Image Description Generator using Residual Neural Network and Long Short-Term Memory

Mahesh Kumar Morampudi, Nagamani Gonthina,
Nuthanakanti Bhaskar, V. Dinesh Reddy

Abstract

Human beings can describe scenarios and objects in a picture through vision easily whereas performing the same task with a computer is a complicated one. Generating captions for the objects of an image helps everyone to understand the scenario of the image in a better way. Instinctively describing the content of an image requires the apprehension of computer vision as well as natural language processing. This task has gained huge popularity in the field of technology and there is a lot of research work being carried out. Recent works have been successful in identifying objects in the image but are facing many challenges in generating captions to the given image accurately by understanding the scenario. To address this challenge, we propose a model to generate the caption for an image. Residual Neural Network (ResNet) is used to extract the features from an image. These features are converted into a vector of size 2048. The caption generation for the image is obtained with Long Short-Term Memory (LSTM). The proposed model is experimented on the Flickr8K dataset and obtained an accuracy of 88.4%. The experimental results indicate that our model produces appropriate captions compared to the state of art models.

Keywords: Image, description generator, ResNet, LSTM.

MSC 2020: 68T45, 68T50, 68U15.

ACM CCS Codes: Image Processing and Natural Language Processing → Scene Recognition and Text generation.

doi:10.56415/csjm.v31.01

1 Introduction

Human beings possess a basic talent of describing an image clearly and elaborately by just watching it for a few seconds. Steady research is being conducted in the areas of machine learning and artificial intelligence in building a machine to mimic the capabilities of humans. In the past, extensive research progress was conducted in the areas such as the diagnosis of objects in a prescribed image, feature categorization, image grading, and categorization of activities by humans [1]–[4]. Detection of the image as well as producing the caption with natural language processing (known as an image caption generator system) by the computer is a tedious job, image caption generator system can be used as a solution to various problems such as self-driving cars, aid to the blind, etc. The captions or descriptions for an image are generated from an inverse dictionary that is formed during the training of the model. Automatic image description generation is useful in various fields like picture cataloging, blind persons, social media, and various natural language processing applications.

Generating the description for a picture involves multiple jobs like analyzing the importance in usage of semantics, and framing the meanings in a phrase through which humans can understand. To analyze the usage of semantics, the machine must grasp the relations amongst the things within an image. Generally, presentation in humans happens using natural language, hence building a computer system that generates captions that are acceptable to humans is an exciting task. We have multiple ways to build descriptions, like recognizing visual depiction of objects, setting up relations between the objects, and creating descriptions that are both grammatically and meaningfully perfect. In recent times, there is an immense growth in the availability of digital information on the Internet. One such application is Flickr, a means utilized for exporting, assembling, and distributing computerized information like pictures, audio, and video which can host more than 7 billion images, and this number is going to increase exponentially in the coming years. This application helps to find the images for training and testing a model with ease and helps the research community to describe an image. The image description is a simple process of

allocating words or phrases, which forms meaning to an image and describes the image. In earlier times, picture caption generation techniques agglomerated image knowledge with static object class libraries in the picture and were designed with the help of statistical language models. Few unintended techniques are even proposed to deal with picture caption descriptor issues, like the query evolution process suggested by Yagcioglu et al. [5], through fetching related pictures within a huge dataset and employing the allocation represented with a correlation of the fetched images. A huge amount of research done was prone to the issue of ranking descriptions for a given picture [3], [4], [6]. These methodologies depend on the possibility of co-embedding pictures and text data into a related vector space. Neural networks are utilized to co-embed pictures and text together, picture selections, and sub-sentences [7], [8], yet didn't strive to create narrative captions. Detections and segments aggregated with an end caption using phrases holding distinguished objects and relationships are used to deduce a triad of scene components that are translated to text with the help of templates [9], [10]. A further complicated graph of recognition behind triads is shown by Kulkarni et al. [10]. Deep learning architecture was used for image caption generation in [11]. In [12], [13], the authors used Convolutional Neural Network (CNN) in combination with Long Short-Term Memory (LSTM) to produce image descriptions but fail to achieve promising accuracy.

