mathematics

MDPI

Article

Modeling of Hyperparameter Tuned Deep Learning Model for Automated Image Captioning

Mohamed Omri [1], Sayed Abdel-Khalek [2,3], Eied M. Khalil [4,5], Jamel Bouslimi [6] and Gyanendra Prasad Joshi [7,*]

1 Deanship of Scientific Research, King Abdulaziz University, Jeddah 21589, Saudi Arabia; Omrimoha2002@yahoo.fr
2 Mathematics Department, Faculty of Science, Taif University, Taif 21944, Saudi Arabia; sabotalb@tu.edu.sa
3 Mathematics Department, Faculty of Science, Sohag University, Sohag 82524, Egypt
4 Department of Mathematics, Faculty of Science, Taif University, Taif 21944, Saudi Arabia; eiedkhalil@tu.edu.sa
5 Mathematics Department, Faculty of Science, Azhar University, Cairo 11884, Egypt
6 Physics Department, Faculty of Science, Taif University, Taif 21944, Saudi Arabia; jamelabouaysem@yahoo.fr
7 Department of Computer Science and Engineering, Sejong University, Seoul 05006, Korea
* Correspondence: joshi@sejong.ac.kr; Tel.: +82-2-6935-2481

Abstract: Image processing remains a hot research topic among research communities due to its applicability in several areas. An important application of image processing is the automatic image captioning technique, which intends to generate a proper description of an image in a natural language automated. Image captioning is a recently developed hot research topic, and it started to receive significant attention in the field of computer vision and natural language processing (NLP). Since image captioning is considered a challenging task, the recently developed deep learning (DL) models have attained significant performance with increased complexity and computational cost. Keeping these issues in mind, in this paper, a novel hyperparameter tuned DL for automated image captioning (HPTDL-AIC) technique is proposed. The HPTDL-AIC technique encompasses two major parts, namely encoder and decoder. The encoder part utilizes Faster SqueezNet with the RMSProp model to generate an effective depiction of the input image via insertion into a predefined length vector. At the same time, the decoder unit employs a bird swarm algorithm (BSA) with long short-term memory (LSTM) model to concentrate on the generation of description sentences. The design of RMSProp and BSA for the hyperparameter tuning process of the Faster SqueezeNet and LSTM models for image captioning shows the novelty of the work, which helps to accomplish enhanced image captioning performance. The experimental validation of the HPTDL-AIC technique is carried out against two benchmark datasets, and the extensive comparative study pointed out the improved performance of the HPTDL-AIC technique over recent approaches.

Keywords: image captioning; deep learning; machine learning; encoder; decoder; hyperparameter tuning

Citation: Omri, M.; Abdel-Khalek, S.; Khalil, E.M.; Bouslimi, J.; Joshi, G.P. Modeling of Hyperparameter Tuned Deep Learning Model for Automated Image Captioning. *Mathematics* **2022**, *10*, 288. https://doi.org/10.3390/math10030288

Academic Editors: Javier Martínez, Bo-Hao Chen and Francisco Chiclana

Received: 19 November 2021
Accepted: 11 January 2022
Published: 18 January 2022

1. Introduction

Over the last years, the image processing and computer vision (CV) system has made significant progress in various fields such as object detection and image classification. Benefitting from the advancement of object detection and image classification, it becomes possible to generate more than one sentence automatically for understanding the visual content of an image, called image captioning. Automatically creating natural images and complete description has greater impacts, namely titles description related to healthcare images, attached to news images, accessing data for blind users, human–robot communication, and text-based image retrieval [1]. This application in image captioning has significant practical and theoretical research values. Hence, image captioning is a sophisticated but useful task in the era of artificial intelligence (AI) technology.

Provided a novel image, an image captioning method needs to output descriptions based on images at a semantic level. For instance, the input images consist of waves, people,

and boards. At the bottom, there is a sentence that describes the image content—the object appearing in the action, the scene, and the image is defined in this sentence [2]. For image captioning tasks, one could understand image contents easily and express it in the form of natural language sentences based on certain requirements; however, for computers, it needs the combined use of natural language processing (NLP), image processing, CV, etc. The main challenges of the image captioning method are to develop a method that could fully utilize image data to create a more human-like rich image description [3]. The significant descriptions of higher-level image semantics require scene recognition in the image or understanding of objects; understanding the relationships amongst them; generating syntactically and semantically correct sentences; and the ability to analyze their states.

Several studies have been dedicated to automated image captioning, and also it is classified in different ways [4]. The retrieval-based approach detects visually similar images using the caption from the trained dataset, and image captioning is chosen from similar images using the caption. Template-based image captioning finds the actions, objects, or attributes and fills the blank slots in a fixed template [5]. This method is capable of generating proper captions, but it is not able to create semantically correct and image-specific captions. In contrast to this, the novel image caption generation approach produces image caption from the visual contents using language models and then examines the visual content of an image [6]. In comparison with the above two classes, novel caption generation could create proper captions for the provided image, which is semantically correct when compared to earlier methods [7]. Various studies are based on deep learning (DL) and machine learning (ML) techniques. A deep neural network (DNN) system has been utilized for the image captioning method due to effective approximation abilities. The image captioning technique has remarkably advanced due to the tremendous growth of the DNN system [8]. Convolutional Neural Network (CNN) has attained remarkable effect in CV tasks, such as object detection, and image classification in recent years. Moreover, Recurrent Neural Network (RNN) system performed an important role in NLP. A massive amount of study-based DL approaches was published in the past few years. Although several studies have been available in the literature, there is still a need to design effective image captioning techniques for improved performance.

This paper develops a novel hyperparameter tuned DL for automated image captioning (HPTDL-AIC) technique. The HPTDL-AIC technique encompasses two major parts, namely encoder and decoder. The encoder part utilizes Faster SqueezNet with the RMSProp model to generate an effective depiction of the input image via inserting it into a predefined length vector. At the same time, the decoding unit employs a bird swarm algorithm (BSA) with long short-term memory (LSTM) model to concentrate on the generation of the description sentences. The designs of RMSProp and BSA for the hyperparameter tuning process of the Faster SqueezeNet and LSTM models show the novelty of the work. In order to examine the HPTDL-AIC technique's enhanced outcomes, a series of simulations were performed on two benchmark datasets. In short, the contributions of these studies are summarized as follows:

- Develops a novel HPTDL-AIC technique for the automated image captioning process;
- Aims to create correct descriptions for the input images by the use of encoder–decoder structure;
- Employs the Faster SqueezeNet with RMSProp model for the extraction of visual features that exist in the image;
- Presents a BSA with LSTM as a language modeling tool to generate description sentences and decodes the vector into sentences;
- Validate the performance of the HPTDL-AIC technique using two benchmark datasets and inspect the results under several aspects.

The rest of the study is organized as follows. Section 2 offers a detailed description of the HPTDL-AIC technique, and its experimental analysis take place in Section 3. Lastly, Section 4 draws the major findings of the study with future scope.

2. Literature Review

Ren et al. [9] presented an advanced decision-making architecture for captioning images. They employ value and policy networks to collectively produce captions. The value network acts as a lookahead and global guidance by estimating each possible extension of the present state. In addition, the policy network acts as local guidance by offering the confidence of forecasting the following word as per the present state. It alters the aim of forecasting the accurate word towards the aim of creating captions that are the same as the ground truth caption. Kesavan et al. [10] systematically analyzed distinct deep DNN-based pre-trained models and image caption generation methods to accomplish the effective models by finetuning. The examined model contains with and without 'attention' concepts for optimizing the caption generation capacity. Each model is trained on a similar dataset for actual comparison.

Wang et al. [11] presented a multi-layer dense attention approach for image captioning process. The authors utilized a faster recurrent convolutional neural network (Faster R-CNN) for the extraction of image features as the coding layer, the long short-term memory (LSTM) attend can be utilized for decoding the multi-layer dense attention approach, and the description text is created. The hyperparameters of the model are tuned by the use of strategy gradient optimization in reinforcement learning. The utilization of the dense attention scheme at the coding layer eliminates the interference of non-salient information and selectively outputs the respective description text for the decoding procedure.

Sharma [12] proposed a novel image captioning method that considers the text present in the image. The study employs the idea of the morphology of the word and, therefore, creates Fisher Vectors-based morphology of a word. The presented method is estimated on two open-source datasets. The caption generated by the presented approach is similar to advanced art captioning models. Cheng et al. [13] proposed a semi-supervised DL approach, named the N-gram + Pseudo Label NIC technique. The approach integrates present DNN systems, for example, pseudo labels, N-gram, and NIC (Neural Image Caption) methods. This technique produces pseudo labels by the N-gram search method and enhances the effects of the method by employing people's descriptive habits and previous knowledge of the N-gram tables.

Zeng et al. [14] developed a technique of ultrasound image captioning-based region detection. Simultaneously, this technique encodes and detects the focus region in ultrasound images, uses the LSTM to decode the vector, and produces annotation text data to describe the disease contents in an ultrasound image. Shen et al. [15] designed a Variational Autoencoder and Reinforcement Learning-based Two-stage Multi-task Learning Model (VRTMM) for the remote sensing image captioning process. Initially, finetune the VAE and CNN models. Next, the transformer generates text descriptions with semantic and spatial features. Then, the Reinforcement Learning (RL) approach is used for enhancing the quality of the sentence. Although several models are available in recent times, the proposed model focuses on the design of encoding and decoding units to generate an effective depiction of the input image via insertion into a predefined length vector and concentrating on the generation of description sentences.

3. The Proposed Image Captioning Model

For an effective and automated image captioning process, a novel HPTDL-AIC technique has been developed, which aims to produce appropriate descriptions for input images by the use of an encoder–decoder structure. In particular, the encoder unit includes the Faster SqueezeNet with RMSProp model for generating a one-dimensional vector representation of the input image. Then, the BSA with the LSTM model is utilized as a decoder to produce description sentences and decode the vector into a sentence. In addition, RMSProp and BSA techniques are applied to appropriately tune the hyperparameter involved in it. Figure 1 showcases the overall working process of the HPTDL-AIC technique. The steps involved in the proposed model are listed as follows.

