[1]Vasudha Tiwari *

[2]Charul Bhatnagar

Attention-Based Multi-Layered Encoder-Decoder Model for Summarizing Non-Interactive User-Based Videos

Abstract: - Video summarization extracts the relevant contents from a video and presents the entire content of the video in a compact and summarized form. User based video summarization, can summarize a video as per the requirement of the user. In this work, a non interactive and a perception-based video summarization technique is proposed that makes use of attention mechanism to capture user's interest and extract relevant keyshots in temporal sequence from the video content. Here, video summarization has been articulated as a sequence-to-sequence learning problem and a supervised method has been proposed for summarization of the video. Adding layers to the existing network makes it deeper, enables higher level of abstraction and facilitates better feature extraction. Therefore, the proposed model uses a multi-layered, deep summarization encoder-decoder network (MLAVS), with attention mechanism to select final keyshots from the video. The contextual information of the video frames is encoded using a multi-layered Bidirectional Long Short-Term Memory network (BiLSTM) as the encoder. To decode, a multi-layered attention-based Long Short-Term memory (LSTM) using a multiplicative score function is employed. The experiments are performed on the benchmark TVSum dataset and the results obtained are compared with recent works. The results show considerable improvement and clearly demonstrate the efficacy of this methodology against most of the other available state-of-art methods.

Keywords: Multi-layered encoder-decoder, video summarization, attention, BiLSTM, LSTM

I. INTRODUCTION

Recently, a tremendous rise in the availability of multimedia content can be seen on the internet. The ease of capturing and uploading of videos on the social networking sites and internet is one of the major contributors. The rapid growth of this content has made browsing, retrieval, indexing, processing, storage and sharing of this content a tough task.

Therefore, we need a mechanism that can summarise the content in a form such that it contains the most relevant and important part of the content and thus ease the above-mentioned tasks. Video summarization can serve the purpose as it aims to produce summaries of videos that include the most important data of a video in compact form [1]. The video summaries generated can be either a static summary or a dynamic summary. The static summary comprises of a collection of keyframes and the dynamic summary also known as a video skim is mainly a collection of keyshots that are chosen on the basis of the importance scores that are assigned to the frames or shots respectively. Static summary is mainly in the form of still images, thumbnails or storyboards whereas a dynamic summary mainly consists of video segments in temporal sequence. Importance scores help in identifying the keyframes or keyshots as per the relevance of the application or requirement of the user. However, it must be noted that generation of the importance scores is not a trivial task. Finding out what is important from the entire content is highly subjective as it can vary for different persons depending upon various factors such as requirement of the user or application, genre of the video etc. During the last decade, many researchers have worked in the field of video summarization [1-5] and proposed various summarization techniques based on genre, applications, user preferences etc. However, it has been observed that user-based video summarization i.e., generation of a video summary as per the demand of the user is the need of the hour. User based video summarization can be further categorised as interactive and non-interactive or perception-based video summarization. In interactive video summarization the user provides an extra input along with the input video such as a query [6-9] for summarization. The query can be expressed as textual description of some object or event, image, video segments or keywords and the summary is generated based on the query. In non-interactive or perception-based video summarization, instead of using an additional input such as a query, importance score generation is based on cues such as attention, emotions, facial expressions etc. [10-13]. Video summarization techniques follow either an unsupervised or a supervised approach. Most of the previous

[1] Department of CEA, GLA University, Mathura, India

vasudhatiwari1608@gmail.com

[2] Department of CEA, GLA University, Mathura, India

charul@gla.ac.in

studies focus on unsupervised approach, but it is seen that the recent studies have shifted their focus on exploring supervised approaches. In a supervised approach, the problem of video summarization is formulated as a sequence-to-sequence learning problem, wherein a set of frames of a video are taken as input in sequence and the predicted importance scores for these frames are the output. On the basis of these scores the keyframes or keyshots are further selected to generate the output summary. In most of these approaches, Recurrent Neural Network (RNN), commonly a LSTM is used for modelling the sequential data. However, it has been noticed that LSTM has several drawbacks, it is not a suitable solution for long video streams as it is capable of handling only a frame length of 30-80 frames [14]. Another, drawback with LSTM is that it is not able to distinguish among frames effectively as each frame is assigned an equal weight. To address these issues, we propose a deep framework, a multi-layered attention based, encoder - decoder network for video summarization where we use multi-layered BiLSTM as the encoder and multi-layered LSTM as decoder. The attention mechanism helps in assigning proper weights to each frame and thus leads to efficient selection of keyshots for final dynamic summary generation. The following are some notable contributions made by this work:

1. Proposed is a deep summarization model that includes a multi-layered encoder-decoder framework.
2. Utilizing attention, diverse weights are assigned to video frames, enabling the generation of a highly relevant and semantically representative summary while concurrently capturing long-term dependencies within the content.
3. The experiments were conducted on the widely used benchmark TVSum dataset, and it is evident that our model outperforms the majority of state-of-the-art methods.

The paper is further organised as mentioned: Section 2 lists the related work. Section 3 discusses the proposed framework and finally Section 4 discusses the Experiment and Results.

II. RELATED WORK

Video Summarization techniques are broadly classified as unsupervised technique and supervised technique. Unsupervised approaches use heuristic criteria for ranking and selecting keyframes or keyshots. Most of the unsupervised techniques use clustering-based approaches and sparse coding for summarization [15]. Clustering based methods select the cluster centres for creation of the summary of the video. Various clustering techniques such as k-means [16], graph clustering [17], prototype selection etc are mostly used for summarization process. Sparse coding techniques consider the video summarization problem as a minimum sparse reconstruction problem [18]. Reinforcement learning is proposed in [19] for video summarization which uses a label and free reward function that jointly considers both diversity and representativeness for generating a video summary. [20] proposes the use of Generative Adversarial Network (GAN) framework for video summarization. It comprises of a summarizer and a discriminator. An autoencoder, a LSTM is used for the summarizer part and again an LSTM is used for the discriminator part.