The proposed work aims at creating relevant, fluent captions like humans do for the given images without depending on any object identifiers, classifiers, transcribed regulations, or heuristics. We used Flickr8K dataset to find the efficiency of the proposed model.

1.1 Motivation

The aim and motivation of the proposed work are the latest advancements in machine conversion, where the job is converting a sentence "Z" taken in an original language, to "X" of the destination language, by enhancing the conditional probability $P(Z|X)$. Machine translation was even attained through a set of different jobs like converting words separately, joining words, rearranging, and so on from the past few

years. The latest research shows that conversion can be achieved efficiently with Residual Neural Network (ResNet) and can achieve better performance when compared to the state of art approaches [14].

1.2 Contributions

- We propose a model to generate the captions for an image using ResNet and LSTM [15].
- A generative approach is presented in this article that issues textual data, instead of using computer vision techniques (cv). Rather than using object identifiers, the data in a fresh image is estimated depending on the image's similarity to accessible images in the dataset, and the description of the image is output.
- Our approach creates relevant, fluent captions, like humans do for the given images not depending on any object identifiers, classifiers, transcribed regulations, or heuristics.
- This generative approach has experimented on the publicly available Flickr8K dataset. Better results are achieved compared to the conventional model by investigating and intelligently extracting the semantic knowledge encoded in the image descriptions.

2 Related Work

The task of creating captions similar to natural language, derived from visual data has been studied in computer vision mostly in video applications [16], [17]. It produces complicated machines containing visual primitive recognizers mixed with a structured formal language, e.g., And-Or Graphs or logic systems, that were additionally changed to natural language through rule-based frameworks. These things were mostly handcrafted, comparatively sensitive, and are shown uniquely in restricted spaces, e.g., sports or traffic scenes.

A vast amount of work was prone to the issue of ranking descriptions for a given picture [3], [4], [6]. Such methodologies depend on the possibility of co-embedding images and text in a similar vector space.

For a picture query, captions that fall close to the image in the embedding space are retrieved. Mostly, neural networks are utilized to co-embed images and sentences together [8], or even combine cropped pictures and sub-sentences [7] together. They didn't strive to create narrative captions. The previously mentioned methods could not represent earlier hidden compositions of objects, even though the individual objects might have been seen in the training data. Besides this, they avoid conveying the issue of assessing how well the created caption is.

Most recently the issue of picture caption generation using natural text has acquired attention. Utilizing the improvements in object recognition, their features, and positions, permits us to take forward natural language generation systems, even though they are restricted in their phrasing. Farhadi et al. [8] proposed a method to deduce a triad of scene components that are translated to text using the templates. Li et al. [9] introduced a method to provide an end caption with phrases consisting of distinguished objects and relationships. An even more complicated graph of recognition behind triads is proposed by Kulkarni et al. [10] using template-based text generation. In Vinyals et al. [12], the generative model is trained in such a way that, given the training image the likelihood of the final description of sentence is maximized. The methods discussed so far are capable of narrating images "in the wild", but they are mostly handcrafted and fixed in the case of text generation. Lebret et al. [18] proposed a simple language model based on caption syntax statistics to produce appropriate captions for an identified test image with the phrases deduced. N. K. Kumar et al. [11] proposed Regional Object Detector (RODe) to detect, recognize, and generate descriptions that aim at deep learning to still enhance on top of the prevailing image description generator systems.

Kinghorn et al. [19] proposed a region-based deep learning architecture in image caption generation by using a regional object detector, recurrent neural network (RNN)-based attribute prediction, and an encoder-decoder language generator embedded with two RNNs to generate processed and thorough captions for an identified image. Z. Zhou et al. [20] proposed a deep hierarchical framework to recognize images and a syntactic tree-based model to generate the natural language respectively. To support on-line image search, these two models

are described to evenly draw out the features of human beings and various object classes to generate well-proportioned sentences narrating the exact actions in the image. Bo Dai et al. [21] framework is dependent on Conditional Generative Adversarial Networks (CGAN), which mutually analyzes a generator to generate captions constrained on images and an evaluator to examine the fitness of caption in the visual content. Y. H. Tan et al. [13] proposed phi-LSTM, which decodes image captions from phrase to sentence. It contains a phrase decoder that decodes noun phrases of irregular size, and an abbreviated sentence decoder to decode the abbreviated form of the image description. An absolute image description is generated by combining the generated phrases with sentences in the course of the conclusion stage.