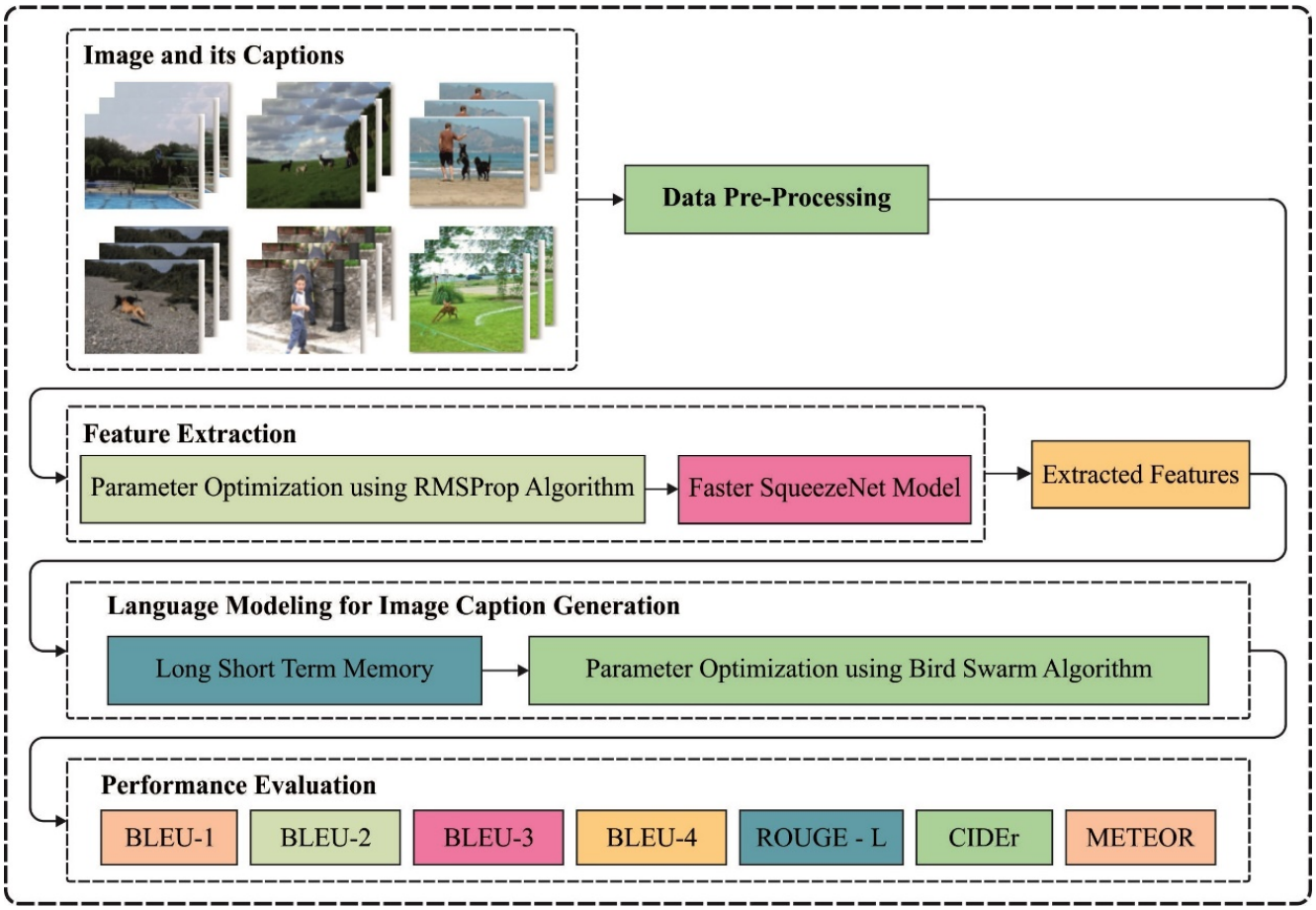

Figure 1. The overall process of the HPTDL-AIC method.

Step 1: Preprocessing. At the primary stage, actual input data can be transformed into a useful format by the inclusion of several subprocesses such as lower case conversion, punctuation mark removal, tokenization, and vectorization.

Step 2: Feature Extraction. Next to data preprocessing, the feature extraction process is performed by using Faster RCNN with RMSProp model, which is commonly utilized to generate visual features.

Step 3: Image Caption Generation. Finally, the textual description of the images is automatically generated by the use of the LSTM model, and the hyperparameter tuning of the LSTM model is appropriately adjusted by the use of BSA.

3.1. Pre-Processing

Initially, data preprocessing occurs from varying levels as listed here:

- The dataset text has words with distinct letter cases, which creates issues to components the same as the words with varying capitalized are regarded as altered. Thus, this improves issue vocabulary and afterward results in complexity. Therefore, it can be essential to alter the entire text to lower case in order to prevent this problem.
- The presence of punctuation improves the complexity of these issues; therefore, they are removed from the dataset.
- Numerical data present from the text retain an issue in the component as it increases the vocabulary that is extracted.
- Indicates initial and final order: word tokens '<start>' and '<end>' are further initial and final of every sentence for representing the initial and last token of the forecast order to the component.

- Tokenization: clean text is separated into constituent words, and a dictionary including the entire vocabulary to word-to-index and index-to-word equivalent are obtained.
- Vectorization: For resolving different sentence lengths, the short sentence is padded to the length of long sentence orders.

3.2. Feature Extraction: Optimal Faster SqueezeNet Model

At this stage, the Faster SqueezeNet model is utilized to generate visual features of the applied images. For improving the performance of the electronic module classifier, the Faster SqueezeNet was presented. For preventing overfit, BatchNorm and remaining frameworks can be used. Simultaneously, as DenseNet, it utilizes concatenation for connecting distinct layers for enhancing the expressiveness of the initial layers from the network. The Faster SqueezeNet has one BatchNorm layer, three-block layers, four convolutional, and global average pooling layers. Faster SqueezeNet is mostly enhanced in subsequent manners: In order to enhance the data flow amongst layers, it can imitate the DenseNet framework and present various connection modes. It contains a pooling layer and fire module and, eventually, the two concat layers are linked to the next convolutional layer. The existing layer obtains every feature map of the earlier layer, and it can be utilized $x_0, \ldots, x_{l-1}$ as input; afterwards, x_l is demonstrated as follows:

$$x_l = H_l\left([x_0,\ x_1,\ \ldots,\ x_{l-1}]\right), \tag{1}$$

where $[x_0,\ x_1,\ \ldots,\ x_{l-1}]$ signifies the association of feature graph created from the layer $0, 1, \ldots, l-1$, and H_l concatenates multiple inputs [16]. Without extremely enhancing the number of network variables, the efficiency of the network has improved from initial phases; simultaneously, some two-layer network is directly connected data. For ensuring optimum network convergence, it can be learned in the ResNet framework and presents various structure blocks that have pooling layers and fire modules. At last, when the two layers are summed, they can be linked to the next convolution layer. Figure 2 illustrates the framework of the SqueezeNet model.

In the ResNet model, the shortcut links utilize identity mapping indicating that the input of the convolutional stack was provided straight to the resultant convolutional stack. Properly speaking, for representing the desired fundamental mapping as $H(x)$, assume the stacked non-linear layer appropriate for another mapping of $F(x) := H(x) - x$. A new mapping is a reform to $(x) + x$. $F(x) + x$, which is realized by a framework named as shortcut linking from the actual encoded method. The shortcut connections generally skip more than one layer. Thus, it can utilize the remaining framework of ResNet to address the gradient vanishing problem without enhancing the number of network parameters.

In order to properly adjust the hyperparameter of the Faster SqueezeNet model, RMSProb is utilized. RMSProp (root mean square propagation) is an optimization method developed by Geoffrey E. Hinton in Coursera. For additionally optimizing the loss functions in the upgrade of extreme swings and accelerating the convergence function, the RMSProp method utilized the differential square weight average for the gradient of weight Wand bias b. Consequently, it makes great advancement in the direction wherever the variable space is gentler. The amount of squares of the historical gradient is small due to the gentler direction that results in a small learning drops. Assume t iteration process, which is described as follows:

$$s_{dw} = \beta s_{dw} + (1-\beta) dW^2, \tag{2}$$

$$s_{db} = \beta s_{db} + (1-\beta) db^2, \tag{3}$$

$$W = W - \alpha \frac{dW}{\sqrt{s_{dw}} + \varepsilon}, \tag{4}$$

$$b = b - \alpha \frac{db}{\sqrt{s_{db}} + \varepsilon}, \tag{5}$$

where as s_{dw} and s_{db} represent gradient and gradient momentum accumulated using the loss function in the preceding iteration $t-1$, as well as β vector, which is an exponential of gradient accumulation. For avoiding the situation where the denominator becomes zero, ε becomes a smaller value. RMSProp assists in removing the direction of the larger swing and is employed for correcting the swing in order render the swing in all the dimensions small. Alternatively, it makes the network function converge fast.