Recently, the supervised techniques are being explored by the researchers and gaining more attention. The supervised approaches use human labelled annotations for selecting the keyframes or keyshots for summary generation. [21] proposes Sequential Determinantal Point Process (seqDPP) for video summarization and considers the video summarization problem as a supervised subset selection problem. [22] proposes a hierarchical neural network for summarization process that has two layers. One layer is used to encode the short subshots and the final hidden state of each subshot of the original video and the second layer calculates the importance score. Attention based models [5,23,10,24,25][33-39] are capable of producing better results as they are able to focus on more relevant things and are therefore capable of producing better summary. [23] proposes a fully convolutional sequence model and establishes a relation between semantic segmentation and video summarization.[5][40-48] uses an encoder-decoder framework where encoder uses a BiLSTM for encoding of contextual information and two attention-based LSTM for decoding purpose. [10] proposes a deep and a hierarchical LSTM network with attention mechanism for video summarization that uses a 3D-CNN for extraction of spatial -temporal features. Inspired by the use of supervised techniques and performance of the models based on attention, in this paper, we suggest a multi-layered encoder-decoder framework with attention mechanism.

III. PROPOSED FRAMEWORK

3.1 Problem Formulation

Video summarization is considered as a problem in which a video is given as input and output is either a static summary consisting of a set of keyframes in a sequence or a dynamic summary consisting of keyshots in a sequence. Previous works represent both the forms of output, either as binary labels or as frame level importance scores which help in selection of keyframes for a static summary or keyshots for a dynamic summary [23]. Most of the pre-existing datasets also have the ground truth annotations in either of the two forms. We have formulated the video summarization as a sequence-to-sequence problem where the video is taken as input and the sequence of keyshots are generated as output, considering the binary labelled annotations. A video V with n frames is considered for summarization. The frames of the video are pre- processed and their feature vector representations are taken as input. The output sequence consists of frame level scores that are converted into shot level scores and the final summary (sequence of keyshots) is generated by selecting the keyshots in temporal sequence and with a length budget. The proposed model consists of an attention-based multi layered encoder -decoder framework followed by a summary generation or keyshot selection model for selection of keyshots as the final summary. Fig 1 shows the framework of the suggested model.

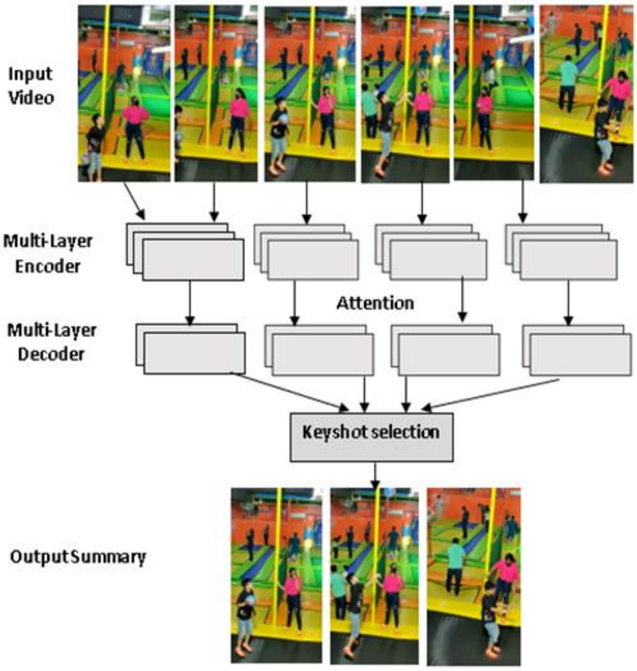


Figure 1 Proposed Framework of ML-AVS network

The encoder decoder model has proved to be a powerful solution for sequence to sequence-based prediction problems. The encoder extracts the features from the input and generates a context vector. The context vector contains all the information of the input data and is further send as input to the decoder for final predictions as output. It is often observed that with large data these network models do not work well as the generated context vector of fixed length is not capable of remembering the entire input sequence. Therefore, a mechanism is required that can help the neural network to remember the long sequence and at the same time mimic the human brain by selectively focussing on the more relevant things while ignoring the less important ones. Attention mechanism can be utilised to implement this. In addition to this, stacking multiple layers in the network model can produce better results, as adding layers to the existing network makes it deeper. The deeper models provide a higher level of abstraction and predict more accurately. The additional layers are able to recombine the learned representations from the previous layers and create new representations having higher level of abstraction. Each layer of the network processes some part of the input and sends it over to the next layer, until the final layer produces the output. Here, we have used attention and multi-layers in the encoder-decoder model for generating the output summary. Fig. 2 shows the flowchart of the proposed Multi-Layered Attention based Video Summarization (ML-AVS) model.