The literature reveals that the techniques designed to generate the captions for an image uses convolutional neural networks; the other deep learning models fail to generate the appropriate description of the scenario in the given image. Therefore, we proposed a model using ResNet and LSTM to provide a description of the scenario in the given image efficiently and accurately.

3 Proposed System

We proposed an approach to create relevant, fluent captions like humans do for the given images, without depending on any object identifiers, classifiers, transcribed regulations, or heuristics. The proposed system consists of two phases: the training and testing phase. Preprocessing, data preparation, and creation of a model are the different operations involved in the training phase of the proposed system. During the testing phase, the image vector is generated from the image, and a description of the image is displayed as an output using the generated model. Figure 1 depicts the generic architecture of the proposed model.

3.1 Preprocessing

This phase consists of two tasks, 1) Preprocessing images 2) Preprocessing captions.

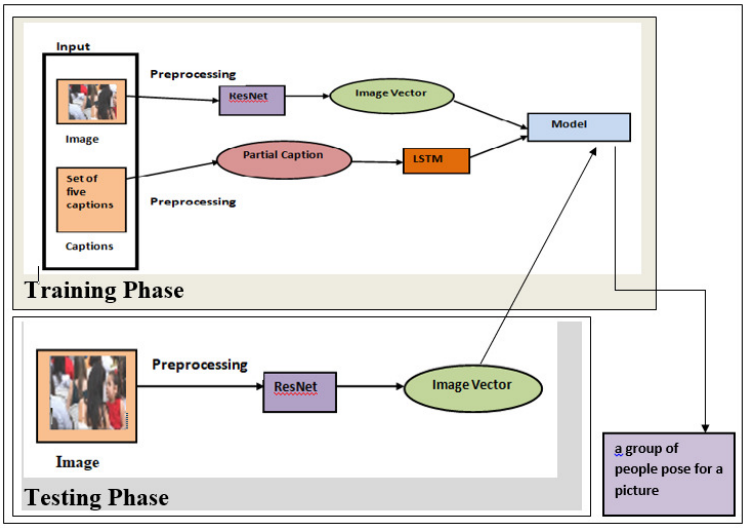


Figure 1. Block diagram of the proposed architecture

3.1.1 Preprocessing images

Image preprocessing is the process considered to set up images before training the model and inference. Images are given as input to the model in the form of an Image vector. Every image needs to be converted to a constant length vector that can be given as input to the neural network. To achieve the aforesaid, ResNet (Convolutional Neural Network) is used [14]. We created a dictionary named "image features", where the key will be the name of the image and the value will be output from the ResNet model. CV2 reads the image in BGR format, so the image is converted into RGB and resized to 224×224. It can be transferred into ResNet and reshaped to 2048 vector values. Retrieving only the name of the image from the whole path of the image is achieved using the slice function.

3.1.2 Preprocessing captions

During this step, each word is assigned an index. As there are 1652 unique words present in the database, each word is assigned a number from 1 to 1652. We also compute the maximum length of the caption.

The model will predict the caption for an image at the end. Given an image, it is difficult to predict the entire caption at once. So, we will predict Word by Word. In this regard, each word is encoded into a fixed-sized vector.

3.2 Data Preparation

Machine Learning or Deep Learning models cannot directly take the text and fit it into a model. Initially, the text is to be cleaned by dividing it into words, handling punctuation and case sensitivity problems. As English words can't be understood by the computers directly, they should be represented with numbers. Every word of the vocabulary should be matched to a unique index value, and each word is to be encoded into a fixed-length vector. There after every word is represented as a number. The text represented in binary numbers only can be readable by the machine and captions for the image can be generated. When the captions are run, the output will be images and the number of captions for a single image along with it. To achieve this, a dictionary is created with key as the image and all the other 5 captions of that image as the value. For the 1500 images available in the dataset, we need to check whether all the 1500 images append their captions. If the image name is available in the image feature and not in the captions dictionary, then we set the image name as a key and append those captions. Now we add $startofseq$ and $endofseq$ to the tokens. The prediction of the next word for a given image and partial description for text and numbers is shown in Figure 2 and Figure 3.

