
4 captions	0.531	0.362	0.239	0.162	0.187	0.111
5 captions	**0.556**	**0.383**	**0.256**	**0.175**	**0.206**	**0.120**

Figure 4 also summarizes the CNN+GRU results across the five multi-caption settings. In general, moving from $k = 1$ to multi-caption settings generally improved BLEU-1 to BLEU-4 scores, with the strongest BLEU-4 score observed at $k = 5$. CIDEr and SPICE also tended to improve when more captions were included, although the trend was not perfectly monotonic across all k values. CIDEr reached its highest value at $k = 1$, while SPICE improved substantially from $k = 1$ and remained relatively stable from $k = 2$ to $k = 5$. These observed trends suggest that multi-caption training can improve both lexical overlap and semantic alignment, although the gains vary across metrics.

Table 4 reports the CNN+GRU results across all multi-caption training scenarios. The highest CIDEr score for CNN+GRU is achieved at $k = 5$ with 0.226, and BLEU-4 reaches 0.192, which is very close to the value at $k = 3$ (0.191). SPICE improves markedly from 0.095 at $k = 1$ to 0.118 at $k = 2$, and peaks at 0.119 at $k = 5$. Overall, relative to the single-caption condition, the multi-caption setting shows observed improvements for content-oriented scoring (CIDEr and SPICE) and higher-order n-gram overlap (BLEU-4), supporting the effectiveness of leveraging multiple reference captions in training.

Table 4. Ablation study of single and multi-caption training in the CNN+GRU model on the internal held-out test set (n = 1,560).

k	BLEU-1	BLEU-2	BLEU-3	BLEU-4	CIDEr	SPICE
1 caption	0.469	0.314	0.202	0.134	0.175	0.095
2 captions	0.504	0.339	0.220	0.146	0.186	0.118
3 captions	0.571	0.392	0.268	0.191	0.217	0.118
4 captions	**0.574**	**0.396**	0.264	0.182	0.197	0.114
5 captions	0.564	0.390	**0.269**	**0.192**	**0.226**	**0.119**

3.3 Error Analysis

In addition to the automatic evaluation metrics, we conducted a qualitative error analysis to better understand how the generated captions changed across different multi-caption training scenarios. We selected representative examples from the internal held-out test set and compared the generated captions from $k = 1$ to $k = 5$. The analysis focuses on common captioning issues, including generic descriptions, repeated phrases, missing key information, object misidentification, and hallucinated content.

Table 5 shows three representative examples. In the first example, all generated captions correctly identify the image as a computer screen, but the lower k settings mostly produce generic and repetitive descriptions. As k increases, the captions become more aligned with the references, especially at $k = 5$, where the model begins to mention a message on the screen. However, the generated caption still did not fully capture the specific meaning of the repair prompt. This suggests that multi-caption supervision may help the model produce more informative descriptions, although fine-grained textual content remains difficult.

In the second example, the model gradually improved from repetitive and generic outputs at $k = 1$ and $k = 2$ toward a more accurate caption at $k = 4$, where it correctly describes a person holding a bottle of medicine. However, the $k = 5$ output introduces an incorrect phrase, "pickling hand sanitizer," showing that adding more captions does not always produce monotonic qualitative improvement. This is consistent with the quantitative results, where some metrics also fluctuate across different k values.

The third example illustrates a more severe error pattern. The ground-truth captions describe a book cover showing two dogs, while the generated captions incorrectly refer to a food box, barcode, or frozen food product. This indicates object-level mismatch and possible visual ambiguity. Even with higher k, the model produces more detailed captions, but the details remain semantically incorrect. Therefore, although multi-caption supervision can improve specificity in some cases, it does not completely eliminate object misidentification or hallucinated content, especially in challenging VizWiz images.

