International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys.13, No.1 (May-23)

http://dx.doi.org/10.12785/ijcds/1301103

Automatic Spoiler-Sensitive Video Preview Generation with Reinforcement Learning

Mayank Saxena[1] **and Rakhi Saxena**[2,*]

[1]*Columbia University, New York, USA*
[2]*Deshbandhu College, University of Delhi, New Delhi, India*
[*]*Corresponding author*

Received 8 Jun. 2022, Revised 12 May. 2023, Accepted 14 May. 2023, Published 30 May. 2023

Abstract: With the plethora of video content uploaded to the internet each day, the amount of watchable content far exceeds the amount of time available for us to consume it. That being said, the aim of this research is not to create something that does the watching for us and merely generates a short summary; instead, the goal is to help us select videos that we would enjoy watching the most while preserving the viewing-integrity of the original video. In other words, we wish to be able to generate preview video summaries that do not reveal any essential plot items. To accomplish this task, we developed an algorithm that is able to select a fraction of a video's total frames to create a video trailer/summary while excluding important frames that would detract from the main viewing experience (i.e. it does not include any main plot points, spoilers, plot twists, etc.). This output video would allow us to decide better whether we are interested in a particular video, and at the same time will still allows us to fully enjoy the video if we do choose to watch it since we will have avoided any spoilers.

Keywords: Video Summarization, Reinforcement Learning, Spoiler-Sensitive

1. INTRODUCTION

Research in the field of video summarization has gained attention over the last few years due to the exponential increase of video content that is so readily available to people worldwide. However, with the multitude of options available to watch, deciding what to watch is increasingly hard. To facilitate this decision making, video summarization methods aim to generate brief, condensed video synopsis while maintaining the representativeness as well as diversity of the original content [1]. However, these methods do not take into account spoilers while generating of video summaries. Attempts at spoiler-sensitive video summaries, especially non-supervised approaches, are very limited. Applications for developing spoiler-sensitive summaries lie in automatic generation of movie trailers as well as generation of trailers for sports/e-sports highlights. The current methodology for creating movie trailers involves a lot of manual labor and does not involve any kind of automation. We believe that although there is certainly an art involved in creating a captivating and interesting trailer, this problem can be looked at as a traditional video summarization problem, but with an additional constraint of removing the spoiler from the generated summary.

A spoiler is an element of a movie plot or a show that reveals the plot twist or climax, thus spoiling a viewer's experience when watching the video for the very first time. In other words, it is information that was intended to be kept hidden from the viewer and not be disclosed prematurely [2]. Therefore, it is very important that we do not incorporate any spoilers in the generated video when automating the process of creating a movie trailer. This will ensure that the movie watching experience remains enjoyable even after viewing the automated video summary.

Inspired by the work done by Zhou et al. [1], we accomplish the task of spoiler-sensitive video summarization through a sequential decision making process using a deep summarization network. For the training of the network, we used a reinforcement learning based approach which used a reward function based on the diversity and representativeness of the generated summaries. We use a similar model architecture as [1] but modified the reward function in order to account for the spoilers.

The reward function models the intuitive yardstick of desirable properties of a high-quality video summary and consists of two components - a diversity reward and a representativeness reward. The diversity reward measures the degree of diversity of the video summary and the representativeness reward measures how well selected key frames represent the original video [1]. The former is calculated by evaluating the dissimilarity between the frames whereas the latter is calculated by quantifying the distance between frames and

E-mail address: ms5736@columbia.edu, jyc2160@columbia.edu, rsaxena@db.du.ac.in

their nearest selected key frames. Reward obtained by the agent is higher when the summary exhibits high diversity as well as high representativeness. Spoilers in the storyline are key frames towards which temporally near frames culminate by building the narrative. The representativeness reward evaluates how representative the selected frames are of the other frames through centrality in the features space. Therefore, we believe that frames with spoilers will have a high representative score and we should exclude those scenes from our summary. Towards that end, we modified the reward function to make it suitable for our problem by penalizing summaries with high representativeness. We want the agent to favor a diverse summary that is partially representative over a non-diverse summary that is non-representative.