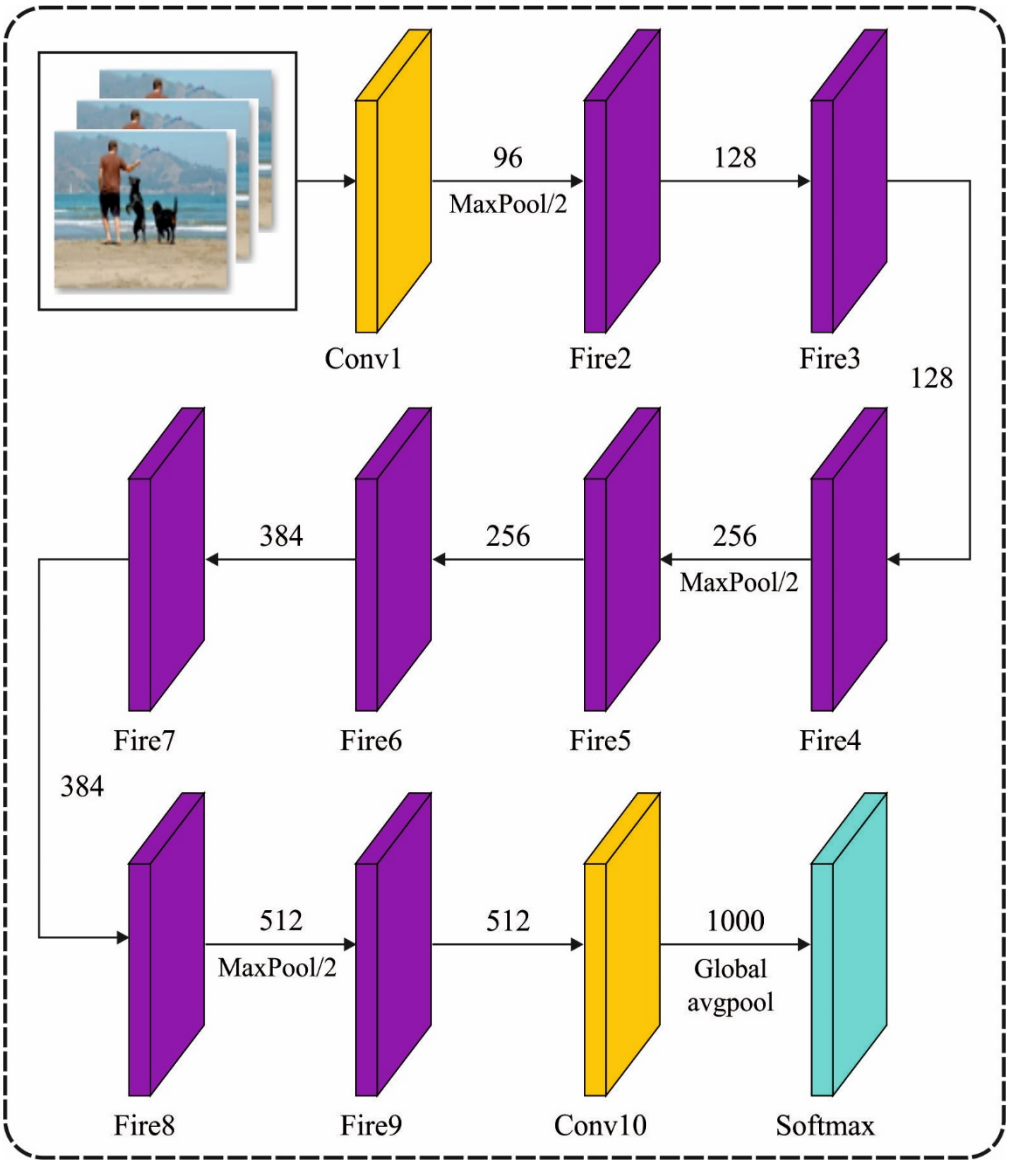

Figure 2. Structure of SqueezeNet model.

3.3. Language Modeling for Image Caption Generation

Finally, the LSTM model is applied to produce effective description sentences of the applied input images. The LSTM network is effectively utilized for accomplishing the tasks of machine translation and order generation. During the structure, LSTM can be applied as a language method for generating suitable captions dependent upon the input vector in ResNet50 output:

$$f_t = \sigma\left(W_f \cdot [h_{t-1},\ x_t] + b_f\right), \tag{6}$$

where the resultant vector of the preceding cell h_{t-1} with a novel element of order x_t has concatenated [17].

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t. \tag{7}$$

In two generated vectors, the states in C_{t-1} to C_t were utilized for updating. Thus, it multiplies the past state by f_t for forgetting data detection, as it is unnecessary from the preceding step; afterwards, add $i_t * \widetilde{C}_t$.

$$\begin{aligned} i_t &= \sigma\left(W_j \cdot [h_{t-1},\ x_t] + b_j\right), \\ \widetilde{C}_t &= \tanh(W_c \cdot [h_{t-1},\ x_t] + b_c). \end{aligned} \tag{8}$$

The input gate defines what value is upgraded, and the tanh layer generates the vector of novel candidates to $\widetilde{C}_t$, and the values are more towards a cell state.

$$h_t = o_t * \tanh(C_t). \tag{9}$$

The attained values of C_t and h_t are transferred to the NN input at time $t+1$.

$$\begin{aligned} o_t &= \sigma(W_o \cdot [h_{t-1},\ x_t] + b_o), \\ p_t &= softmax\ (h_t). \end{aligned} \tag{10}$$

The multiplicative filter permits efficient training of LSTM because it is optimum for preventing explosions as well as the vanishing gradient. Non-linearity has been offered as the sigmoid $\sigma(\cdot)$ and the hyperbolic tangent $h()$. During the final formula, h_t refers the fed to softmax function for calculating the probability distributions p_t on every word. This function has been computed and optimized on the entire trained dataset. The word with maximal likelihood was chosen at every time step and passed to succeeding ones for generating complete sentences.

For proper hyperparameter tuning of the LSTM model, BSA is applied in such a manner that the overall performance becomes improved. BSA is a nature inspired technique, which is stimulated by social behavior and social interaction in the bird's swarm. It stimulates the nature of the birds' foraging, vigilance, and flight behavior. Therefore, swarm behavior can be effectively derived from the swarm of birds for optimization processes. The birds' swarm technique can be simplified by five rules:

- Rule1: All birds are switched amongst vigilant as well as foraging behaviors. If a bird forages or retains vigilance, it can be defined as a stochastic decision.
- Rule2: If foraging, all birds record and upgrade their preceding optimum experience and swarm earlier optimum experience. The experience is utilized for searching for food. Social information is distributed concurrently amongst the entire swarm.
- Rule3: While maintaining vigilance, all birds attempt to move nearby the center of swarm. This characteristic can be determined by disturbance due to swarm competitions. the birds with higher reserves further tend towards adjacent swarm centers than birds with lower reserves.
- Rule4: The bird flies to other locations frequently. Upon flying to other places, birds frequently switch amongst production as well as scrounging. The bird with maximum reserves becomes a producer, and others with minimum reserves are scroungers. Another bird with maximal as well as minimal reserves was arbitrarily chosen to be the producer as well as a scrounger.
- Rule5: The producer actively seeks food. The scroungers arbitrarily follow a producer for searching the food.

Based on Rule1, the time interval of all birds' flight performance FQ, the probability of foraging performance $P(P \in (0,1))$, and uniform, arbitrary value $\delta \in (0,1)$ can be

determined. When the amount of iteration is lesser than FQ and $\delta \leq P$, the bird has foraging performance. Rule2 has been expressed mathematically as follows [18]:

$$x_{i,j}^{t+1} = x_{i,j}^{t} + \left(p_{i,j}^{t} - x_{i,j}^{t}\right) \times C \times rand(0,1) + \left(g_{j}^{t} - x_{i,j}^{t}\right) \times S \times rand(0,1), \quad (11)$$

where C and S refer to two positive numbers; the previous number is known as a cognitive accelerated coefficient, and the final number is termed as a social accelerated coefficient. At this point, $p_{i,j}$ implies the ith bird optimum preceding place, and g_j stands for the optimum preceding swarm place. When the amount of iteration is lesser than FQ and $\delta > P$, the bird has vigilance performance. Rule3 is expressed as mathematical procedure y as follows:

$$x_{i,j}^{t+1} = x_{i,j}^{t} + A_1\left(mean_{j}^{t} - x_{i,j}^{t}\right) \times rand(0,1) + A_2\left(p_{k,j}^{t} - x_{i,j}^{t}\right) \times rand(-1,1), \quad (12)$$

$$A_1 = a_1 \times \exp\left(-\frac{pFit_i}{sumFit + \varepsilon} \times N\right), \quad (13)$$

$$A_2 = a_2 \times \exp\left(\left(\frac{pFit_i - pFit_k}{|pFit_k - pFit_i| + \varepsilon}\right) \times \frac{N \times pFit_k}{sumFit + \varepsilon}\right), \quad (14)$$

where a_1 and a_2 refer to the two positive constants from 0 and 2, $pFit_i$ signifies the optimum fitness value of ith bird, and $sumFit$ indicates the sum of swarms' optimum fitness value.

When the amount of iteration is equivalent to FQ, the bird has flight performance that is separated as to the performances of the producer as well as scrounger by fitness. Rule3 and Rule4 have been formulated as a mathematical model as follows:

$$x_{i,j}^{t+1} = x_{i,j}^{t} + randn(0,1) \times x_{i,j}^{t}, \quad (15)$$

$$x_{i,j}^{t+1} = x_{i,j}^{t} + \left(x_{k,j}^{t} - x_{i,j}^{t}\right) \times FL \times rand\ (0,1), \quad (16)$$

where FL ($FL \in [0,2]$) denotes the scrounger who will follow the producer in searching for food. The BSA approach develops an FF for attaining enhanced classifier efficiency. It defines a positive integer for representing the optimum efficiency of candidate solution. During this analysis, the minimized classification error rate has been assumed as FF is provided in Equation (17). An optimum solution is a lesser error rate, and the lowest solution gains an enhanced error rate.

$$fitness(x_i) = ClassifierErrorRate(x_i) = \frac{number\ of\ misclassified\ documents}{Total\ number\ of\ documents} * 100 \quad (17)$$

4. Performance Validation

In this section, the performance validation of the HPTDL-AIC technique takes place by using two benchmark datasets, namely Flickr8K [19] and MSCOCO [20] caption dataset. The results are examined under various measures, namely BLEU, Meter, CIDEr, and Rouge-L.

4.1. Implementation Data

The Flickr dataset generally comprises Flickr8k and 30K datasets. The Flickr8K dataset includes 8000 images, which exhibit human activities. The images in the dataset contain five sentences of textual descriptions. The MSCOCO dataset has collected data with many objects with scenarios. Figure 3 showcases the sample set of test images that exist in the dataset. The dataset file contains image captions for every sample image.