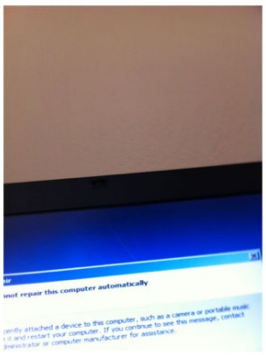
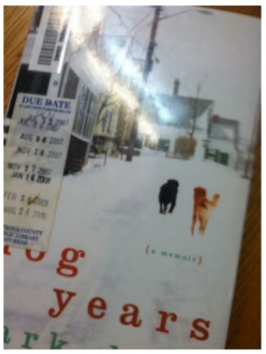
3.4 Best Configuration and Trend Analysis

This section summarizes the best-performing setting for each architecture and then analyzes the overall performance trend as the number of captions per image (k) increases.

Regarding trends over k, both models show clear gains when moving from single-caption training (k = 1) to multi-caption training. For CNN-LSTM, the jump from k = 1 to k = 2 improves all BLEU metrics (e.g., Bleu-4 rises from 0.136 to 0.166), and CIDEr also increases

from 0.174 to 0.190. After that, performance fluctuates slightly for k = 3 and k = 4, before reaching the best overall scores at k = 5 (Bleu-4 = 0.175, CIDEr = 0.206, SPICE = 0.120). This pattern indicates that the largest benefit comes early from adding a second caption per image, while additional captions provide incremental improvements that are more sensitive to variability in the training signal.

Table 5. Examples of qualitative assessment across different multi-caption training scenarios.

Example	Ground-truth Captions	Generated Captions
	1. A computer screen shows a repair prompt on the screen. 2. A computer screen with a repair automatically pop up. 3. Partial computer screen showing the need of repairs. 4. Part of a computer monitor showing a computer repair message. 5. The top of a laptop with a blue background and dark blue text.	(k=1): A computer screen with a windows screen and a computer screen. (k=2): A computer screen with a blue screen and a blue screen. (k=3): A computer screen with a blue screen and a blue background. (k=4): A computer screen with a blue background and a blue background. (k=5): A computer screen showing a windows screen with a message.
	1. A person is holding a bottle that has medicine for the night time. 2. A bottle of medication has a white twist top. 3. Night time medication bottle being held by someone. 4. A person holding a small black bottle of NIGHT TIME. 5. A bottle of what appears to be cough syrup held in hand.	(k=1): A person holding a bottle of a bottle of a water bottle. (k=2): A person holding a bottle of a hand holding a bottle of a hand. (k=3): A person holding a bottle of a hand sanitizer. (k=4): A person is holding a bottle of medicine in their hand. (k=5): A person holding a bottle of pickling hand sanitizer.
	1. a white paper showing an image of black and brown dog. 2. A library book with pictures of two dogs on the cover on a wooden table. 3. A book with a black and a tan dog walking down a snowy street. 4. The book cover shows two dogs in the snow. 5. A book cover title Dog Years with an image of a black and brown dog walking up the street, on the left side it has a due date sticker from a library.	(k=1): A book with a white and white label on it. (k=2): A box of food with a barcode on it. (k=3): A box of food with a picture of a food on it. (k=4): A box of a frozen food product with a picture of the contents on the front. (k=5): A box of a food item with a picture of a man on the front.

CNN-GRU shows a slightly different progression, with a notable improvement around the mid-k range. While the increase from k = 1 to k = 2 yields modest BLEU improvements and a clear SPICE gain (0.095 to 0.118), the strongest rise appears at k = 3, where Bleu-4 increases to 0.191 and CIDEr reaches 0.217. The scores dip somewhat at k = 4 (Bleu-4 = 0.182, CIDEr = 0.197), but recover again at k = 5, where CNN-GRU achieves its best CIDEr (0.226) and maintains high Bleu-4 (0.192) along with SPICE = 0.119. The non-monotonic behavior at k = 4 is plausible in VizWiz because captions may vary in quality and specificity; adding more captions increases diversity but also increases the chance of introducing noisy or weak descriptions, which can temporarily lower overlap-based metrics before the overall supervision becomes sufficiently rich.

Across both architectures, the best-performing configuration occurs at k = 5, which is consistent with the dataset design where each image is paired with five reference captions. Practically, this implies that using all available captions during training is a reasonable default for VizWiz, especially when the goal is to maximize CIDEr and higher-order BLEU. Based on these results, the CNN-GRU model at k = 5 is treated as the strongest configuration in terms of BLEU-4 and CIDEr, while CNN-LSTM remains competitive and achieves slightly higher semantic quality as reflected by SPICE.