Reinforcement learning is deployed to train the network as the intent is to improve the frame selection process repetitively until no further improvement is possible [1]. Also, the training process does not require any labels and is thus completely unsupervised. Human labels for frame importance and spoilers can be highly subjective, so an unsupervised approach avoids these biases.

For training and evaluation of our network, since there does not exist any dataset for videos and their corresponding spoilers, we decided to use the SumMe [3] dataset - a popular video summarization dataset that has importance scores associated for each frame. We changed the previously annotated importance scores by manually assigning an importance score of 0 to every frame which we considered as a spoiler for the video. Even though these importance scores were not used for training, we used this information for evaluating our model. We compared the summaries generated by our network with a state-of-the-art video summarization network [1], [4], [5], [6], [7] and found that our model learned to account for spoilers better due to the new reward function.

2. Related Works

There exist many approaches that tackle the more traditional video summarization problem. Recent methods rely on deep neural network architectures and the training strategies can be categorised as supervised and unsupervised [8], [9].

Supervised summarization methods are trained using available ground truth data indicating importance of video frames [8]. These methods predict importance scores for input frames by modelling the spatio-temporal dependency between them. Researchers have improvised these methods by introducing tensor-train embedding layers [6], attention mechanisms [10], [11], and by embedding semantic preserving networks [12]. The problem with supervised strategies is that the production of ground-truth data is laborious, time-consuming and challenging since it calls for manual annotation of video frames with importance scores.

Unsupervised summarization approaches overcome the need for ground truth data by training the model using heuristics such as representativeness, diversity, coherence, sparsity, uniformity, ad dispersion of the input features [1], [4], [7], [13]. Since the proposed approach falls under the unsupervised video summarization category, we cover recent literature from this category in more detail below.

Zhou [1] learned a deep summarization network to make decisions on which frames to include in the video summary based on a diversity-representativeness reward function using reinforcement learning. This approach is a large inspiration for our own methods. Yaliniz et al. [13] also use reinforcement learning but in addition to diversity-representativeness reward functions, they employ a uniformity reward function with the aim of enhancing the coherence of the video summary. Gonuguntla et al. [7] train a Temporal Segment Network using a reward function that aims to preserve the spatio-temporal order of the video frames in the summary.

Lu and Grauman [14] discover the story of an egocentric video by a defined random-walk based metric of influence between subshots that captures event connectivity. This metric defines a clear objective for the optimal k-subshot summary. Similarly, Zhang [15] uses Long Short-Term Memory (LSTM) to model temporal dependencies between video frames. This model accounts for sequential structure to generate video summaries. Wang et al [4] introduce an unsupervised auxiliary summarization loss module with LSTM to capture the long-term dependencies for video summarization. Otani et al. [16] look to semantics for video summarization. This paper details a deep neural network to map videos and their descriptions to a latent semantic space, and deep video segment features are clustered for summarization. Ma [17] proposes methods of audio-visual attention model features to model the viewer's attention. This approach forgoes semantic analysis in favor of computational attention models. Zhao et al. [6] present a method that trains the deep learning model by incorporating feedback from reconstruction of the video from the generated summary.

Another perspective for video summarization is to exploit additional modalities such as text-based video metadata in addition to the video frames for learning [18]. Gaikwad et al. [19] use publicly available metadata namely, IMDb plot summaries and match it with scene dialogues, available through subtitles to create movie previews. The movie2trailer framework of Orest et al. [20] creates high-quality trailers by identifying anomalous frames relying on the selected set of visual and audio features. Xu [21] proposes fixation variance, a measure of video attractiveness, and learns an attractiveness model to produce video summaries with maximal attractiveness. This trailer generation aims to encourage viewers to watch the original video. Smeaton [22] focuses on action movies, extracting audiovisual features for selecting frames for exciting scenes to include in the video summary. Irie [23] presents Vid2Trailer, a

content-based movie trailer generation method. This method extracts the movie title logo and main theme music, as well as performs affective content analysis to maximize affective impact of the video summary for effective advertisement of the movie.

However, video summarization with the additional constraint of excluding spoilers or climactic scenes is a relatively novel field. Summaries of videos such as movies or sporting events need to not just capture essence of videos but exclude any video frames that would spoil the viewing experience. In other words, the aim is to generate preview video summaries that do not reveal any essential plot items. Overall, very few attempts have been made to explicitly exclude any essential scenes.

