2021 6th International Conference for Convergence in Technology (I2CT)
Pune, India. Apr 02-04, 2021

Generating Short Video Description using Deep-LSTM and Attention Mechanism

Naveen Yadav
Department of Information Technology
National Institute of Technology Karnataka
Mangalore, India
ny826yadav@gmail.com

Dinesh Naik
Department of Information Technology
National Institute of Technology Karnataka
Mangalore, India
dinnaik@gmail.com

2021 6th International Conference for Convergence in Technology (I2CT) | 978-1-7281-8876-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/I2CT51068.2021.9417907

***Abstract*—In modern days, extensive amount of data is produced from videos, because most of the populations have video capturing devices such as mobile phone, camera, etc. The video comprises of photographic data, textual data, and auditory data. Our aim is to investigate and recognize the visual feature of the video and to generate the caption so that users can get the information of the video in an instant of time. Many technologies capture static content of the frame but for video captioning, dynamic information is more important compared to static information. In this work, we introduced an Encoder-Decoder architecture using Deep-Long Short-Term Memory (Deep-LSTM) and Bahdanau Attention. In the encoder, Convolution Neural Network (CNN) VGG16 and Deep-LSTM are used for deducing information from frames and Deep-LSTM combined with attention mechanism for describing action performed in the video. We evaluated the performance of our model on MSVD dataset, which shows significant improvement as compared to the other video captioning model.**

***Index Terms*—Computer Vision, Machine Translation, Recurrent Neural Network, Natural Language Processing, Video Captioning.**

I. Introduction

Frequently producing caption for the video is trending nowadays. After the discovery of the CNN, researchers started to use it to solve challenging problems such as recognition of activity, detection of an object, recognition of an item in image/video, etc. It can be used in many intelligent control systems and IOT based device[30]. Generating captions for videos is indeed a more complicated task, it is open problem therefore many researchers[11], [12], [13], [15] have worked on it. In case of YouTube, Netflix and Amazon prime, trillions of video watched by people everyday. Our work is to summarise the video content in the form of text so that people can get the information of the video without watching the full video. It is also useful to classify the videos without watching those videos.

Video captaining task can be summarized in two generalize approach

- bottom-up approach: It first extracts the semantic facets and generates the word from those facets and using some rules and algorithms converts them into sentences[11], [12], [13].
- the top-down approach, first it creates a global representation of the whole video and tries to learn the sentences directly from that global representation [14], [15], [16], [17].

Initially researchers started video captaining using triplets available in the sentences(subject, verb, and object). Matching the objects form the training visual data to the triplets, After that, it generates the sentences using predefined templates in the testing process[18], [19]. This approach has very low accuracy and it is highly dependent on the syntactical structures of templates. The major drawback of this mechanism is that it only takes care of static features. In the case of an image, we have to combine only static features of the image, but in the case of video, we have to take care of the dynamic content and static content. The dynamic caption is generated when we move from one frame to another frame. Dynamic captions are more important in video captioning. Many times description may be inappropriate because of the very low grammatical corrections in the templates.

But top-down approach got very high success [20], [21], [22], [23]. It uses the encoder and decoder approach. These techniques basically used the end-to-end approach, It generates the sentence from the whole global representation which is made in the encoding phase. It generally uses the CNN layers for feature extraction and forms the global representation of the whole video later LSTM is used to form the sentences.

All these approaches focusing on each frame of video with equal importance therefore it suffers low accuracy, So we decided to develop a more sophisticated model using Deep-LSTM and Bahdanau Attention. It is producing better results and focusing on the relevant part of the video. The main contribution will be as follows.

- We utilised the basic Encoder-Decoder Architecture[20].
- In the encoder, initially frames are generated from video, then frames are given as input to pretrained CNN model, it generates the features vector. The generated feature vector is given as input to Deep-LSTM that deduce information from frames.
- The decoder consists of the Attention mechanism followed by Basic LSTM[24] based RNN[25] network. The combined output produced by the VGG16 model and Deep-LSTM given as input to the attention mechanism that decides on which part video should be focused in

order to predict the next word.