Create a dictionary Vocabulary that contains all the words in the captions is created with count=1 and checked word by word using the split function. If there is no count in $count\_words$, we set that word as a key and count as an integer value, and the value of the count is incremented by 1. len($count\_words$) is obtained as 40461. The words are converted into integer values since the neural network can only work with integer values. Previously there was: key – image name and values – captions; but now: key – integer and value – integer. Three variables, where the first one holds the image features, the second variable holds previous words, and the last variable holds

	X_i		Y_i
j	**Image Feature Vector**	**Partial Description**	**Target word**
1	Image_1	startofseq	a
2	Image_1	Startofseq a	group
3	Image_1	Startofseq a group	of
4	Image_1	Startofseq a group of	people
5	Image_1	Startofseq a group of people	pose
6	Image_1	Startofseq a group of people pose	for
7	Image_1	Startofseq a group of people pose for	a
8	Image_1	Startofseq a group of people pose for a	picture
9	Image_1	Startofseq a group of people pose for a picture	endofseq
10	Image_2	startofseq	cyclists
11	Image_2	Startofseq cyclists	in
12	Image_2	Startofseq cyclists in	a
13	Image_2	Startofseq cyclists in a	race
14	Image_2	Startofseq cyclists in a race	of
15	Image_2	Startofseq cyclists in a race of	bike
16	Image_2	Startofseq cyclists in a race of bike	is
17	Image_2	Startofseq cyclists in a race of bike is	riding
18	Image_2	Startofseq cyclists in a race of bike is riding	the
19	Image_2	Startofseq cyclists in a race of bike is riding the	busy
20	Image_2	Startofseq cyclists in a race of bike is riding the busy	street
21	Image_2	Startofseq cyclists in a race of bike is riding the busy street	endofseq

Figure 2. Data matrix for both images and captions

the next word to be predicted, are considered. Now we import the packages *to_categorical* and *pad_sequences* from keras. The value of MAX_LEN is 36. The $VOCAB\_SIZE$ will be the length of the count of words. We append image features to the x variable, y_in for input, and y_out for predicting the next word. $Pad\_sequence$ is used to convert the variable length to MAX_LEN. Figure 4 shows how the zeros are appended to each sequence to make them same length of 36. The *to_categorical* converts the out sequence into vocab size. It appends 0's and 1's, where 0 means the least probability, and 1 means maximum probability. When we check the length of all these variables, it will be 96528. So, to achieve faster execution, all the variables are converted to NumPy arrays.

	X_i		Y_i
j	Image Feature Vector	Partial Description	Target word
1	Image_1	[9]	10
2	Image_1	[9,10]	1
3	Image_1	[9,10,1]	2
4	Image_1	[9,10,1,2]	8
5	Image_1	[9,10,1,2,8]	6
6	Image_1	[9,10,1,2,8,6]	4
7	Image_1	[9,10,1,2,8,6,4]	7
8	Image_1	[9,10,1,2,8,6,4,7]	5
9	Image_1	[9,10,1,2,8,6,4,7,5]	3
10	Image_2	[9]	4
11	Image_2	[9,4]	7
12	Image_2	[9,4,7]	10
13	Image_2	[9,4,7,10]	5
14	Image_2	[9,4,7,10,5]	2
15	Image_2	[9,4,7,10,5,2]	6
16	Image_2	[9,4,7,10,5,2,6]	1
17	Image_2	[9,4,7,10,5,2,6,1]	11
18	Image_2	[9,4,7,10,5,2,6,1,11]	8
19	Image_2	[9,4,7,10,5,2,6,1,11,8]	13
20	Image_2	[9,4,7,10,5,2,6,1,11,8,13]	12
21	Image_2	[9,4,7,10,5,2,6,1,11,8,13,12]	3

Figure 3. Data matrix after replacing the words with their indexes

3.3 Creating a model

The architecture of the model is shown in Figure 5. The left layer represents captions, the right layer represents images as input, the center represents the LSTM model which is a concatenation of both values, and the bottom layer is the dense layer. Figure 5 consists of three parts: 1) Feature extractor, 2) Sequence Processor, and 3) Decoder. Feature extractor helps to extract the features from the image. The feature vector generated from this phase is of size 1×2048. The second part generates the image caption from the extracted features by using LSTM model. LSTM helps to carry out the relevant information and to discard non-relevant information. The last part decodes the ouput by concatenating the above two layers. It has 4074 probabilities for each vocabulary.