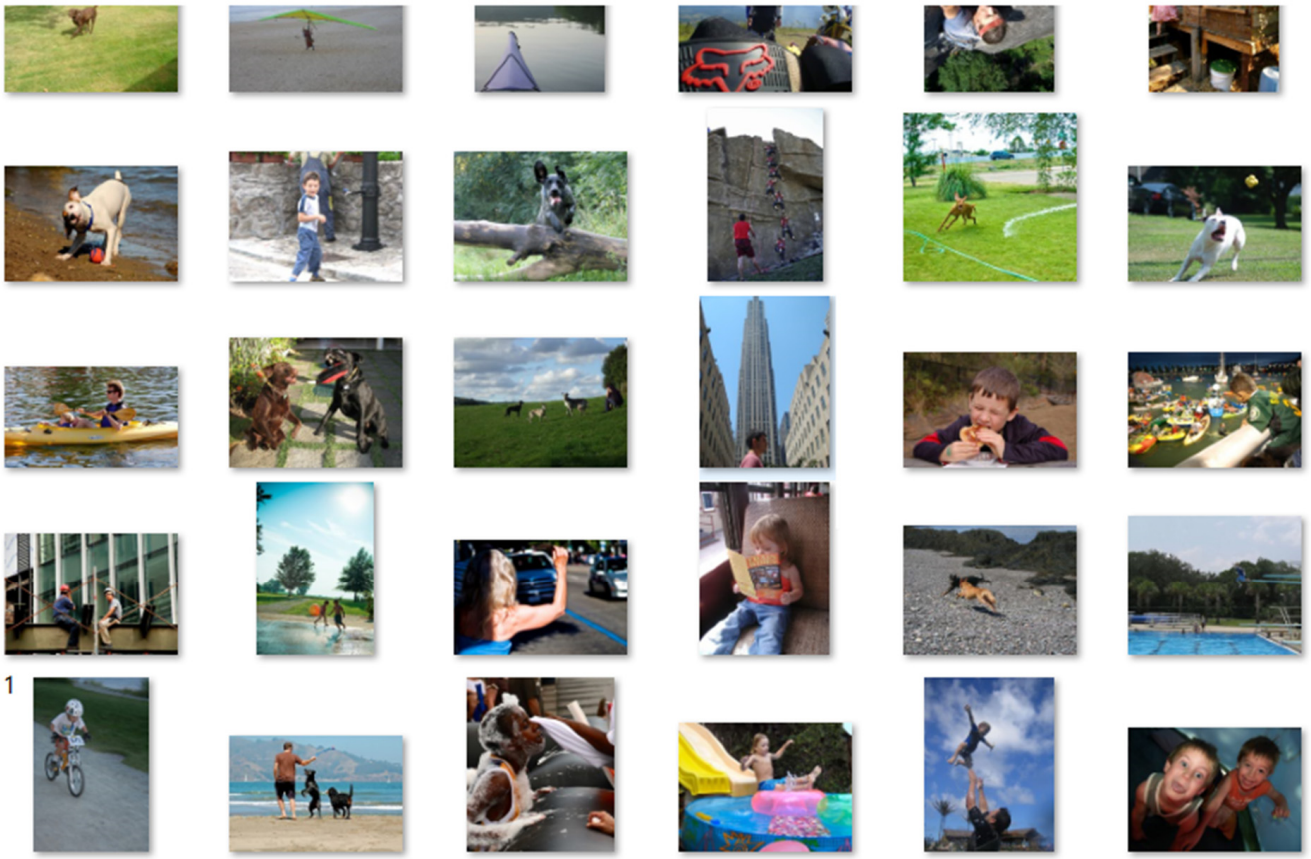

Figure 3. Sample Images.

4.2. Performance Measures

In this study, four measures are used for experimental validation, namely BLEU, Meter, CIDEr, and Rouge-L. Bleu [21] is a widely employed measure to estimate the quality of the generated text. The value of Blue needs to be high for better machine translation performance. It can be determined as follows:

$$\begin{aligned} BP &= \min\left(1, e^{1-\frac{r}{c}}\right) \\ BLEU_N &= BP \cdot e^{\frac{1}{N}\sum_{n=1}^{N} logp_n} \end{aligned} \tag{18}$$

where BP indicates the penalty factor, and r and c denote the length of the reference and generated sentences, respectively. The METEOR [22] measure mainly depends upon word recall rate and single precision weighted harmonic mean. It determines the reconciliation mean of accuracy and recalls among the optimal candidate and reference translations. It can be computed by using Equation (19):

$$METEOR = (1 - Pen)F_{mean} \tag{19}$$

where, α, γ, and θ denote the default variables.

The CIDEr [23] index considers every sentence as a "document" and reports it in the form of a TF-IDF vector. It determines cosine similarity among the generated caption (s_{ij}) and reference caption using a score value as defined in the following.

$$CIDEr_n(c_i, S_i) = \frac{1}{m}\sum_j \frac{g^n(c_i)^T g^n(s_{ij})}{\|g^n(c_i)\| \cdot \|g^n(s_{ij})\|} \tag{20}$$

ROUGE [24] (Recall-Oriented Understudy for Gisting Evaluation) is a similarity measurement approach that depends upon the recall rate. It determines the co-occurrence probability of N-gram in the reference translation and the translation to be examined. It can be mathematically formulated as follows.

$$ROUGE - N = \frac{\sum_{S \in \{ReferencesSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferencesSummaries\}} \sum_{gram_n \in S} Count(gram_n)} \quad (21)$$

4.3. Visualization Results

Figure 4 visualizes sample image captioning results obtained by the HPTDL-AIC technique. Figure 4a shows the sample test image, and the respective image caption generated image is provided in Figure 4b. The figure implied that the HPTDL-AIC technique has properly provided the textual description of the image as "man with dog in the mountain".

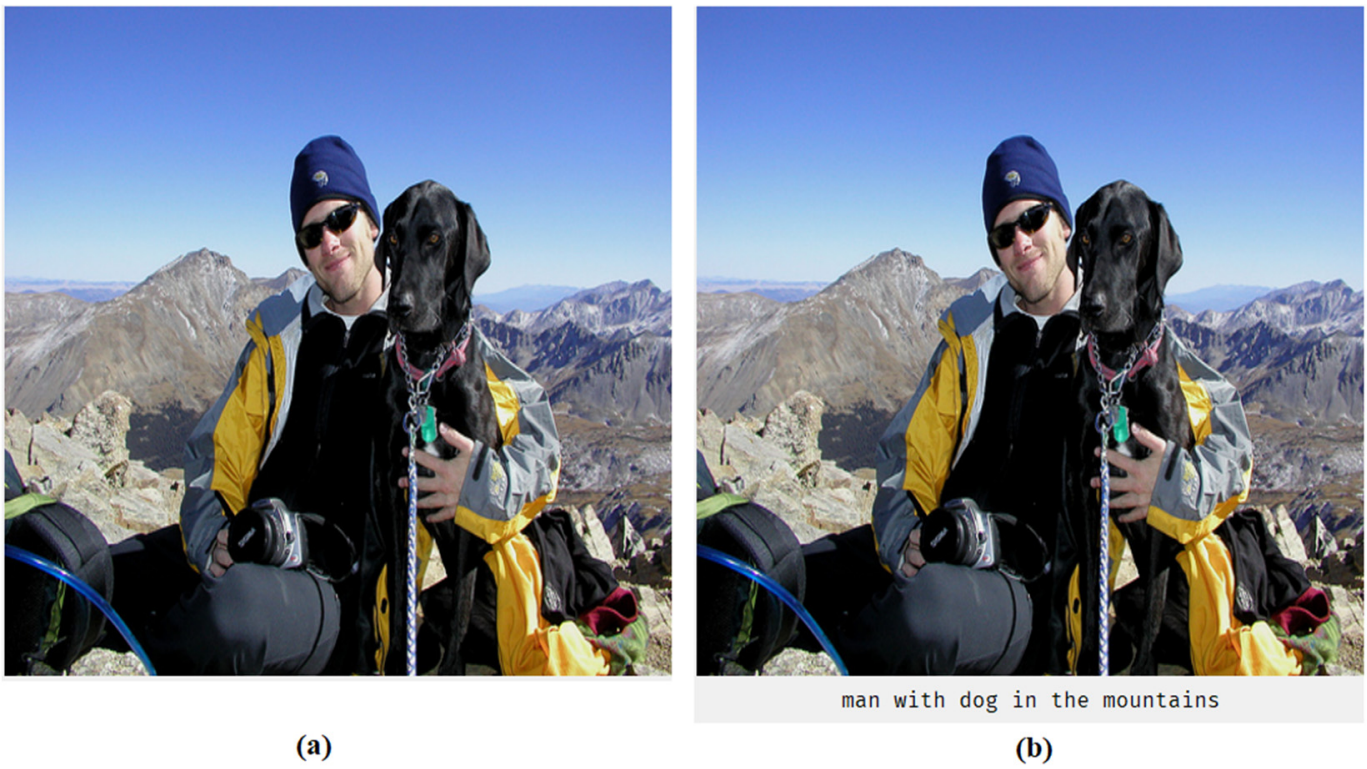

Figure 4. (**a**) Original image. (**b**) Captioning image.

4.4. Results Analysis on Flickr8K Dataset

Table 1 and Figure 5 provide BLEU analysis of the HPTDL-AIC technique on the test Flickr8K dataset. The results show that the M-RNN and DVS techniques obtained ineffective results with the minimal values of BLEU. At the same time, GoogleNICG and ResNet50 models obtained slightly enhanced values of BLEU. Eventually, L-Bilinear and VGA-16 models reached moderately reasonable values of BLEU. However, the proposed HPTDL-AIC technique has showcased enhanced performance over the other methods with a maximum BLEU-1, BLEU-2, BLEU-3, and BLEU-4 of 0.679, 0.461, 0.378, and 0.273, respectively.

Table 1. Result analysis of HPTDL-AIC technique in terms of BLEU on Flickr8K dataset.