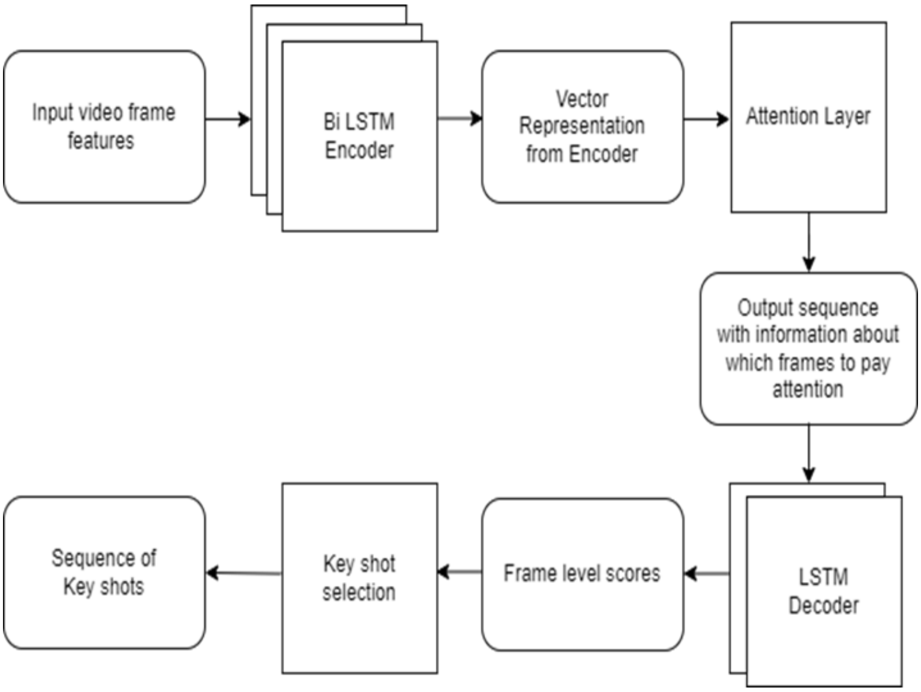

Input video frame features
Bi LSTM Encoder
Vector Representation from Encoder
Attention Layer
Output sequence with information about which frames to pay attention
LSTM Decoder
Frame level scores
Key shot selection
Sequence of Key shots

Figure 2 Flowchart of the proposed ML-AVS model

3.2 Multi-Layered BiLSTM Encoder

The encoder, in the encoder decoder model converts the input sequence F = {f_1, f_2, f_3, …, f_t} into a representation vector v = {v_1, v_2, v_3 …., v_t}. Here, the frame features of a video input are represented through f_1, f_2, f_3, ….., f_t. The existing state-of-art methods mostly use a variant of RNN called LSTM for the encoder. LSTM has gained ample popularity as it is able to encapsulate the long-term temporal dependency and thus is very suitable for video data, but at the same time has its own limitations [14] as mentioned earlier. BiLSTM network has shown significant improvement in performance over LSTM in encoding the relevant information of the sequential data. It is also capable of considering the bidirectional long-term structural dependencies between the frames of the video. Therefore, here we have chosen BiLSTM as the encoder. As shown in Fig. 3, BiLSTM uses two independent LSTM, where the first LSTM computes data in the forward direction (forward states) and the second LSTM computes data in the backward direction (backward states).

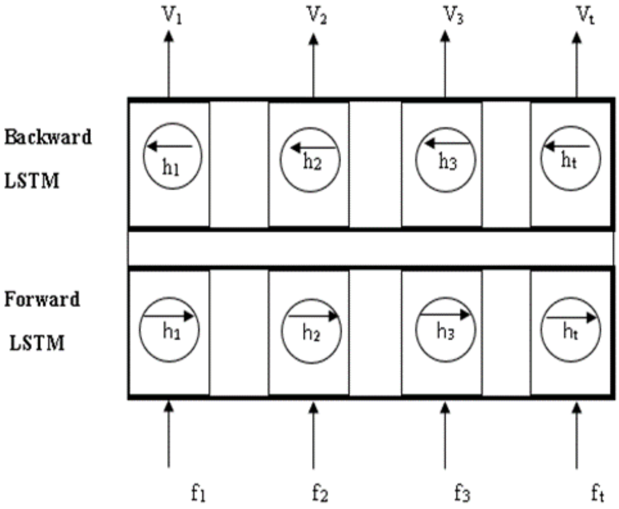


Figure 3 BiLSTM Encoder

In forward LSTM, the inputs are f_{t-1}, f_t, f_{t+1} and the outputs are $\vec{h}_{t-1}, \vec{h}_t, \vec{h}_{t+1}$, in backward LSTM, the inputs are f_{t+1}, f_t, f_{t-1} and the outputs are $\overleftarrow{h}_{t+1}, \overleftarrow{h}_t, \overleftarrow{h}_{t-1}$. Therefore, BiLSTM is able to consider the bidirectional long-term structural dependencies between the frames of the video. We have stacked two additional layers to the BiLSTM network for efficient representation vector generation and obtain v_t for each f_t.

The BiLSTM calculates the input sequence,

$f = [f_1, f_2, f_3 \dots \dots \dots, f_n]$ from a forward hidden sequence $h_{tF} = [h_{1F}, h_{2F}, \dots, h_{nF}]$ and a backward hidden sequence $h_{tB} = [h_{1B}, h_{2B}, \dots \dots, h_{nB}]$.

The encoder v_t can be obtained by concatenating the final forward and backward output. as follows:

$$v_t = [h_{tF}, h_{tB}] \tag{1}$$

$$h_{tF} = \sigma(W_{hFf} f^t + W_{hFhF} h_{(t-1)F} + b_{hF} \tag{2}$$

$$h_{tB} = \sigma(W_{hFf} f^t + W_{hFhF} h_{(t+1)B} + b_{hF} \tag{3}$$

$$v_t = W_{vhF} h_{tF} + W_{vhF} h_{tB} + b_v \tag{4}$$

The architecture of encoder θ depends upon the input of the application, and thus can be represented as follows:

$$\begin{bmatrix} v_t \\ h_t \end{bmatrix} = \theta(f_t) \tag{5}$$

3.3 Multi layered Decoder with attention mechanism

The decoder used here is multi-layered LSTM. It has an additional layer stacked over the LSTM network that generates an output sequence y , where

y= {y_1, y_2, y_3, …, y_n} for the representation vector received from the encoder. Fig. 4 shows the decoder with attention mechanism.