	X_i		Y_i
j	Image Feature Vector	Partial Description	Target word
1	Image_1	[9,0,0,....,0]	10
2	Image_1	[9,10,0,0,....,0]	1
3	Image_1	[9,10,1,0,0,....,0]	2
4	Image_1	[9,10,1,2,0,0,....,0]	8
5	Image_1	[9,10,1,2,8,0,0,....,0]	6
6	Image_1	[9,10,1,2,8,6,0,0,....,0]	4
7	Image_1	[9,10,1,2,8,6,4,0,0,....,0]	7
8	Image_1	[9,10,1,2,8,6,4,7,0,0,....,0]	5
9	Image_1	[9,10,1,2,8,6,4,7,5,0,0,....,0]	3
10	Image_2	[9,0,0,....,0]	4
11	Image_2	[9,4,0,0,....,0]	7
12	Image_2	[9,4,7,0,0,....,0]	10
13	Image_2	[9,4,7,10,0,0,....,0]	5
14	Image_2	[9,4,7,10,5,0,0,....,0]	2
15	Image_2	[9,4,7,10,5,2,0,0,....,0]	6
16	Image_2	[9,4,7,10,5,2,6,0,0,....,0]	1
17	Image_2	[9,4,7,10,5,2,6,1,0,0,....,0]	11
18	Image_2	[9,4,7,10,5,2,6,1,11,0,0,....,0]	8
19	Image_2	[9,4,7,10,5,2,6,1,11,8,0,0,....,0]	13
20	Image_2	[9,4,7,10,5,2,6,1,11,8,13,0,0,....,0]	12
21	Image_2	[9,4,7,10,5,2,6,1,11,8,13,12,0,0,....,0]	3

Figure 4. Appending zeros to each sequence to make them all of same length 36

Fitting the model Model fitting is a measure of the extent to which a machine learning model generates data similar to the one it was trained on. A model which is well-fitted generates the most appropriate results. We initialized the $batch\_size = 512$ and $epochs = 90$, and trained the model until we got the maximum accuracy, keeping in mind the problem of overfitting. The value of the epochs should be in such a way that the model should not get under-fitted or overfitted.

4 Experimental Results

Dataset: Flickr8k database [3] is used to validate the efficiency of the proposed model. The database consists of 8000 images and 5 English captions for each image which is taken from the online photo-sharing

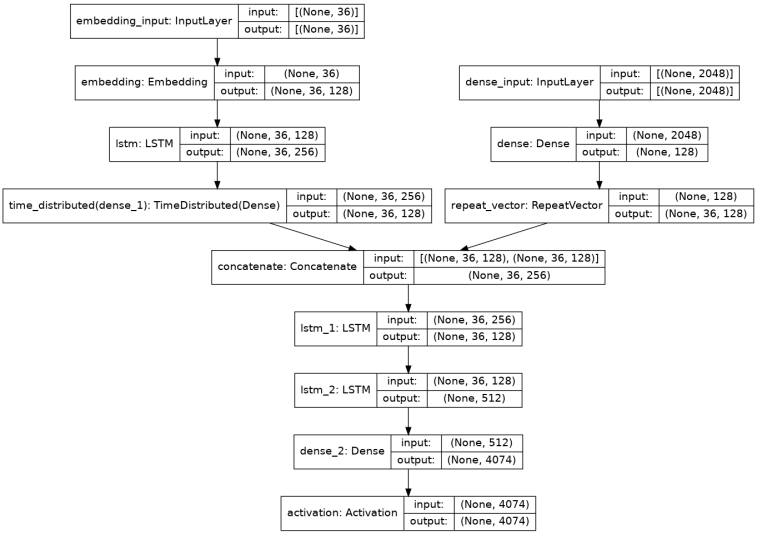


Figure 5. Architecture of the model

application Flickr.com. Out of which 6000 images are used for training and 1000 images are used for validation and testing. Annotators were asked to write sentences that describe the depicted scenes, situations, events, and entities (people, animals, other objects). Spoken captions for Flickr8k were collected by [22] having Amazon Mechanical Turk workers pronounce the originally written captions.