Methods	BLEU-1	BLEU-2	BLEU-3	BLEU-4
M-RNN	0.585	0.290	0.240	0.149
Google NICG	0.639	0.419	0.277	0.160
L-Bilinear	0.662	0.429	0.282	0.182
DVS	0.588	0.385	0.254	0.168
ResNet50	0.624	0.458	0.370	0.266
VGA-16	0.674	0.442	0.340	0.227
HPTDL-AIC	0.679	0.461	0.378	0.273

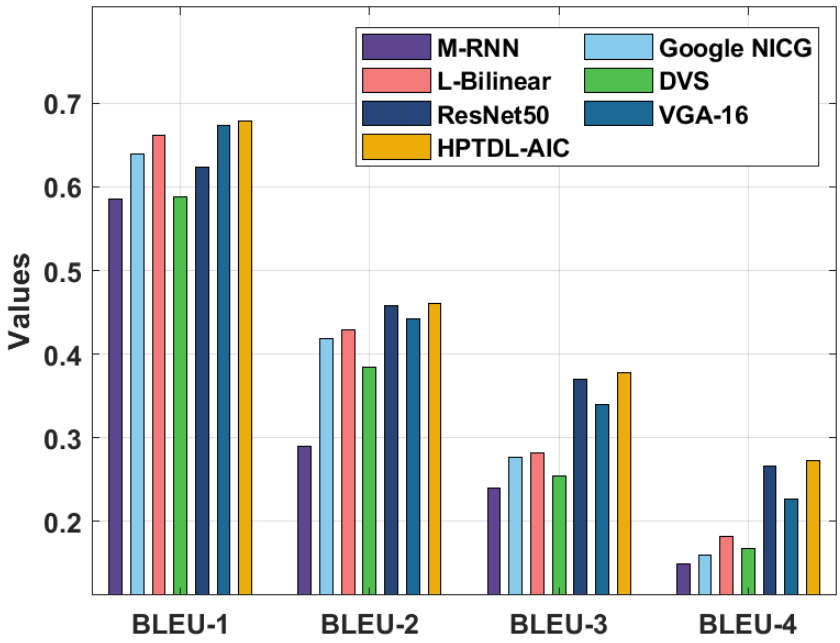

Figure 5. Result analysis of HPTDL-AIC technique on Flickr8K dataset.

A brief comparative study of the HPTDL-AIC technique on the test HPTDL-AIC technique with recent methods is provided in Table 2. Figure 6 investigates the Meter analysis of the HPTDL-AIC technique with existing techniques on the Flickr8K dataset. The figure demonstrated that GoogleNIC and A-NIC techniques obtained poor outcomes with the least Meter values of 20 and 20, respectively. In line with this, the SCST-IN, SCST-ALL, and DenseNet models portrayed certainly increased Meter values of 23, 24, and 23, respectively. However, the presented HPTDL-AIC technique outperformed the other ones with a maximum Meter value of 26.

Table 2. Comparative analysis of HPTDL-AIC technique with existing approaches on Flickr8K dataset.

Methods	Meter	CIDEr	Rouge-L
SCST-IN	23.00	159.00	45.00
SCST-ALL	24.00	156.00	45.00
Google NIC	20.00	153.00	46.00
A-NIC	20.00	156.00	47.00
DenseNet	23.00	168.00	47.00
HPTDL-AIC	26.00	171.00	50.00

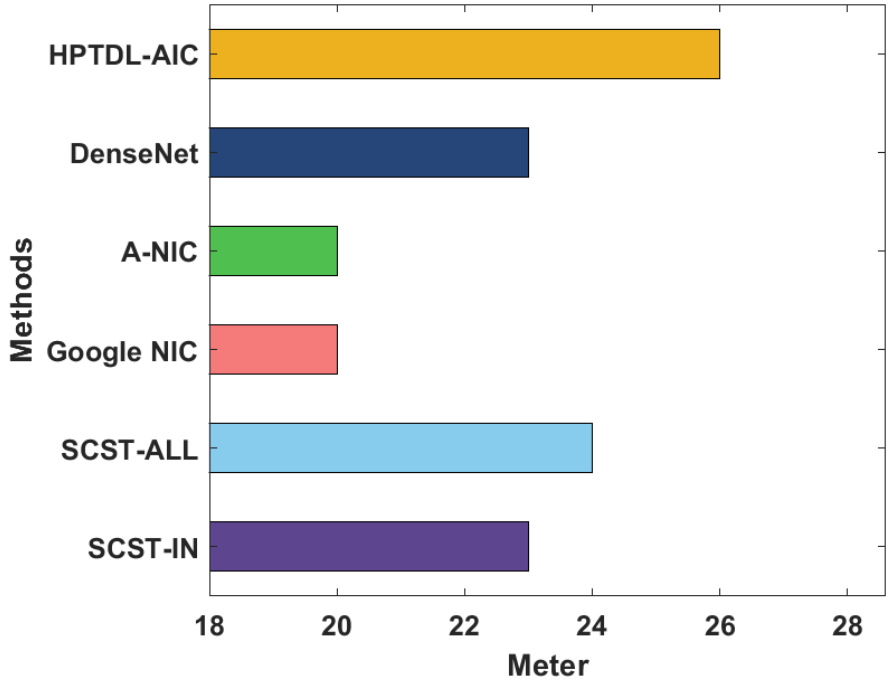

Figure 6. Meter analysis of HPTDL-AIC technique on Flickr8K dataset.

Next, the comparative CIDEr analysis of the HPTDL-AIC technique on the test Flickr8K dataset is performed in Figure 7. The results reported that the Google NIC technique resulted in lowering performance with a CIDEr value of 153. Subsequently, SCST-IN, SCST-ALL, and A-NIC techniques have reached moderately closer CIDEr values of 159, 156, and 156, respectively. Although the DenseNet model has accomplished a reasonable CIDEr value of 168, the HPTDL-AIC technique has exhibited improved outcomes with a higher CIDEr value of 171.

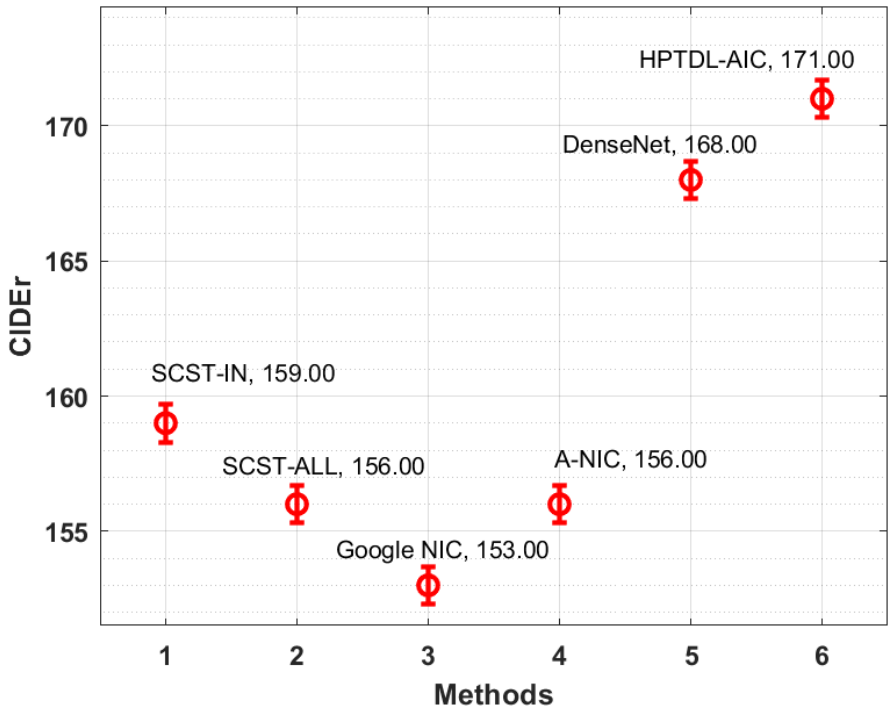

Figure 7. CIDEr analysis of HPTDL-AIC technique on Flickr8K dataset.

Figure 8 examines the ROUGE-L analysis of the HPTDL-AIC technique with existing techniques on the Flickr8K dataset. The figure revealed that the SCST-IN and SCST-ALL techniques have acquired reduced outcomes with minimum ROUGE-L values of 45 and 45, respectively. Along with that, Google NIC, A-NIC, and DenseNet models portrayed certainly increased ROUGE-L values of 46, 47, and 47, respectively. However, the presented HPTDL-AIC technique outpaced the other ones with an increased ROUGE-L value of 50.

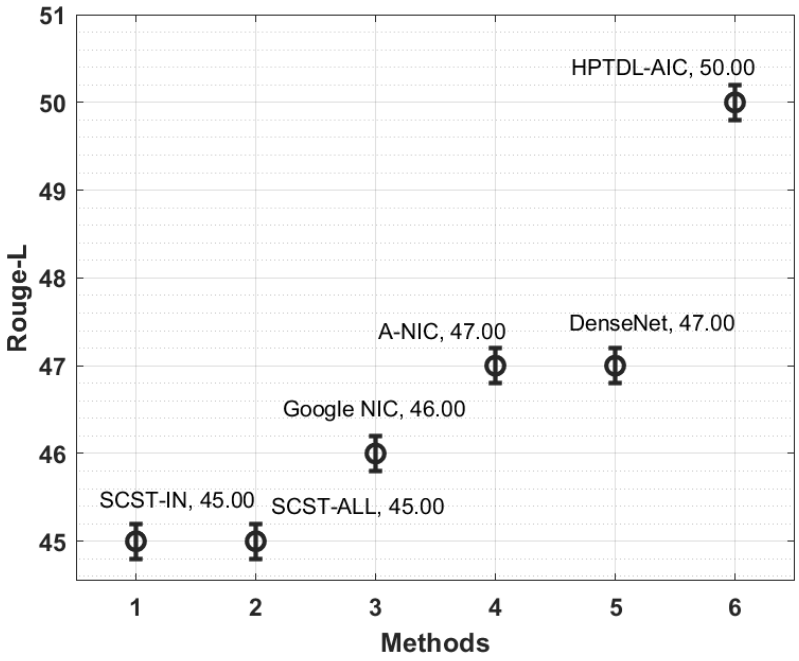

Figure 8. Rouge-L analysis of HPTDL-AIC technique on Flickr8K dataset.