- Video caption is generated Using an Basic LSTM[24] Based RNN network from the previous stages representation.

II. Related Work

In the paper [1], researchers proposed hierarchical LSTM with attention mechanism for image and video captioning. In this paper, they have used spatial and temporal attention for selecting the regions in frames and adaptive attention is used whether to depend on language context or visual information present in the image. The hierarchical LSTM is designed to produce the caption by taking both low-level visual information and high-level language context information concurrently.

The approach in paper[2] used a dual-stream RNN framework that finds and combine the hidden states of both semantic and visual streams. Firstly, the video streams are passed through RNN based network to generate the latent space embedding. After that, an attention-based multi-grained encoder module is proposed to exploit the global semantics feature for the local feature learning. In the end, a dual-stream decoder is used to combine both semantic and visual features.

Authors Bin Zhao et. al. in paper [3] proposed a model based on recurrent neural network (CAM-RNN). In this author, describes that the Co-Attention Model CAM is used for encoding visual and also text features, whereas RNN is employed as a decoder in generating the caption of the video. CAM uses a text attention module, a visual attention module, and a balancing gate as a combination. While generating the caption the salient regions in each frame and its correlated frames with the caption are focused by the visual attention module. The most appropriate previously obtained words or phrases are focused by the text attention module. Balancing gate is used between the attention modules, which take care of how many models have to depend on visual features and text features during caption generation.

A spatial-temporal attention mechanism (STAT) for video caption generation using encoder-decoder approach proposed in paper[4]. This mechanism considers both temporal and spatial structures in the video which helps the decoder to choose the important regions in a video for word prediction on its own. Since just capturing subsets of the frames is not sufficient they have tried to capture the significant regions of the subsets too by taking both spatial and temporal features of the video into consideration.

In paper[5] dense video captioning task was developed as a new visual cue that helped to summarise sentences. During the division stage, multiple sentence descriptions are created to explain the various visual contents during the division stage. With hierarchical attention mechanisms, two-stage LSTM is used in the caption generation phase. The first layer of LSTM incorporates visual and textual features as a series of hidden representations, while the second-stage LSTM network, fitted with a newly proposed hierarchical attention system, functions as a decoder to produce one descriptive sentence.

III. Methodology

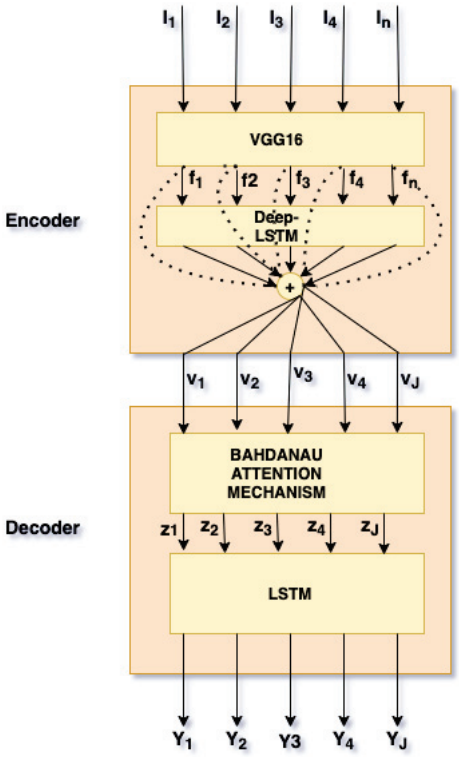

Figure 1: Encoder-Decoder-Architecture

There are four stages in Encoder-Decoder Architecture. CNN is used for describing the visual content of the image and the LSTM unit is used for finding the temporal relationship between the generated words.

- The Pretrained VGG16 model is used to obtain corresponding features from the video on each of the raw frames.
- To define actions carried out in successive frames, we have used Deep-LSTM for finding out the temporal relationships and complementary information.
- The output produced by all layers of Deep-LSTM are concatenated together with the VGG16 model output for each frame of video. This resultant output is feed to the bahdanau attention[6] model in the decoder. Now, on which portion of the frame we should focus in order to predict the next word by considering the description generated so far is decided by the attention model.
- The variable-length caption is generated word by word using the LSTM network from the previous stages representation.