Training The first step in the process of generating comments to the image is to create a fixed-length vector that effectively summarizes the content of an image. We use CNN, in particular the ResNet50 architecture. This network is preliminarily trained for 1.2 million images of the ImageNet dataset. Therefore, ResNet50 has a reliable initialization for object recognition and allows reducing training time. For any image from the training set, we get the output vector representation of size 2048 from the last convolutional layer. This vector is fed to the LSTM input. During implementation, we considered 6000 images out of 8000 images for training.

Results: Figure 6 shows an example of how caption is generated using the proposed model. The first two images (Figure 6a and Figure 6b) and their captions are considered to train the model and the third image (Figure 6c) is used to test our model. It can be inferred from Figure 6 that the model builds the vocabulary using the captions of the train images. The model generates the caption for the test image using the vocabulary created during the training phase. The captions on some of the other images from the test dataset are shown in Figure 7a, Figure 7b, and Figure 7c. The accuracy achieved was 88.4% and the predictions made were almost correct.

(a)

(b)

(c)

Figure 6. An example: a) (Train image1) Caption: The black cat sat on the grass b) (Train image2) Caption: The white cat is walking on road c) (Test image) Caption: The black cat is walking on grass

The accuracy indicates that the proposed model is not an 100% best model and it also gives the captions wrongly in some scenarios. Figure 8 shows the images for which the model generates the captions wrongly. The color of the shirt got mixed with the color in the background in figure 8a. So, it generates the caption wrongly. The model classifies the famous tennis player Rafael Nadal as a woman in Figure 8b. This is due to his long hair. The caption generated in Figure 8c is grammatically incorrect.

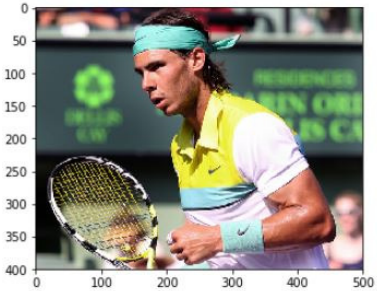
(a)

(b)

(c)

Figure 7. Some of the test images with their captions

(a)



(b)

(c)

Figure 8. The images for which the model generates the captions wrongly. a) man in black shirt is stakeboarding down ramp. b) a woman in tennis racket on the court. c) a boy is walking on the beach with ocean.

5 Conclusion and Future works

We introduced a system for creating relevant, fluent captions like humans do for the given images independent on any object identifiers, classifiers, transcribed regulations, or heuristics. Our model uses the ResNet to extract the features of an images and LSTM to provide the caption for an image. The proposed model is experimented on the publicly available database Flickr8K. The experimental results indicate that our model produces appropriate captions compared to the state-of-the-art methods.

Despite the fact that we have numerous enhancements in the area of image description generators, there is always a scope for development. Taking advantage of larger unsupervised data or weakly supervised methods is a challenge to explore in this area. Another major challenge could be generating summary or description for short videos. This work can also be extended to other sets of natural languages apart from English.

References

[1] E. Kim, S. Helal, and D. Cook, "Human activity recognition and pattern discovery," *IEEE pervasive computing*, vol. 9, no. 1, pp. 48–53, 2009.

[2] R. Bernardi, R. Cakici, D. Elliott, A. Erdem, E. Erdem, N. Ikizler-Cinbis, F. Keller, A. Muscat, and B. Plank, "Automatic description generation from images: A survey of models, datasets, and evaluation measures," *Journal of Artificial Intelligence Research*, vol. 55, pp. 409–442, 2016.

[3] M. Hodosh, P. Young, and J. Hockenmaier, "Framing image description as a ranking task: Data, models and evaluation metrics," *Journal of Artificial Intelligence Research*, vol. 47, pp. 853–899, 2013.

[4] V. Ordonez, G. Kulkarni, and T. Berg, "Im2text: Describing images using 1 million captioned photographs," *Advances in neural*

information processing systems, vol. 24, pp. 1143–1151, 2011.