4. Conclusions

This study suggests that multi-caption supervision can improve image captioning performance on VizWiz-

style assistive data. The results indicate that multiple human captions should not be treated merely as redundant annotations, but as complementary supervision that captures different perspectives, levels of detail, and possible interpretations of the same image. This is especially valuable in assistive settings, where images are often noisy, incomplete, or visually ambiguous.

Overall, the observed results show that using more available captions during training can provide practical benefits for classical CNN-RNN captioning baselines, especially when all five captions are used. However, these findings should be interpreted with consideration of the evaluation design, as each configuration was trained using a single run without multi-seed analysis or variance estimation. Therefore, the reported improvements represent observed performance trends rather than statistically confirmed gains across different random initializations. Future studies should include multi-seed experiments, confidence interval analysis, broader caption sampling strategies, stronger architectures, and human-centered evaluation to further validate the robustness and practical value of multi-caption supervision.

Declarations

AI Usage Statement

The authors used ChatGPT only for language refinement, readability enhancement, and manuscript formatting support. AI was not used to generate or interpret research findings, conduct analysis, or substitute for the authors' scientific judgment. All scientific content, analysis, interpretation, conclusions, and responsibility for the manuscript are solely the authors' responsibility.

Author Contributions

Fauzan Firdaus: conceptualization, methodology, software, data curation, formal analysis, investigation, visualization, writing – original draft, writing – review and editing, manuscript revision and project coordination; Angel Metanosa Afinda: validation and writing – review and editing; Edward Ferdian: investigation, formal analysis and validation; Muhammad Maulana Yusuf: software, data preprocessing, experimental execution, metric evaluation, and visualization; Ema Rachmawati: conceptualization, formal analysis, methodology and validation.

Funding Statement

This research was funded by Telkom University under grant number 074/LIT06/PPM-LIT/2025.

Conflict of Interest Statement

The authors declare no conflict of interest.

Ethical Clearance

This study used a publicly available dataset and did not involve direct human participants or animal subjects. Therefore, formal ethical clearance was not required. The authors complied with applicable ethical standards for secondary data analysis and academic use of publicly available data.

Data and Code Availability

The dataset used in this study is the publicly available VizWiz-Captions dataset. The experimental code is not publicly deposited, but is available from the corresponding author upon reasonable request.

References

[1] D. Gurari, Y. Zhao, M. Zhang, and N. Bhattacharya, "Captioning Images Taken by People Who Are Blind," Jul. 15, 2020, *arXiv*: arXiv:2002.08565. doi: https://doi.org/10.48550/arXiv.2002.08565.

[2] P. Dognin *et al.*, "Image Captioning as an Assistive Technology: Lessons Learned from VizWiz 2020 Challenge," *J Artif Int Res*, vol. 73, May 2022, doi: https://doi.org/10.1613/jair.1.13113.

[3] H. Sharma, "A Survey on Image Captioning datasets and Evaluation Metrics," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1116, no. 1, p. 012184, Apr. 2021, doi: https://doi.org/10.1088/1757-899X/1116/1/012184.

[4] Y. Mao *et al.*, "Improving Reference-Based Distinctive Image Captioning with Contrastive Rewards," *ACM Trans Multimed. Comput Commun Appl*, vol. 20, no. 12, p. 390:1-390:24, Nov. 2024, doi: https://doi.org/10.1145/3694683.

[5] A. Radford *et al.*, "Learning Transferable Visual Models From Natural Language Supervision," Feb. 26, 2021, *arXiv*: arXiv:2103.00020. doi: https://doi.org/10.48550/arXiv.2103.00020.

[6] Z. Sun *et al.*, "Alpha-CLIP: A CLIP Model Focusing on Wherever You Want," Dec. 13, 2023, *arXiv*: arXiv:2312.03818. doi: https://doi.org/10.48550/arXiv.2312.03818.