A. Work Flow Diagram

In the initial phase of the model, we took the MSVD dataset and passed it as input. After cleaning the dataset, separate frames are generated for each video. VGG16 pretrained model is used to identifying the objects and actions which further

worked input for LSTM. Here LSTM is used as a cell in the RNN network to avoid the vanishing and exploding gradient problem.

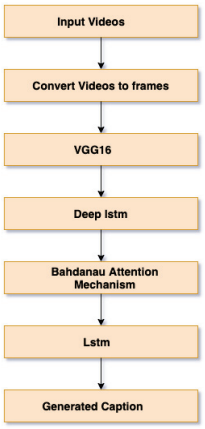

Figure 2: Overall Work-Flow Diagram

B. Encoder

we need to describe the video in order to

- understand what kind of object and structure are there
- what are the relationships and actions of these objects

There are various pretrained CNN models available for first part that may be used to describe the images. These pretrained models can be distinguished based on the architecture or dataset they are using. we are using VGG16 pretrained Architecture for object and context information. VGG16 generates a sequence Vc for a given video, that contains a feature vector each of dimension d and there are J such frames.

The feature vector V_c is processed by Deep-LSTM and it generates a new Sequence $V_b = v_1, v_2, ...v_J$ of J vector. Finally, the encoder combines the vectors created by the VGG16 and the Deep-LSTM (V_c, V_b) generating a final vector V of J dimension.

C. Decoder

The Decoder is an LSTM Network that acts as a Language model and takes information from the encoder. Apart from LSTM, the decoder consists of an attention mechanism. Attention mechanism is one of the most essential components of neural networks that is able to comprehend sequences, whether it be a real-life action sequence, voice, text, video sequence, or some other data. It is not surprising that our brain applies attention at many levels, in order to choose only the useful data to process, and remove the vast amount of background information that is not required for processing the task.

Attention helps the decoder to find out important components within the frame and help in producing each output word. The sequence V generated by the encoder is given as an input to the decoder and the decoder at each time step t applies the attention mechanism weights to J feature vector and combines them to form a single context vector C_t.

Finally, the LSTM network takes the context vector C_t generated by the bahdanau attention mechanism, previously generated word,and the last hidden state to generate the caption.

D. Deep LSTM

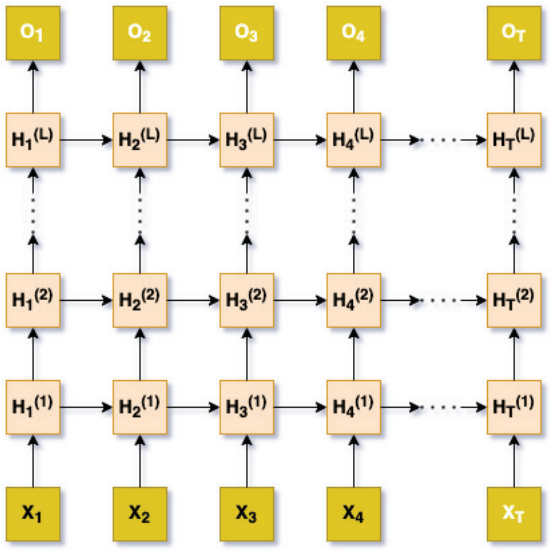

Figure 3: Deep Lstm Architecture

In the linear model, we can simply add the layers but in RNN it is little bit trickier because we have to decide where and how to add the extra non-linearity. we might place several RNN layers on top of each other. This turnout in a flexible method, because of the combination of several simple layers.

From Figure 3 we can understand Deep-LSTM architecture.It consists of L hidden layer. Each hidden state is continuously passed to the current time step of the next layer and the next time step of the current layer.