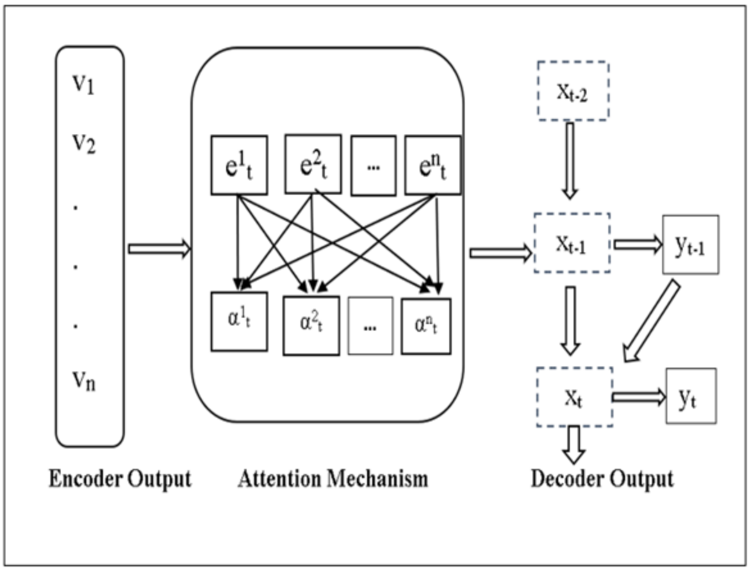


Figure. 4 Decoder with attention mechanism

As already mentioned, the encoder is a bidirectional LSTM which has a forward and a backward hidden state. The final encoder state can thus be obtained by the concatenation of these two states.

Now, the decoder hidden state can be obtained as follows:

$$x = f(x_{t-1}, y_{t-1}, v) \tag{6}$$

The, LSTM decoder β, therefore can be represented as:

$$\begin{bmatrix} p(y_t|\{y_t|i<t\},v) \\ x_t \end{bmatrix} = \beta(x_{t-1}, y_{t-1}, v) \qquad (7)$$

When a huge dataset is provided to the model to learn, there is a possibility that few significant parts of the data may get ignored by the model. Paying attention to relevant part in the data can increase the performance of the model. An additional attention mechanism, therefore if added in the model can help us achieve this. Neural network architectures having different layers and can easily incorporate the attention mechanism through one of its layers. In a video data the importance scores of the frames in a shot are generally continuous but varies for various shots, therefore the decoder has to learn both the long-term dependency and the short-term dependency of these frames. The attention mechanism provides an additional attention weight to the inputs and enables the decoder to focus selectively on the input. The attention mechanism therefore, changes the representation vector v to $V_{t,}$ where V_t is the attention vector at time t.

The LSTM decoder β after implementation of attention can be finally represented as:

$$\begin{bmatrix} p(y_t|\{y_t|i<t\},V_t) \\ x_t \end{bmatrix} = \beta(x_{t-1}, y_{t-1}, V_t) \qquad (8)$$

where V_t can be calculated as:

$$V_t = \sum_{i=1}^{n} \alpha_t^i \, v_t \qquad (9)$$

α_t^i is the attention weight that is calculated at time t. This represents the amount of attention of the i^{th} feature of the input video. To compute this, relevance score e_t^i needs to be computed. This is a combination of x_{t-1} the previous hidden state of the decoder and v_t the output from the encoder at time t. It can be represented as:

$$e_t^i = \mathrm{score}(x_{t-1}, v_t) \qquad (10)$$

This relevance score can be computed either by using an additive score function or a multiplicative score function. We have used the multiplicative score function and thus the relevance score is calculated as follows:

$$e_t^i = v_i^T W_a x_{t-1} \qquad (11)$$

After the relevance scores have been computed for all the frames it is normalised to obtain α_t^i.

$$\alpha_t^i = \exp(e_t^i) / \sum_{j=1}^{n} \exp(e_t^j) \qquad (12)$$

3.4 Summary Generation (Keyshots selection)

After the frame level scores are obtained from the model, it needs to be converted into shot level scores so that the shots maybe further selected for the final summary. Based on the change points from the dataset used, the visually similar frames are converted into shots. The average of the predicted frame level scores of a particular shot is taken to compute the shot level score. Thereafter to obtain the final summary the optimization problem needs to be solved where we need to select the shots having highest weight and in temporal sequence. This exactly becomes similar to a 0/1 knapsack problem and is solved by dynamic programming. The shots are then combined together to create the final summary.

IV. EXPERIMENTS AND RESULTS

4.1 Implementation

The input to the multi-layered BiLSTM encoder is a frame feature vector of 1024 dimension. During training, the video is down sampled to 320 frames, therefore a total input of 320 x 1024 is given to the encoder. Two additional layers of BiLSTM are stacked to the BiLSTM encoder. Each layer of a neural network has a specific number of nodes or neurons which decide how many variables should be there in that layer. Starting from 128 at BiLSTM layer, we proceeded with increasing and decreasing the number of nodes at each layer and found out that increasing number of nodes did not help in getting better scores. The optimal starting point was found to be 64 nodes. Learning Rate of Adam optimizer was set to 0.001. A total of 10 epochs were considered. The output of the encoder is combined with the attention weights produced by the attention layer and sent as input to the multi-layered LSTM decoder. One additional LSTM layer is stacked to the LSTM decoder. The decoder makes the frame level score predictions which is further sent to the keyshot selection part. These frame level scores are then converted to shot level scores and the keyshots are selected for final summary by solving the optimisation problem that is exactly similar to 0/1 knapsack problem. The shots are then combined to create the final summary which is less than or equal to 20% length of the entire length of the input video.