[7] J. Li, D. Li, S. Savarese, and S. Hoi, "BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models," Jun. 15, 2023, *arXiv*: arXiv:2301.12597. doi: https://doi.org/10.48550/arXiv.2301.12597.

[8] J.-B. Alayrac *et al.*, "Flamingo: a Visual Language Model for Few-Shot Learning" Nov. 15, 2022, *arXiv*: arXiv:2204.14198. doi: https://doi.org/10.48550/arXiv.2204.14198.

[9] W. Dai *et al.*, "InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning," Jun. 15, 2023, *arXiv*: arXiv:2305.06500. doi: https://doi.org/10.48550/arXiv.2305.06500.

[10] H. Liu, C. Li, Y. Li, and Y. J. Lee, "Improved Baselines with Visual Instruction Tuning," May 15, 2024, *arXiv*: arXiv:2310.03744. doi: https://doi.org/10.48550/arXiv.2310.03744.

[11] Q. Ye, X. Zeng, F. Li, C. Li, and H. Fan, "Painting with Words: Elevating Detailed Image Captioning with Benchmark and Alignment Learning," presented at the The Thirteenth International Conference on Learning Representations, Oct. 2024. Accessed: Jan. 05, 2026. [Online]. Available: https://openreview.net/forum?id=636M0nNbPs

[12] L. Yu, M. Nikandrou, J. Jin, and V. Rieser, "Quality-agnostic Image Captioning to Safely Assist People with Vision Impairment," in *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, Macau, SAR China: International Joint Conferences on Artificial Intelligence Organization, Aug. 2023, pp. 6281–6289. doi: https://doi.org/10.24963/ijcai.2023/697.

[13] A. Thobhani *et al.*, "A Survey on Enhancing Image Captioning with Advanced Strategies and Techniques," *Comput. Model. Eng. Sci.*, vol. 142, no. 3, pp. 2247–2280, Mar. 2025, doi: https://doi.org/10.32604/cmes.2025.059192.

[14] Kavitha Soppari, Pakide Kavya, Kotla Pranay Teja, and Bethi Pavan Sai, "A survey on image captioning methods," *World J.*

Adv. Res. Rev., vol. 26, no. 2, pp. 3134–3143, May 2025, doi: https://doi.org/10.30574/wjarr.2025.26.2.1705.

[15] A. Abdussalam, Z. Ye, A. Hawbani, M. Al-Qatf, and R. Khan, "NumCap: A Number-controlled Multi-caption Image Captioning Network," *ACM Trans Multimed. Comput Commun Appl*, vol. 19, no. 4, p. 148:1-148:24, Feb. 2023, doi: https://doi.org/10.1145/3576927.

[16] M. U. Hadi *et al.*, "A Survey on Large Language Models: Applications, Challenges, Limitations, and Practical Usage", Accessed: Jan. 14, 2026, doi: https://doi.org/10.36227/techrxiv.23589741.v1.

[17] A. Joshi, K. Kalal, D. Bhandare, V. Patil, U. Kulkarni, and S. M. Meena, "Image Captioning Using CNN-LSTM," in *Advances in Communication and Applications*, N. R. Shetty, N. H. Prasad, and H. C. Nagaraj, Eds., Singapore: Springer Nature, 2024, pp. 421–433. doi: https://doi.org/10.1007/978-981-99-7633-1_32.

[18] A. A. E. Osman, M. A. W. Shalaby, M. M. Soliman, and K. M. Elsayed, "A Survey on Attention-Based Models for Image Captioning," *Int. J. Adv. Comput. Sci. Appl. IJACSA*, vol. 14, no. 2, Feb. 2023, doi: https://doi.org/10.14569/IJACSA.2023.0140249.

[19] L. Celona, S. Bianco, M. Donzella, and P. Napoletano, "Improving image captioning descriptiveness by ranking and LLM-based fusion," *Neural Comput. Appl.*, vol. 37, no. 32, pp. 27279–27299, Nov. 2025, doi: https://doi.org/10.1007/s00521-025-11672-x.

[20] V. Zouhar and O. Bojar, "Quality and Quantity of Machine Translation References for Automatic Metrics," Apr. 10, 2024, *arXiv*: arXiv:2401.01283. doi: https://doi.org/10.48550/arXiv.2401.01283.