In deep architecture with L hidden layer shown above we can give shape to the functional dependencies.Let's consider input $X_t \in R^{n\times d}$ where n is number of example and d is input in each example at time step t. let the hidden state of l^{th} hidden layer at same time state be $H_t^{(l)} \in R^{n\times h}$ (where number of hidden units is represented by h) and the output layer variable be $O_t \in R^{n\times q}$ (where number of output is represented by q). Let $H_t^0 = X_t$, the hidden state of l^{th} hidden layer that uses the activation function ϕ. The gates at time step t are defined as follows:

$$\mathbf{I}_t = \sigma\left(\mathbf{X}_t\mathbf{W}_{xi} + \mathbf{H}_{t-1}\mathbf{W}_{hi} + \mathbf{b}_i\right) \quad (1)$$

$$\mathbf{F}_t = \sigma\left(\mathbf{X}_t\mathbf{W}_{xf} + \mathbf{H}_{t-1}\mathbf{W}_{hf} + \mathbf{b}_f\right) \quad (2)$$

$$\mathbf{O}_t = \sigma\left(\mathbf{X}_t\mathbf{W}_{xo} + \mathbf{H}_{t-1}\mathbf{W}_{ho} + \mathbf{b}_o\right) \quad (3)$$

Where $I_t \in R^{n\times h}$ represents Input gate, $F_t \in R^{n\times h}$ represents the Forget gate, and $O_t \in R^{n\times h}$ represents the Output gate. The weight Parameters are $W_{xi}, W_{xf}, W_{xo} \in R^{d\times h}$ and $W_{hi}, W_{hf}, W_{ho} \in R^{h\times h}$ and bias parameters are $b_i, b_f, b_o \in R^{1\times h}$.

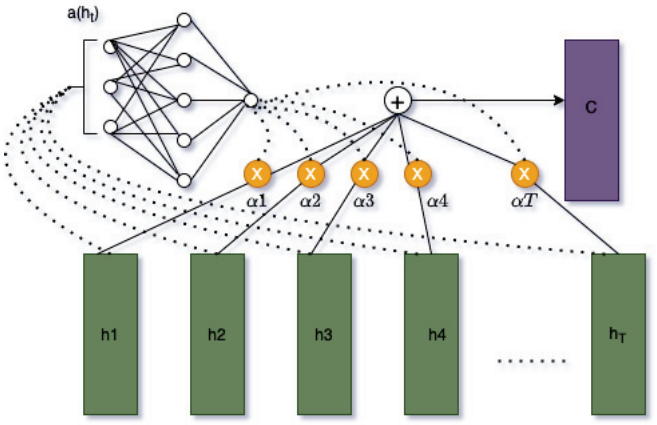

Figure 4: Bahdanau Attention Mechanism

E. Bahdanau Attention Mechanism

We can observe that the Performance of RNN[25] reduced when we give long series despite it is sequential. In general, the performance of vanilla RNN breaks down when the length of the sequence is 4 or 5. This is because of the Vanishing and exploding descent problem but the ability of vanilla RNN to learn long sequences ranging between 12-15 is increased with the help of LSTM and GRU[10]. However, in our dataset, most of sequences are of length greater than 18.

In 2014 Bahdanau et al.,[6] suggested an approach that uses an attention mechanism to alleviate the above issue. This mechanism learns from the sums of preceding outputs by forming an additional layer. This helps the network in determining and retaining the important features. so, this mechanism makes sure that even at longer sequence RNN performs well.

The attention mechanism makes use of the last state and the weighted mix of all the input states to predict the output for the current cell. keeping in mind the ultimate objective of obtaining output for the next cell the Attention weights allow the network to choose the sum of each previous input.