4.2 Dataset

The proposed framework is trained and evaluated on a pre-processed TVSum dataset [23]. The TVSum dataset [26] is a publicly available benchmark dataset that consists of total 50 videos. These videos are downloaded from YouTube in 10 different categories. The detailed description of TVSum dataset is given in Table 1. The pre-processed TVSum dataset used here has total 50 groups for video1 to video 50. The dataset consists of the following details, the length of each video, feature vector, label, change points (start and end of each segment), number of frames in each segment and user summary from 20 users represented as binary vector. The detailed description of the pre-processed dataset is given in Table 2.

Table 1 Description of the TV Sum Dataset

Name	Description
Video id	Total of 50 videos
Video Category	Total 10 categories (videos are grouped in 5 videos per category). The categories are specified as two letter codes as follows: VT: Changing Vehicle Tire VU: Getting Vehicle Unstuck GA: Grooming and Animal MS: Making Sandwich PK: Parlour PR: Parade FM: Flash Mob Gathering BK: Bee Keeping BT: Attempting Bike Tricks DS: Dog Show
Video Title	Total of 50 videos
url	Has the url of the 50 videos that were downloaded from YouTube in 2014
Length	Time duration of the 50 videos in (mm:ss) format
Shot level importance score	Has the annotations made by 20 users, from 1 (low) to 5 (high). Each shot has been annotated by 20 different users so all the frames in a particular shot have the same importance score.

Table 2 Description of the Dataset

Name	Description
Length	Number of frames
Feature	Shape (320,1024)
Label	Shape (320,)
Change_points	Begin and end of each segment
N_frame_per_seg	Number of frames in each segment
User_summary	Summary of 20 users, each row is a binary vector

4.3 Evaluation Metric

For assessment and further comparison with other state of art methods, the generated summaries are contrasted with the human generated summaries. The similarity of the summaries is measured through the three popular metrics, F1 score, precision and recall. These metrics are utilised for objective or quantitative analysis of the technique. For a given video V, if generated video summary is Vs and ground truth summary is Vg the precision, recall and F1 score are calculated as follows:

$$\mathrm{P} = \frac{\mathrm{V_s} \cap \mathrm{V_g}}{\mathrm{V_s}}, \quad \mathrm{R} = \frac{\mathrm{V_s} \cap \mathrm{V_g}}{\mathrm{V_g}} \tag{13}$$

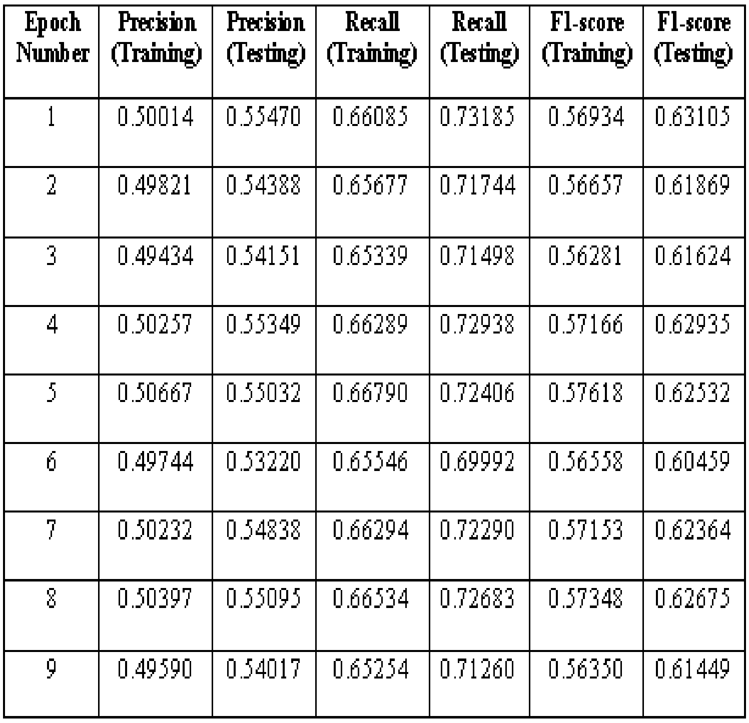
$$\mathrm{F1} = \frac{2\mathrm{x}\,(\mathrm{P+R})}{(\mathrm{P+R})}\ \mathrm{x}\ 100 \tag{14}$$

where P and R represent Precision and Recall and F1 represents the F1 score.

4.4 Implementation Results

The similarity of the summary processed by the suggested model and the human annotated summary is measured by the F1 score. Table 3 lists the values of precision, recall and F1 scores computed for 10 epochs during the training and testing process.

Table 3 Performance metrics of the suggested model



Epoch Number	Precision (Training)	Precision (Testing)	Recall (Training)	Recall (Testing)	F1-score (Training)	F1-score (Testing)
1	0.50014	0.55470	0.66085	0.73185	0.56934	0.63105
2	0.49821	0.54388	0.65677	0.71744	0.56657	0.61869
3	0.49434	0.54151	0.65339	0.71498	0.56281	0.61624
4	0.50257	0.55349	0.66289	0.72938	0.57166	0.62935
5	0.50667	0.55032	0.66790	0.72406	0.57618	0.62532
6	0.49744	0.53220	0.65546	0.69992	0.56558	0.60459
7	0.50232	0.54838	0.66294	0.72290	0.57153	0.62364
8	0.50397	0.55095	0.66534	0.72683	0.57348	0.62675
9	0.49590	0.54017	0.65254	0.71260	0.56350	0.61449

As discussed, to obtain the final summary the shots having highest weight and in temporal sequence are selected. The final summary is created by joining those shots and it is less than or equal to 20% of the entire length of the original video. Table 4 shows the results of 10 videos randomly chosen from the dataset and lists the details of the original video and the generated summary video.