The accuracy results analysis of the HPTDL-AIC system on the test Flickr8K is demonstrated in Figure 9. The results showcased that the HPTDL-AIC technique has resulted in increased training and validation accuracies. It can be clear that the HPTDL-AIC technique has the ability of gained maximum validation accuracy over training accuracy.

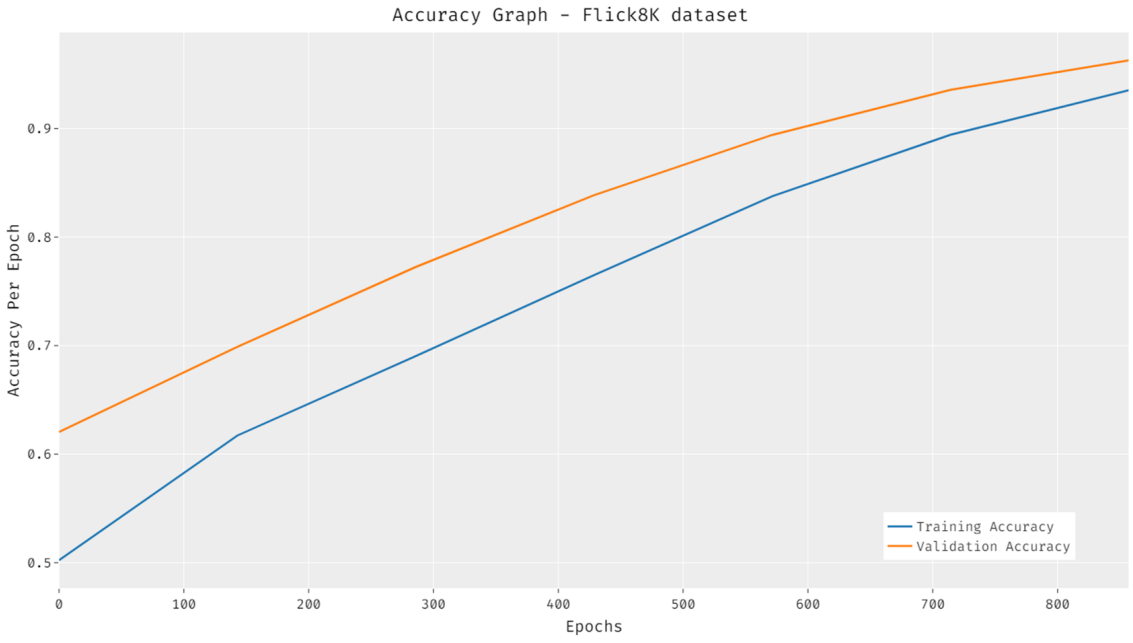

Figure 9. Accuracy analysis of HPTDL-AIC technique on Flickr8K dataset.

The loss outcome analysis of the HPTDL-AIC technique on the test Flickr8K is provided in Figure 10. The figure referred that the HPTDL-AIC technique has reached lower training and validation losses. It can be noted that the HPTDL-AIC method has the capability of accomplishing a reduction in validation loss and overtraining loss.

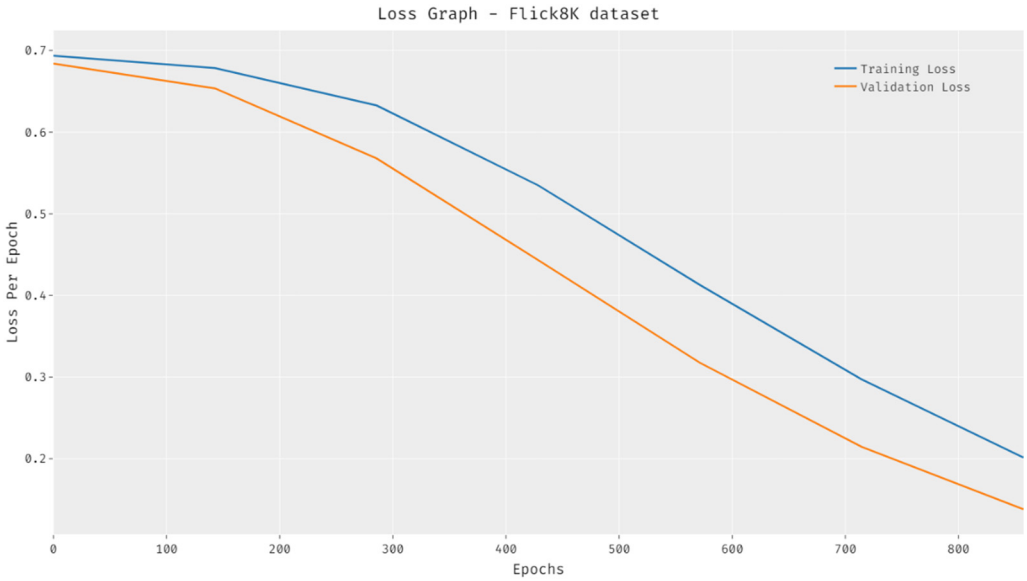

Figure 10. Loss analysis of HPTDL-AIC technique on Flickr8K dataset.

4.5. Results Analysis on MS COCO 2014 Dataset

Table 3 and Figure 11 offer a brief comparative BLEU analysis of the HPTDL-AIC technique on the test MS COCO 2014 dataset. The results demonstrated that the M-RNN and DVS techniques attained worse outcomes with the least BLEU. In addition, GoogleNICG and ResNet50 models reached somewhat improved values of BLEU. Next to that, L-Bilinear and VGA-16 models resulted in moderately sensible values of BLEU. However, the proposed HPTDL-AIC technique has reported improved outcomes over the other methods with higher BLEU-1, BLEU-2, BLEU-3, and BLEU-4 of 0.742, 0.587, 0.428, and 0.343, respectively.

Table 3. Result analysis of HPTDL-AIC technique in terms of BLEU on MS COCO 2014 dataset.

Methods	BLEU-1	BLEU-2	BLEU-3	BLEU-4
KNN	0.489	0.288	0.168	0.103
Google NICG	0.673	0.463	0.339	0.252
L-Bilinear	0.713	0.497	0.349	0.251
DVS	0.633	0.457	0.328	0.235
ResNet50	0.739	0.568	0.413	0.331
VGA16	0.707	0.544	0.400	0.299
HPTDL-AIC	0.742	0.587	0.428	0.343

A detailed comparative analysis of the HPTDL-AIC technique on the test HPTDL-AIC technique with recent methods is portrayed in Table 4 [15]. Figure 12 examines the Meter analysis of the HPTDL-AIC technique with existing techniques on the MS COCO 2014 dataset. The figure revealed that the GoogleNIC and A-NIC techniques have gained reduced outcomes with minimal Meter values of 24 and 23, respectively. Along with that, SCST-IN, SCST-ALL, and DenseNet models have depicted surely amplified Meter values of

26, 27, and 25, respectively. However, the presented HPTDL-AIC technique has outpaced the other ones with a supreme Meter value of 30.

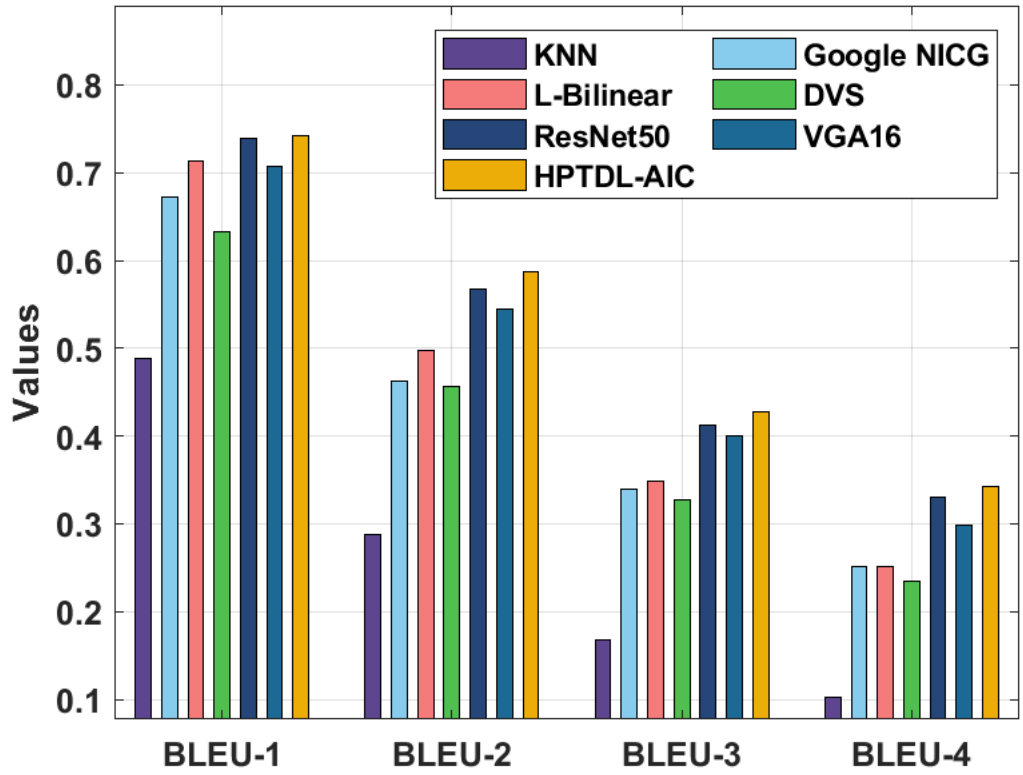

Figure 11. Result analysis of HPTDL-AIC technique on MS COCO 2014 dataset.

Table 4. Comparative analysis of HPTDL-AIC technique with existing approaches on MS COCO 2014 dataset.