In this manner, it helps to determine which state's output is more necessary and useful for the current cell output. Mathematically, for a hidden state h_t at each time step,the context vector C_t is calculated as

$$C_t = \sum_{j=1}^{T} \alpha_{tj} h_j \tag{4}$$

where number of time steps represented by T, α_{t_j} is the calculated weight at time step t for state h_j. A new state sequence s_t is processed by this context vector where s_t depends upon model's output at time t-1, context vector C_t and past state sequence s_{t-1}. The weight αt_j is calculated as follows

$$e_{tj} = a(s_{t-1}, h_j) \tag{5}$$

$$\alpha_{tj} = \frac{\exp(\text{score}(e_{tj}, h_s))}{\sum_{k=1}^{T} \exp(e_{tk}, h_s)} \tag{6}$$

IV. Result Analysis

A. Data set

Our MSVD dataset consists of 1970 video snippets and 120 thousand sentences. This dataset generated in 2010 in which workers Mechanical Turk were paid to summarize the video snippet in a single sentence. The dataset is split into train, validation, and test dataset.

B. Sample Input and Output

(a) Frame At 1 sec

(b) Frame At 4sec

Figure 5: Test Clip 1
Generated Caption : a woman is cutting meat
Actual Caption : A woman is cutting octopus

(a) Frame At 2 sec

(b) Frame At 5sec

Figure 6: Test Clip 2
Generated Caption : a man is riding a bike
Actual Caption : A man is riding a motorcycle.

C. Evaluation Matrix

To calculate the efficiency We can not match the whole sentences, because everyone has a different way to caption the video and more than one caption may be correct therefore we need to try a different way. so the evaluation matrix came into the picture. First, we evaluate the different types of evaluation matrices. which is used to calculate the closeness of the captions. Some common evaluation matrices used are BLEU[7], ROUGE[8], METEOR[9], etc.

D. BLEU Score

To calculate BLUE[7], the generated text is matched against the set of already available references. The score value is calculated for each sentence but the BLEU score does not care about the syntactical correctness. It only takes care of the total numbers of words matched with the actual caption.

$$BLEU = \min\left(1, \frac{\text{generated length}}{\text{reference length}}\right)\left(\prod_{i=1}^{4}(\text{ precision }_i)^{1/4}\right) \tag{7}$$

Let C_W = correct words in generated sentences ,T_W total words in the generated sentence.

$$Precision = \frac{C_W}{T_W} \tag{8}$$

E. ROGUE Score

We can also calculate the ROUGE[8] score to observe the quality of generated caption. It matches the sequences of words, pairs of words, and n-gram with the reference. It is the set of matrices to evaluate the video caption/machine translation automatically. As we know that BLEU scores work on the n-gram i.e how many n-grams are matching from the generated text to reference text with respect to the total n-gram available in the generated text. By doing this we are only taking care of the precision but ROUGE takes care of both precision and recall. so ROUGE-1 tell about the precision and recall with respect to the uni-gram[8]. Let O_W = overlapping words,T_R Total words in reference text,T_G Total words in the generated text.

$$Precision = \frac{O_W}{T_R} \tag{9}$$

$$Recall = \frac{O_W}{T_G} \tag{10}$$

F. METEOR

It also works in the same way that helps to compute the score of machine-generated text. In this case, uni-gram is matched between the actual caption and model generated caption. The score is calculated for many references, the highest score is selected as the answer. It also takes care of synonymy during the score evaluation.The METEOR score is calculated in equation (13).

$$10/Fmean = 1/Precision + 9/Recall \tag{11}$$

$$Penalty = 0.5 * Fragmentation\text{^}3 \tag{12}$$

$$Score = Fmean * (1 - Penalty) \tag{13}$$

Where *precision* is the ratio of correct words in generated sentences to the total words present in the generated sentence. *Recall* is the ratio of correct words in generated sentences to the total words present in the reference sentence. Fragmentation is the number of chunks in the hypothesis to the total number of word matches.

Table 1: Evaluation matrix

Evaluation Matrix	VGG16+Deep-LSTM+Attention
BLUE-4	0.412
METEOR	33.4
ROUGE	66.6

We can see the comparison of our model with sequence to sequence video description model in [20], factor graph model in [26], soft Attention model in [27],bidirectional-LSTM with Attention model in [28] and Hierarchical-LSTM with Attention model in [29] in Table 2 .Our model results are significantly better than other models.