Table 4 Description of Results obtained

Original Video				Summary Video		
Video id	Original Length	Total frames	Total shots	Summary Length	Total frames	Total shots

2	2.36 mins	4687	32	0.38 sec	923	10
5	1.51 mins	3326	23	0.26 sec	635	10
14	3.14 mins	4852	33	0.39 sec	939	12
25	4.34 mins	6579	44	0.54 sec	1301	15
30	2.47 mins	4004	27	0.32 sec	779	10
35	2.29 mins	4462	30	0.36 sec	877	12
42	3.18 mins	5938	31	0.48 sec	1172	15
45	1.44 mins	2499	17	0.20 sec	482	5
47	3.09 mins	4739	32	0.38 sec	934	10
50	3.50 mins	6911	31	0.56 sec	1363	19

To do a fair comparison, Table 5 lists the F1 scores of some state-of-art methods that have performed and evaluated their proposed summarization technique on similar dataset. The obtained results clearly show that our ML-AVS model shows improvement on TVSum dataset and outperforms the listed state-of-art methods

Table 5 Results (F1 scores) of various state-of-art methods on TVSum Dataset

Method	**Unsupervised**	**Supervised**	**F1 Score**
ML-AVS (Ours)		✓	61.9
M-AVS [5]		✓	61.0
DHAVS [10]		✓	60.8
AC-SUM-GAN [32]	✓		60.6
HSA-RNN [27]		✓	59.8
A-AVS [5]		✓	59.4
CSNet [28]	✓		58.8
CSNet [28]		✓	58.5
DySeqDPP [29]		✓	58.4
DR-DSN [19]	✓		57.6
SUM-GAN sup [20]		✓	56.3
dppLSTM [30]		✓	54.7
vsLSTM [30]		✓	54.2
Li et al [31]		✓	52.7
SUM-GAN dpp [20]	✓		51.7

Fig. 5 shows the graphical representation of the comparison of the F1 scores of various state-of-art methods and Fig. 6 depicts the average performance of the model.

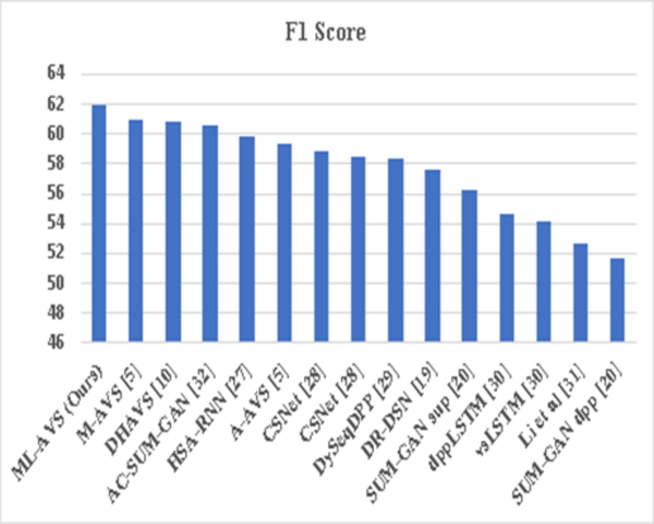


Figure.5 Performance comparison of proposed model ML-AVS and other state-of-art methods

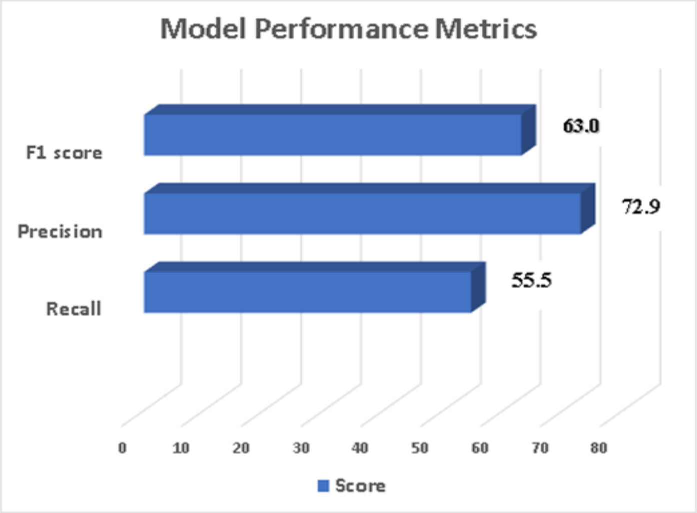


Figure 6 Average Performance of the model

V. CONCLUSION

The model follows a supervised approach for generation of video summaries. A multiplicative attention mechanism is employed for assigning weights to the encoder input, so that the decoder may be able to make the predictions according to the attention weights and therefore assign proper importance scores to the frames. The frame scores are converted into shot level scores and then optimization is achieved by selectively picking up and joining the shots selectively. Addition of multiple layers to the neural network makes it deeper and is able to extract deeper features. The suggested model is able to successfully produces summaries of the original video that are less than or equal to 20% length of the original video and a F1 score of 61.9. The ML-AVS model is compared with recent state of art methods for the same dataset and the obtained results show significant improvement. Stacking of layers on the encoder and decoder model proves to be an efficient technique and a comparatively better approach in comparison to several other state-of-art methods. However, the limited training set proved to be a limitation and can be improved further. As a future work, the technique can be applied for different and more specific genre videos and evaluated for different datasets.

Compliance and Ethical Standards

Conflict of Interest The authors give the declaration that this manuscript has no conflict of interest with any other published work.

Human and Animal Rights The authors declare that no human or animal subject was used in this work.

Informed Consent The authors give declaration that an informed consent was received from the individuals involved if any in the study.