Methods	Meter	CIDEr	Rouge-L
SCST-IN	26.00	111.00	55.00
SCST-ALL	27.00	114.00	56.00
Google NIC	24.00	108.00	55.00
A-NIC	23.00	106.00	55.00
DenseNet	25.00	118.00	57.00
HPTDL-AIC	30.00	121.00	61.00

Subsequently, a comparative CIDEr analysis of the HPTDL-AIC technique on the test MS COCO 2014 dataset is performed in Figure 13. The results stated that the A-NIC technique has resulted in inferior performance with the CIDEr value of 106. At the same time, SCST-IN, SCST-ALL, and Google NIC techniques accomplished considerable CIDEr values of 111, 114, and 108, respectively. Although the DenseNet model has depicted a high CIDEr value of 118, the HPTDL-AIC technique has presented a better outcome with a higher CIDEr value of 111.

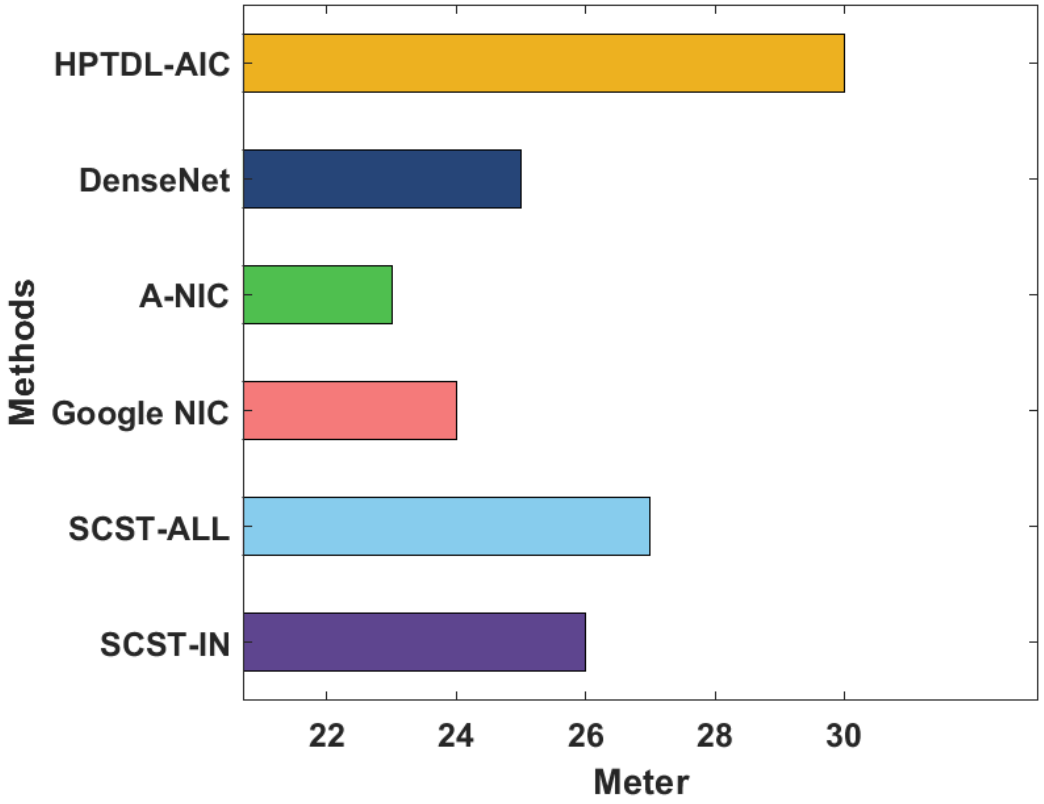

Figure 12. Meter analysis of HPTDL-AIC technique on MS COCO 2014 dataset.

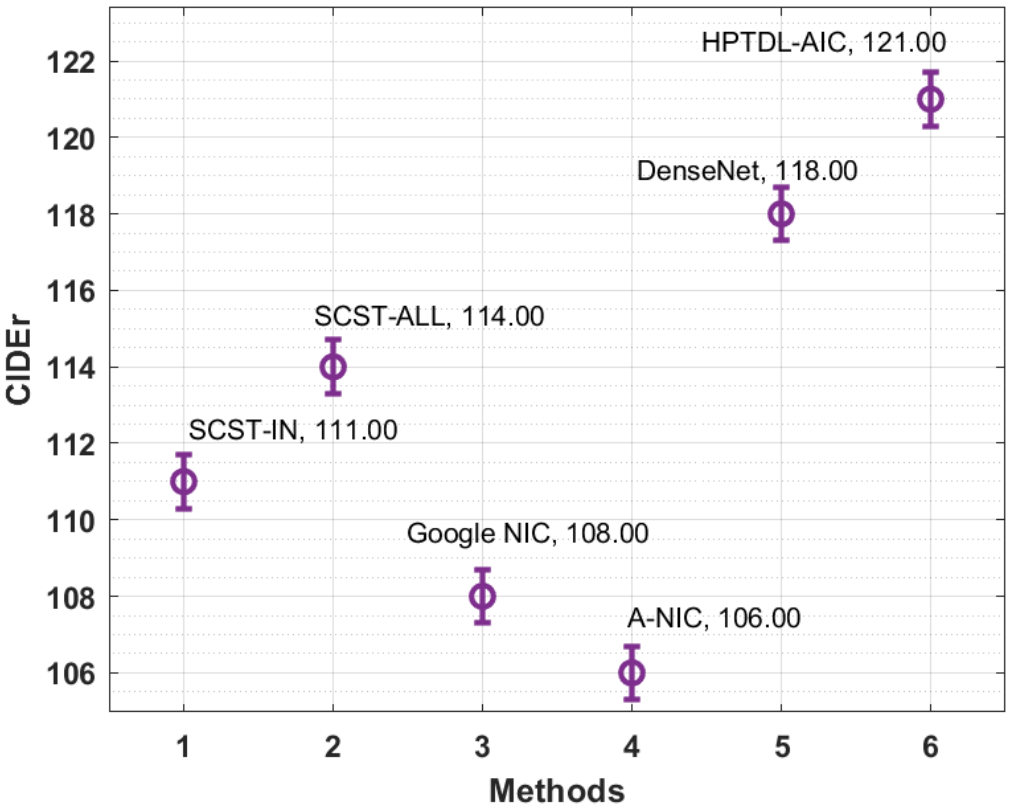

Figure 13. CIDEr analysis of HPTDL-AIC technique on MS COCO 2014 dataset.

Figure 14 inspects the ROUGE-L analysis of the HPTDL-AIC technique with existing techniques on the MS COCO 2014 dataset. The figure exposed that the SCST-IN, A-NIC, and Google NIC techniques have attained reduced outcomes with the lowest ROUGE-L values of 55, 55, and 55, respectively. Moreover, SCST-ALL and DenseNet models have portrayed certainly increased ROUGE-L values of 56 and 57, respectively. However, the presented HPTDL-AIC technique has outperformed the others with a better ROUGE-L value of 61.

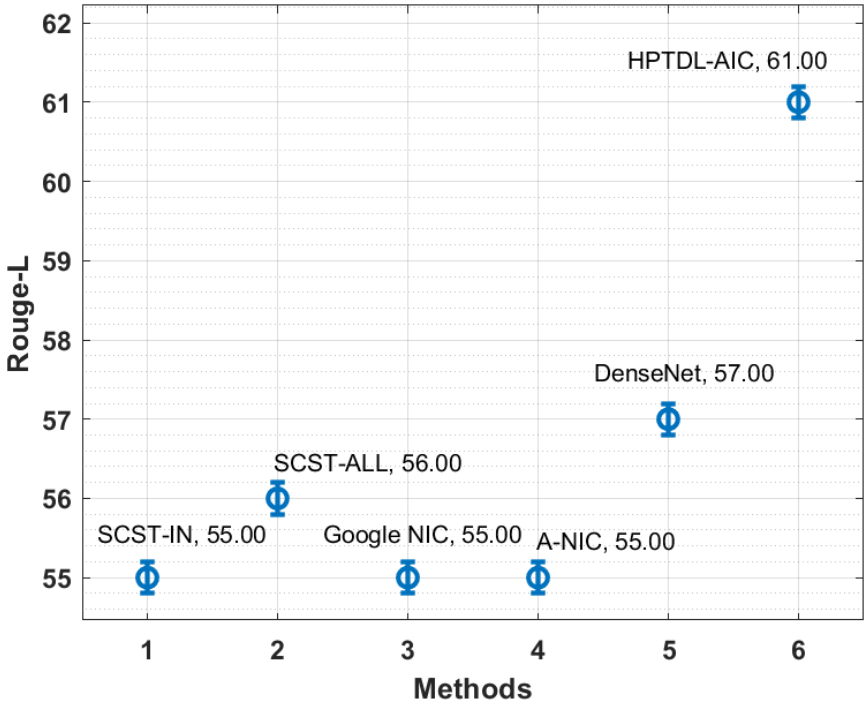

Figure 14. Rouge-L analysis of HPTDL-AIC technique on MS COCO 2014 dataset.

The accuracy outcomes analysis of the HPTDL-AIC technique on the test MSCOCO 2014 is displayed in Figure 15. The results showcased that the HPTDL-AIC method resulted in maximum training and validation accuracy. It can be stated that the HPTDL-AIC manner can attain increased validation accuracy on training accuracy.