[21] S. Casola, Y. J. Liu, S. Peng, O. Kraus, A. Gatt, and B. Plank, "References Matter: Investigating the Impact of Reference Set Variation on Summarization Evaluation," in *Proceedings of the 18th International Natural Language Generation Conference*, L. Flek, S. Narayan, L. H. Phương, and J. Pei, Eds., Hanoi, Vietnam: Association for Computational Linguistics, Oct. 2025, pp. 274–291. Accessed: Jan. 14, 2026. [Online]. Available: https://aclanthology.org/2025.inlg-main.18/

[22] T. Tang *et al.*, "Not All Metrics Are Guilty: Improving NLG Evaluation by Diversifying References," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, K. Duh, H. Gomez, and S. Bethard, Eds., Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 6596–6610. doi: https://doi.org/10.18653/v1/2024.naacl-long.367.

[23] A. Thobhani *et al.*, "A novel image captioning model with visual-semantic similarities and visual representations re-weighting," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 36, no. 7, p. 102127, Sep. 2024, doi: https://doi.org/10.1016/j.jksuci.2024.102127.

[24] Y. H. Zaidan and J. Waleed, "Image Captioning Generation Using Inception V3 and Attention Mechanism," *J. Al-Qadisiyah Comput. Sci. Math.*, vol. 15, no. 2, p. Page 1-10, Sep. 2023, doi: https://doi.org/10.29304/jqcm.2023.15.2.1228.

[25] A. K. Poddar and Dr. R. Rani, "Hybrid Architecture using CNN and LSTM for Image Captioning in Hindi Language," *Procedia Comput. Sci.*, vol. 218, pp. 686–696, Jan. 2023, doi: https://doi.org/10.1016/j.procs.2023.01.049.

[26] Y. Geng, H. Mei, X. Xue, and X. Zhang, "Image-Caption Model Based on Fusion Feature," *Appl. Sci.*, vol. 12, no. 19, p. 9861, Jan. 2022, doi: https://doi.org/10.3390/app12199861.

[27] N. M. Khassaf and N. H. M. Ali, "Improving Pre-trained CNN-LSTM Models for Image Captioning with Hyper-Parameter Optimization," *Eng. Technol. Appl. Sci. Res.*, vol. 14, no. 5, pp. 17337–17343, Oct. 2024, doi: https://doi.org/10.48084/etasr.8455.

[28] K. R. Suresh, A. Jarapala, and P. V. Sudeep, "Image Captioning Encoder–Decoder Models Using CNN-RNN Architectures: A Comparative Study," *Circuits Syst. Signal Process.*, vol. 41, no. 10, pp. 5719–5742, Oct. 2022, doi: https://doi.org/10.1007/s00034-022-02050-2.

[29] R. Khan, M. S. Islam, K. Kanwal, M. Iqbal, M. I. Hossain, and Z. Ye, "A Deep Neural Framework for Image Caption Generation Using GRU-Based Attention Mechanism," Mar. 03, 2022, *arXiv*: arXiv:2203.01594. doi: https://doi.org/10.48550/arXiv.2203.01594.

[30] U. Berger, G. Stanovsky, O. Abend, and L. Frermann, "Surveying the Landscape of Image Captioning Evaluation: A Comprehensive Taxonomy, Trends and Metrics Analysis," Sep. 14, 2025, *arXiv*: arXiv:2408.04909. doi: https://doi.org/10.48550/arXiv.2408.04909.

[31] O. González-Chávez, G. Ruiz, D. Moctezuma, and T. Ramirez-delReal, "Are metrics measuring what they should? An evaluation of Image Captioning task metrics," *Signal Process. Image Commun.*, vol. 120, p. 117071, Jan. 2024, doi: https://doi.org/10.1016/j.image.2023.117071.

[32] A. de S. Inácio and H. S. Lopes, "Evaluation metrics for video captioning: A survey," *Mach. Learn. Appl.*, vol. 13, p. 100488, Sep. 2023, doi: https://doi.org/10.1016/j.mlwa.2023.100488.