Declaration

Funding The authors give the declaration that no funds or grants has been taken from any source for this work.

Author contributions

Vasudha Tiwari: Conceptualisation, Research work, Experiments, Validation, Manuscript Writing, Figures.

Charul Bhatnagar: Conceptualisation, Guidance in Research work, Review and Editing.

Data Availability The used or analysed datasets in this work are publicly available and have been cited for access and further use.

REFERENCES

[1] Tiwari V, Bhatnagar C (2021) A survey of recent work on video summarization: approaches and techniques. Multimedia Tools and Applications 80, no. 18: 27187-27221.

[2] Basavarajaiah M, Sharma P (2019) Survey of compressed domain video summarization techniques. ACM Computing Surveys (CSUR), 52(6), 1-29.

[3] Sreeja M U, Kovoor B C (2019) Towards genre-specific frameworks for video summarisation: A survey. Journal of Visual Communication and Image Representation, 62, 340-358.

[4] Del Molino A G, Tan C, Lim J H, Tan A H (2016) Summarization of egocentric videos: A comprehensive survey. IEEE Transactions on Human-Machine Systems, 47(1), 65-76.

[5] Ji Z, Xiong K, PangY, & Li X (2019) Video summarization with attention-based encoder–decoder networks. IEEE Transactions on Circuits and Systems for Video Technology, 30(6), 1709-1717.

[6] Sharghi A, Gong B, & Shah M (2016) Query-focused extractive video summarization. In European conference on computer vision (pp. 3-19). Springer, Cham.

[7] Sharghi A, Laurel J S, Gong B (2017) Query-focused video summarization: Dataset, evaluation, and a memory network based approach. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4788-4797).

[8] Vasudevan A B, Gygli M, Volokitin A, Van Gool L (2017) Query-adaptive video summarization via quality-aware relevance estimation. In Proceedings of the 25th ACM international conference on Multimedia (pp. 582-590).

[9] Zhang Y, Kampffmeyer M, Zhao X, Tan M (2019) Deep reinforcement learning for query-conditioned video summarization. Applied Sciences, 9(4), 750.

[10] Lin J, Zhong S H, Fares A (2022) Deep hierarchical LSTM networks with attention for video summarization. Computers & Electrical Engineering, 97, 107618.

[11] Kanehira A, Van Gool L, Ushiku Y, Harada T (2018) Viewpoint-Aware video summarization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 7435-7444).

[12] Joho H, Jose J M, Valenti R, Sebe N (2009) Exploiting facial expressions for affective video summarisation In Proceedings of the ACM international conference on image and video retrieval (pp. 1-8).

[13] Peng W T, Chu W T, Chang C H, Chou, C N, Huang W J, Chang W Y, Hung, Y P (2011) Editing by viewing: automatic home video summarization by viewing behavior analysis. IEEE Transactions on Multimedia, 13(3), 539-550.

[14] Zhong S H, Lin J, Lu J, Fares A, Ren T (2022) Deep semantic and attentive network for unsupervised video summarization. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 18(2), 1-21.

[15] Ji Z, Zhao Y, Pang Y, Li X, Han J (2020). Deep attentive video summarization with distribution consistency learning. IEEE transactions on neural networks and learning systems, 32(4), 1765-1775.

[16] De Avila S E F, Lopes A P B, da Luz Jr A, de Albuquerque Araújo A (2011) VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method. Pattern recognition letters, 32(1), 56-68.

[17] Chu W S, Song Y, Jaimes A (2015) Video co-summarization: Video summarization by visual co-occurrence. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3584-3592).

[18] Cong Y, Yuan J, Luo J (2011) Towards scalable summarization of consumer videos via sparse dictionary selection. IEEE Transactions on Multimedia, 14(1), 66-75.

[19] Zhou K, Qiao Y, Xiang T (2018) Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 32, No. 1)

[20] Mahasseni B, Lam M, Todorovic S (2017) Unsupervised video summarization with adversarial lstm networks. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (pp. 202-211).

[21] Gong B, Chao W L, Grauman K, Sha F (2014) Diverse sequential subset selection for supervised video summarization. Advances in neural information processing systems, 27.

[22] Zhao B, Li X, Lu X (2017) Hierarchical recurrent neural network for video summarization. In Proceedings of the 25th ACM international conference on Multimedia (pp. 863-871).

[23] Rochan M, Ye L, Wang Y (2018) Video summarization using fully convolutional sequence networks. In Proceedings of the European conference on computer vision (ECCV) (pp. 347-363).

[24] Lee H, Liu M, Riaz H, Rajasekaren N, Scriney M, Smeaton A F (2021) Attention based video summaries of live online zoom classes. arXiv preprint arXiv:2101.06328.

[25] Sanabria M, Precioso F, Menguy T (2021) Hierarchical multimodal attention for deep video summarization. In 2020 25th International Conference on Pattern Recognition (ICPR) (pp. 7977-7984). IEEE.

[26] Song Y, Vallmitjana J, Stent A, Jaimes A (2015) Tvsum: Summarizing web videos using titles. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 5179-5187).

[27] Zhao B, Li X, Lu X (2018) Hsa-rnn: Hierarchical structure-adaptive rnn for video summarization. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7405-7414).

[28] Jung Y, Cho D, Kim D, Woo S, Kweon I S (2019) Discriminative feature learning for unsupervised video summarization. In Proceedings of the AAAI Conference on artificial intelligence (Vol. 33, No. 01, pp. 8537-8544).

[29] Li Y, Wang L, Yang T, Gong B (2018) How local is the local diversity? reinforcing sequential determinantal point processes with dynamic ground sets for supervised video summarization. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 151-167).