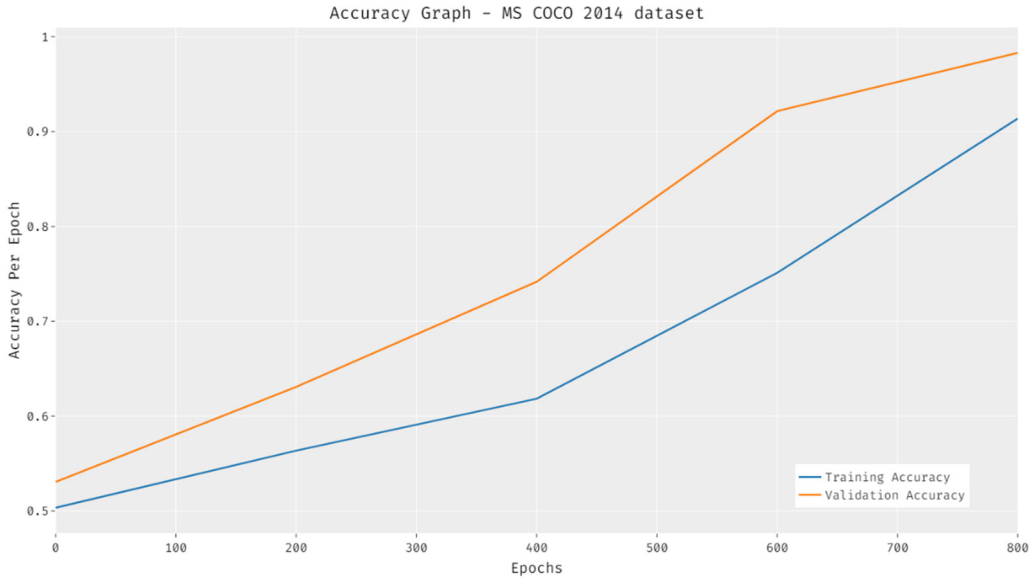

Figure 15. Accuracy analysis of HPTDL-AIC technique on MS COCO 2014 dataset.

The loss outcomes analysis of the HPTDL-AIC technique on the test MSCOCO 2014 is offered in Figure 16. The figure portrayed that the HPTDL-AIC technique has gained reduced training and validation losses. It is noticeable that the HPTDL-AIC approach has the capability of accomplishing decreased validation loss overtraining loss. From the analysis of the abovementioned results, it is apparent that the HPTDL-AIC technique has been employed as an efficient method for image captioning applications in real time.

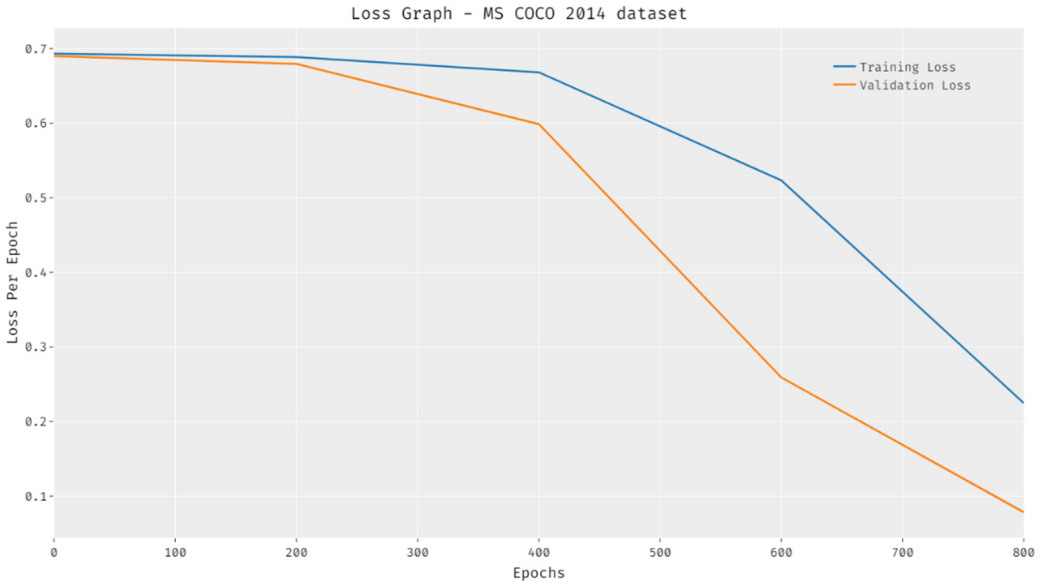

Figure 16. Loss analysis of HPTDL-AIC technique on MS COCO 2014 dataset.

5. Conclusions

In this study, a novel HPTDL-AIC technique has been developed to generate image captions automatically. The HPTDL-AIC technique intends to create correct descriptions for input images by the use of encoder–decoder structures. In particular, the encoder unit includes the Faster SqueezeNet with RMSProp model for generating a one-dimensional vector representation of the input image. Then, the BSA with the LSTM model is utilized as a decoder to produce description sentences and decode the vector into a sentence. For examining enhanced outcomes of the HPTDL-AIC technique, a series of simulations was performed on two benchmark datasets, and the extensive comparative study pointed out the improvement of the HPTDL-AIC technique over recent approaches. The experimental results stated that the inclusion of the hyperparameter tuning process results in improved captioning performance compared to other methods. Therefore, the HPTDL-AIC technique can be utilized as an effective tool for image captioning in NLP tasks. In the future, hybrid DL models can be employed for language modeling to boost overall performance.

Author Contributions: Conceptualization, M.O.; data curation, M.O.; formal analysis, M.O. and J.B.; investigation, S.A.-K.; methodology, S.A.-K.; project administration, G.P.J.; software, E.M.K.; supervision, G.P.J.; validation, J.B. and G.P.J.; visualization, E.M.K. and G.P.J.; writing—original draft, M.O.; writing—review and editing, G.P.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing is not applicable to this article as no datasets were generated during the current study.

Acknowledgments: This project was funded by the Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, under grant No. (D-77-305-1442). The authors, therefore, gratefully acknowledge DSR technical and financial support.

Conflicts of Interest: The authors declare that they have no conflict of interest. The manuscript was written through contributions of all authors. All authors have given approval to the final version of the manuscript.

References

1. Huang, W.; Wang, Q.; Li, X. Denoising-based multiscale feature fusion for remote sensing image captioning. *IEEE Geosci. Remote Sens. Lett.* **2020**, *18*, 436–440. [CrossRef]
2. Hossain, M.Z.; Sohel, F.; Shiratuddin, M.F.; Laga, H. A comprehensive survey of deep learning for image captioning. *ACM Comput. Surv. (CsUR)* **2019**, *51*, 1–36. [CrossRef]
3. Wang, C.; Yang, H.; Bartz, C.; Meinel, C. Image captioning with deep bidirectional LSTMs. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 988–997.
4. Sharma, P.; Ding, N.; Goodman, S.; Soricut, R. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; Volume 1, pp. 2556–2565.
5. Yao, T.; Pan, Y.; Li, Y.; Mei, T. Incorporating copying mechanism in image captioning for learning novel objects. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6580–6588.
6. Hoxha, G.; Melgani, F.; Demir, B. Toward remote sensing image retrieval under a deep image captioning perspective. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 4462–4475. [CrossRef]
7. Liu, H.; Wang, G.; Huang, T.; He, P.; Skitmore, M.; Luo, X. Manifesting construction activity scenes via image captioning. *Autom. Constr.* **2020**, *119*, 103334. [CrossRef]
8. Li, Y.; Yao, T.; Pan, Y.; Chao, H.; Mei, T. Pointing novel objects in image captioning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 12497–12506.
9. Ren, Z.; Wang, X.; Zhang, N.; Lv, X.; Li, L.J. Deep reinforcement learning-based image captioning with embedding reward. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 290–298.
10. Kesavan, V.; Muley, V.; Kolhekar, M. Deep Learning based Automatic Image Caption Generation. In Proceedings of the 2019 Global Conference for Advancement in Technology (GCAT), Bangalore, India, 18–20 October 2019; pp. 1–6.
11. Wang, E.K.; Zhang, X.; Wang, F.; Wu, T.Y.; Chen, C.M. Multilayer dense attention model for image caption. *IEEE Access* **2019**, *7*, 66358–66368. [CrossRef]
12. Sharma, H. A Novel Image Captioning Model Based on Morphology and Fisher Vectors. In *Proceedings of International Conference on Communication and Artificial Intelligence*; Springer: Singapore, 2021; pp. 483–493.
13. Cheng, C.; Li, C.; Han, Y.; Zhu, Y. A semi-supervised deep learning image caption model based on Pseudo Label and N-gram. *Int. J. Approx. Reason.* **2021**, *131*, 93–107. [CrossRef]
14. Zeng, X.; Wen, L.; Liu, B.; Qi, X. Deep learning for ultrasound image caption generation based on object detection. *Neurocomputing* **2020**, *392*, 132–141. [CrossRef]
15. Shen, X.; Liu, B.; Zhou, Y.; Zhao, J.; Liu, M. Remote sensing image captioning via Variational Autoencoder and Reinforcement Learning. *Knowl.-Based Syst.* **2020**, *203*, 105920. [CrossRef]
16. Xu, Y.; Yang, G.; Luo, J.; He, J. An Electronic Component Recognition Algorithm Based on Deep Learning with a Faster SqueezeNet. *Math. Probl. Eng.* **2020**, *2020*, 2940286. [CrossRef]
17. Chu, Y.; Yue, X.; Yu, L.; Sergei, M.; Wang, Z. Automatic image captioning based on ResNet50 and LSTM with soft attention. *Wirel. Commun. Mob. Comput.* **2020**, *2020*, 8909458. [CrossRef]
18. Meng, X.B.; Gao, X.Z.; Lu, L.; Liu, Y.; Zhang, H. A new bio-inspired optimisation algorithm: Bird Swarm Algorithm. *J. Exp. Theor. Artif. Intell.* **2016**, *28*, 673–687. [CrossRef]
19. Phan, N.H.; Hoang, V.D.; Shin, H. Adaptive combination of tag and link-based user similarity in flickr. In Proceedings of the 18th ACM International Conference on Multimedia, Firenze, Italy, 25–29 October 2010; pp. 675–678. Available online: https://www.kaggle.com/adityajn105/flickr8k/activity (accessed on 14 August 2021).
20. Vinyals, O.; Toshev, A.; Bengio, S.; Erhan, D. Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 652–663. [CrossRef] [PubMed]
21. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; pp. 311–318.
22. Denkowski, M.; Lavie, A. Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In Proceedings of the Sixth Workshop on Statistical Machine Translation, Edinburgh, UK, 30–31 July 2011; pp. 85–91.

23. Vedantam, R.; Lawrence Zitnick, C.; Parikh, D. Cider: Consensus-based image description evaluation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4566–4575.
24. Lin, C.-Y. ROUGE: A package for automatic evaluation of summaries. In Proceedings of the ACL-Workshop, Text Summarization Branches Out, Barcelona, Spain, 25–26 July 2004; pp. 1–8.