Table 2: Evaluation matrix

Model Name	BLEU-4	METEOR
FGM [26]	-	29.3
Sequence-to-Sequence[14]	-	29.8
GoogleNet + 3D-CNN[27]	-	29.6
bi-LSTM[28]	0.37	29.8
hi-LSTM[29]	0.364	28.2
VGG16+Deep-LSTM+ Attention	0.412	33.4

G. Training Loss Plot

We run the model for 1000 epoch ,below figure represent the loss after 1000 epoch.

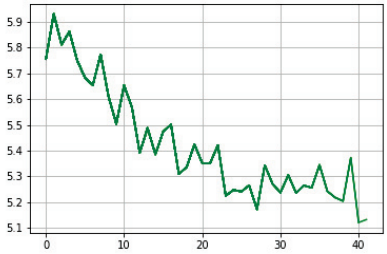

Figure 7: Loss Plot after 1000 epoch

We can observe that the loss value has reduced to 5.1 after running the model for 1000 epochs.

V. Conclusion

In the present work, we have shown a significant improvement wrt to other Encoder-Decoder techniques. CNN (VGG16) and Deep-LSTM is used as encoder and attention with LSTM layers is used as a decoder. In the first phase of the model, we are extracting the objects, actions using VGG16 later with the help of LSTM we generated the video descriptions. To show the effectiveness of the present work BLEU-4, ROUGE, METEOR scores are calculated which are 0.412, 66.6, and 33.4 respectively. We can enhance our results using bi-direction LSTM in encoding phase. Loss function optimisation is another improvement which can be added in the present work.

References

[1] Xiangpeng Li,Jingkuan Song ,and Heng Tao Shen "Hierarchical LSTMs with Adaptive Attention for Visual Captioning" in IEEE Transactions on Pattern Analysis and Machine Intelligence(Volume: 42, Issue: 5, May 1 2020).

[2] Ning Xu ,An-An Liu Yongkang Wong ,Yongdong Zhang "Dual-Stream Recurrent Neural Network for Video Captioning" in IEEE Transactions on Circuits and Systems for Video Technology (Volume: 29, Issue: 8, Aug. 2019).