[5] S. Yagcioglu, E. Erdem, A. Erdem, and R. Cakıcı, "A distributed representation based query expansion approach for image captioning," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 2015, pp. 106–111.

[6] A. Karpathy, A. Joulin, and L. Fei-Fei, "Deep fragment embeddings for bidirectional image sentence mapping," 2014.

[7] R. Socher, A. Karpathy, Q. V. Le, C. D. Manning, and A. Y. Ng, "Grounded compositional semantics for finding and describing images with sentences," *Transactions of the Association for Computational Linguistics*, vol. 2, pp. 207–218, 2014.

[8] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth, "Every picture tells a story: Generating sentences from images," in *European conference on computer vision*. Springer, 2010, pp. 15–29.

[9] S. Li, G. Kulkarni, T. Berg, A. Berg, and Y. Choi, "Composing simple image descriptions using web-scale n-grams," in *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, 2011, pp. 220–228.

[10] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg, "Babytalk: Understanding and generating simple image descriptions," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 12, pp. 2891–2903, 2013.

[11] N. K. Kumar, D. Vigneswari, A. Mohan, K. Laxman, and J. Yuvaraj, "Detection and recognition of objects in image caption generator system: A deep learning approach," in *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*. IEEE, 2019, pp. 107–109.

[12] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell:

A neural image caption generator," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3156–3164.

[13] Y. H. Tan and C. S. Chan, "Phrase-based image caption generator with hierarchical lstm network," *Neurocomputing*, vol. 333, pp. 86–100, 2019.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[16] R. Gerber and N.-H. Nagel, "Knowledge representation for the generation of quantified natural language descriptions of vehicle traffic in image sequences," in *Proceedings of 3rd IEEE international conference on image processing*, vol. 2. IEEE, 1996, pp. 805–808.

[17] B. Z. Yao, X. Yang, L. Lin, M. W. Lee, and S.-C. Zhu, "I2t: Image parsing to text description," *Proceedings of the IEEE*, vol. 98, no. 8, pp. 1485–1508, 2010.

[18] R. Lebret, P. O. Pinheiro, and R. Collobert, "Simple image description generator via a linear phrase-based approach," *arXiv preprint arXiv:1412.8419*, 2014.

[19] P. Kinghorn, L. Zhang, and L. Shao, "A region-based image caption generator with refined descriptions," *Neurocomputing*, vol. 272, pp. 416–424, 2018.

[20] Z. Zhou, K. Li, and L. Bai, "A general description generator for human activity images based on deep understanding framework," *Neural Computing and Applications*, vol. 28, no. 8, pp. 2147–2163, 2017.

[21] B. Dai, S. Fidler, R. Urtasun, and D. Lin, "Towards diverse and natural image descriptions via a conditional gan," in *Proceedings*

of the IEEE International Conference on Computer Vision, 2017, pp. 2970–2979.

[22] D. Harwath and J. Glass, "Deep multimodal semantic embeddings for speech and images," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 237–244.

Mahesh Kumar Morampudi, Nagamani Gonthina, Nuthanakanti Bhaskar, V. Dinesh Reddy

Received December 17, 2021
Revised 1 – May 17, 2022
Revised 2 – August 15, 2022
Accepted September 10, 2022

Mahesh Kumar Morampudi
ORCID: https://orcid.org/0000-0002-6888-4637
Department of Computer Science and Engineering
SRM University AP
Amaravathi
Andhra Pradesh
India
E–mail: morampudimahesh@gmail.com

Nagamani Gonthina
ORCID: https://orcid.org/0000-0001-9559-8030
Department of Computer Science and Engineering
Institute of Aeronautical Engineering
Hyderabad
Telangana
India
E–mail: gnvsk1986@gmail.com

Nuthanakanti Bhaskar
ORCID: https://orcid.org/0000-0001-9852-1004
Department of Computer Science and Engineering
CMR Technical Campus
Hyderabad

V. Dinesh Reddy
ORCID: https://orcid.org/0000-0003-3945-6171
Department of Computer Science and Engineering
SRM University AP
Amaravathi
Andhra Pradesh
India
E–mail: dineshvemula@gmail.com