[30] Zhang K, Chao W L, Sha F, Grauman K (2016) Video summarization with long short-term memory. In European conference on computer vision (pp. 766-782). Springer, Cham.

[31] Li X, Zhao B, Lu X (2017) A general framework for edited video and raw video summarization. IEEE Transactions on Image Processing, 26(8), 3652-3664.

[32] Apostolidis E, Adamantidou E, Metsai A I, Mezaris V, Patras I (2020) AC-SUM-GAN: Connecting actor-critic and generative adversarial networks for unsupervised video summarization. IEEE Transactions on Circuits and Systems for Video Technology, 31(8), 3278-3292.

[33] Agarwal, Ambuj Kumar, Rupesh Kumar Jindal, Deepak Chaudhary, Raj Gaurang Tiwari, and Megha Sharma. "Security and Privacy Concerns in the Internet of Things: A Comprehensive Review." In 2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART), pp. 254-259. IEEE, 2022.

[34] Tiwari, Raj Gaurang, Ambuj Kumar Agarwal, Rajesh Kumar Kaushal, and Naveen Kumar. "Prophetic analysis of bitcoin price using machine learning approaches." In 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), pp. 428-432. IEEE, 2021.

[35] Tiwari, Raj Gaurang, Sandeep Kumar, Gaurav Vishnu Londhe, Ambuj Kumar Agarwal, and Rajat Bhardwaj. "Accurate and Automated Deep Learning Solution for Skin Cancer Detection." International Journal of Intelligent Systems and Applications in Engineering 11, no. 5s (2023): 490-500.

[36] Tiwari, Raj Gaurang, Ambuj Kumar Agarwal, Nishant Gupta, Aman Anand, and Nikita Verma. "Conceptualization of Effective Algorithm for Minimizing Power Consumption in Cloud Servers." In 2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART), pp. 445-449. IEEE, 2022.

[37] Agarwal, Ambuj Kumar, Lekha Rani, Raj Gaurang Tiwari, Tarun Sharma, and Pradeepta Kumar Sarangi. "Honey encryption: fortification beyond the brute-force impediment." In Advances in Mechanical Engineering: Select Proceedings of CAMSE 2020, pp. 673-681. Springer Singapore, 2021.

[38] Agarwal, Ambuj Kumar, Raj Gaurang Tiwari, Rajesh Kumar Kaushal, and Naveen Kumar. "A systematic analysis of applications of blockchain in healthcare." In 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), pp. 413-417. IEEE, 2021.

[39] Agarwal, Ambuj Kumar, Vidhu Kiran, Rupesh Kumar Jindal, Deepak Chaudhary, and Raj Gaurang Tiwari. "Optimized Transfer Learning for Dog Breed Classification." International Journal of Intelligent Systems and Applications in Engineering 10, no. 1s (2022): 18-22.

[40] Tiwari, Raj Gaurang, Ambuj Kumar Agarwal, Rupesh Kumar Jindal, and Anshbir Singh. "Experimental Evaluation of Boosting Algorithms for Fuel Flame Extinguishment with Acoustic Wave." In 2022 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), pp. 413-418. IEEE, 2022.

[41] Tiwari, Raj Gaurang, Pratibha, Sandeep Dubey, and Ambuj Kumar Agarwal. "Impact of IDMA Scheme on Power Line Communication." In Recent Trends in Product Design and Intelligent Manufacturing Systems: Select Proceedings of IPDIMS 2021, pp. 985-996. Singapore: Springer Nature Singapore, 2022.

[42] De, Indrajit, Lekha Rani, Rajat Bhardwaj, Ambuj Kumar Agarwal, and Raj Gaurang Tiwari. "Human Posture Recognition by Distribution-Aware Coordinate Representation and Machine Learning." International Journal of Intelligent Systems and Applications in Engineering 11, no. 5s (2023): 477-489.

[43] Trivedi, Naresh Kumar, Raj Gaurang Tiwari, Ambuj Kumar Agarwal, and Vinay Gautam. "A Detailed Investigation and Analysis of Using Machine Learning Techniques for Thyroid Diagnosis." In 2023 International Conference on Emerging Smart Computing and Informatics (ESCI), pp. 1-5. IEEE, 2023.

[44] Kumar, Ajay, Raj Gaurang Tiwari, Naresh Kumar Trivedi, Abhineet Anand, Ambuj Kumar Agarwal, and Devendra Prasad. "Extended Network Lifespan with Fault-Tolerant Information Transmission." In 2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART), pp. 218-222. IEEE, 2021.

[45] Kumar, Ajay, Raj Gaurang Tiwari, Abhineet Anand, Naresh Kumar Trivedi, and Ambuj Kumar Agarwal. "New Business Paradigm using Sentiment Analysis Algorithm." In 2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART), pp. 419-423. IEEE, 2021.

[46] Tiwari, Raj Gaurang, Abeer A. Aljohani, Rajat Bhardwaj, and Ambuj Kumar Agarwal. "Virtual reality in tourism: assessing the authenticity, advantages, and disadvantages of VR tourism." Augmented and Virtual Reality in Social Learning: Technological Impacts and Challenges 3 (2023): 215.

[47] Tiwari, Raj Gaurang, Sandip Vijay, Sandeep Dubey, Ambuj Kumar Agarwal, and Megha Sharma. "Relevance and Predictability in Wireless Multimedia Sensor Network in Smart Cities." In Convergence of IoT, Blockchain, and Computational Intelligence in Smart Cities, pp. 251-262. CRC Press, 2023.

[48] Tiwari, Raj Gaurang, Ambuj Kumar Agarwal, and Mohammad Husain. "Integration of virtual reality in the e-learning environment." Augmented and Virtual Reality in Industry 5.0 2 (2023): 253.