[3] Bin Zhao, Xuelong Li , Xiaoqiang Lu ”CAM-RNN: Co-Attention Model Based RNN for Video Captioning” in IEEE Transactions on Image Processing (Volume: 28, Issue: 11, Nov. 2019).
[4] Chenggang Yan ;,Yunbin Tu , Xingzheng Wang ;,Yongbing Zhang ”STAT: Spatial-Temporal Attention Mechanism for Video Captioning” in IEEE Transactions on Multimedia (Volume: 22, Issue: 1, Jan. 2020).
[5] Zhiwang Zhang, Dong Xu,Wanli Ouyang,Chuanqi Tan ”Show, Tell and Summarize: Dense Video Captioning Using Visual Cue Aided Sentence Summarization” in IEEE Transactions on Circuits and Systems for Video Technology (Volume: 30, Issue: 9, Sept. 2020).
[6] Jan K. Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, Yoshua Bengio ”Attention-Based Models for Speech Recognition” Advances in Neural Information Processing Systems 28 (NIPS 2015).
[7] Sangavi G, Mrinalini K, Vijayalakshmi P ”Analysis on bilingual machine translation systems for English and Tamil” in 2016 International Conference on Computation of Power, Energy Information and Commuincation (ICCPEIC)
[8] Tingting He, Jinguang Chen, Liang Ma, Zhuoming Gui, Fang Li, Wei Shao, Qian Wang ”ROUGE-C: A fully automated evaluation method for multi-document summarization” in 2008 IEEE International Conference on Granular Computing.
[9] Krishna Subramanian, Dave Stallard,Rohit Prasad, Shirin Saleem, Prem Natarajan ”Semantic translation error rate for evaluating translation systems” in 2007 IEEE Workshop on Automatic Speech Recognition Understanding (ASRU)
[10] Alex Sherstinsky ”Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network” in Elsevier ”Physica D: Nonlinear Phenomena” journal, Volume 404, March 2020: Special Issue on Machine Learning and Dynamical Systems.
[11] N. Krishnamoorthy, G. Malkarnenkar, R. J. Mooney, K. Saenko, and S. Guadarrama, “Generating natural-language video descriptions using text-mined knowledge,” in Proc. AAAI Conf. Artif. Intell., 2013, pp. 541–547.
[12] J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko, and R. Mooney, “Integrating language and vision to generate natural language descriptions of videos in the wild,” in Proc. Int. Conf. Comput. Linguist., 2014, pp. 1218–1227
[13] R. Xu, C. Xiong, W. Chen, and J. J. Corso, “Jointly modeling deep video and compositional text to bridge vision and language in a unified framework,” in Proc. AAAI Conf. Artif. Intell., 2015, pp. 2346–2352.
[14] S. Venugopalan et al., “Sequence to sequence—Video to text,” in Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 4534–4542.
[15] L. Yao et al., “Describing videos by exploiting temporal structure,” in Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 4507–4515.
[16] Y. Bin, Y. Yang, F. Shen, X. Xu, and H. T. Shen, “Bidirectional long-short term memory for video description,” in Proc. ACM Int. Conf. Multimedia, 2016, pp. 436–440.
[17] L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen, “Video captioning with attention-based LSTM and semantic consistency” in IEEE Trans. Multimedia, vol. 19, no. 9, pp. 2045–2055, Sep. 2017.
[18] N. Krishnamoorthy, G. Malkarnenkar, R. J. Mooney, K. Saenko, and S. Guadarrama, “Generating natural-language video descriptions using text-mined knowledge,” in Proc. AAAI Conf. Artif. Intell., 2013, pp. 541–547.
[19] J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko, and R. Mooney, “Integrating language and vision to generate natural language descriptions of videos in the wild,” in Proc. Int. Conf. Comput. Linguist., 2014, pp. 1218–1227.
[20] S. Venugopalan et al., “Sequence to sequence—Video to text,” in Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 4534–4542.
[21] L. Yao et al., “Describing videos by exploiting temporal structure,” in Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 4507–4515.
[22] P. Pan, Z. Xu, Y. Yang, F. Wu, and Y. Zhuang, “Hierarchical recurrent neural encoder for video representation with application to captioning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, 2016, pp. 1029–1038.
[23] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui, “Jointly modeling embedding and translation to bridge video and language,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, 2016, pp. 4594–4602.
[24] Hochreiter, S. Schmidhuber, J. (1997) ”Long short-term memory” in Neural Computation, 9 (8), 1735-1780.
[25] Pengfei Liu ,Xipeng Qiu,Xuanjing Huan ”Recurrent Neural Network for Text Classification with Multi-Task Learning” Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)
[26] J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko, and R. J. Mooney. ”Integrating language and vision to generate natural language descriptions of videos in the wild” Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics s: Technical Papers,pages 1218–1227, Dublin, Ireland, August 23-29 2014.
[27] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville ”Describing videos by exploiting temporal structure” arXiv:1502.08029v4, 2015.
[28] Y. Bin, Y. Yang, F. Shen, N. Xie, H. T. Shen, X. Li, “Describing Video with Attention-Based Bidirectional LSTM”, accepted in IEEE Transactions on Cybernetics, Bilstm wala
[29] L. Gao, X. Li, J. Song and H. T. Shen, “Hierarchical LSTMs with Adaptive Attention for Visual Captioning”, accepted in IEEE Journal of Latex Class Files, Vol. 14, No. 8, August 2015 Ahlstm wala.
[30] Geetha V., Salvi S., Saini G., Yadav N., Singh Tomar R.P. (2021) ”Follow Me: A Human Following Robot Using Wi-Fi Received Signal Strength Indicator” in Tuba M., Akashe S., Joshi A. (eds) ICT Systems and Sustainability. Advances in Intelligent Systems and Computing, vol 1270. Springer, SingaporeICT Systems and Sustainability. Advances in Intelligent Systems and Computing, vol 1270. Springer, Singapore.