With the aim of exploiting this research gap, we propose a spoiler sensitive video preview generation method using reinforcement learning.

3. Methodology

The SumMe dataset [3] contains 25 videos shot using both still and moving cameras; the videos cover topics such events, sports and holidays. Each video in this dataset ranges from 1 to 6 minutes and each frame is manually annotated by 15 to 18 persons for an importance score, resulting in multiple ground truth summaries per video. However, we modified the annotations for all these videos by assigning an importance score of 0 to all the frames which we identified as spoilers for the video. For example, video_8 in the dataset shows a chef performing the "Hibachi Volcano Onion Trick". For nearly 70% of the duration of the video, the chef is seen preparing the ingredients, cutting the onions, and building the onion stack. Towards the end of the video, the chef is shown pouring a flammable liquid inside the stack, after which he lights it on fire using a matchstick to imitate a volcano. For this particular video, we can say that the spoiler is the scene in which the chef is shown lighting the onion volcano on fire. If we were to use a video summarization technique, we would definitely want that scene to be present in the summary, however, for a spoiler-sensitive summary, we want to omit this scene and include other scenes instead which are important to the story line of the video. Similarly, we identified spoilers in all the videos in the dataset and annotated the importance score of these videos as 0 for those frames. Our modified SumMe dataset can be found at [24].

We used the method of standard 5-fold cross validation which is suggested in [25] for evaluating our method i.e. we used 80% of videos for training and the rest for testing. Details regarding the architecture of the network are mentioned in the next section.

4. Implementation

Our implemented deep summarization network is an adaption of the sequential decision-making process proposed by Zhou [1]. More concretely, the deep summarization network (DSN) is trained to associate probabilities with video frames in order to identify the frames that will be part of the video preview output. Using a reinforcement learning framework, we train our DSN with a diversity-representativeness reward function to assess our generated summaries. The overall learning process is illustrated in Figure 1.

The foundation of our deep summarization network (DSN) is an encoder-decoder framework. The video frames $\mathcal{V} = \{v_i\}_{i=1}^{N}$ are provided as input to the encoder which is a convolutional neural network (CNN). The output from the CNN are features $\{y_i\}_{i=1}^{N}$ extracted from the frames. The extracted features are input to the decoder which is a bidirectional recurrent neural network (BiRNN) with a fully connected layer. The decoder outputs forward and backward hidden states $\{s_i\}_{i=1}^{N}$ that sum up the future and past information of the given frame. The final fully connected layer applies the sigmoid function (represented by σ) to predict a probability ρ_i the particular frame will be selected in the action α_i where $\alpha_i \in \{0, 1\}$ indicates whether the i^{th} frame will be included in the summary or not.

$$\rho_i = \sigma(W s_i) \tag{1}$$

$$\alpha_i \sim \text{Bernoulli}(\rho_i) \tag{2}$$

The output video summary is the ordered set of the selected frames $F = \{f_{y_i} | \alpha_{y_i} = 1, i = 1, 2, ...\}$.

For the CNN, we use GoogLeNet [26] pretrained on ImageNet [27] to extract the visual features from the final layers. The RNN cells consist of long short-term memory (LSTM) to capture the temporal dependencies in each video frame. The training consists of updating the weights of the decoder.

After generating summaries, the DSN will receive reward $R(\mathcal{F})$ depending on how successful the summary is. In each iteration, the DSN will attempt to maximize the expected reward of the summaries it produces. The reward function as defined by Zhou [1] is designed as a function of two asects - diversity reward $\mathcal{DR}$ and representativeness reward $\mathcal{RR}$.

The diversity reward assesses the dissimilarity between the feature vectors of the selected frames. Let the indices of the selected frames be $\mathcal{F} = \{y_i | \alpha_{y_i} = 1, i = 1, \ldots, |\mathcal{F}|\}$. The $\mathcal{DR}$ is computed as the average of the pairwise distance between the features of the selected frames:

$$\mathcal{DR} = \frac{1}{|\mathcal{F}|(|\mathcal{F}| - 1)} \sum_{f_i \in \mathcal{F}} \sum_{\substack{f_j \in \mathcal{F} \\ j \neq i}} d(y_i, y_j) \tag{3}$$

with d as the distance function calculated as

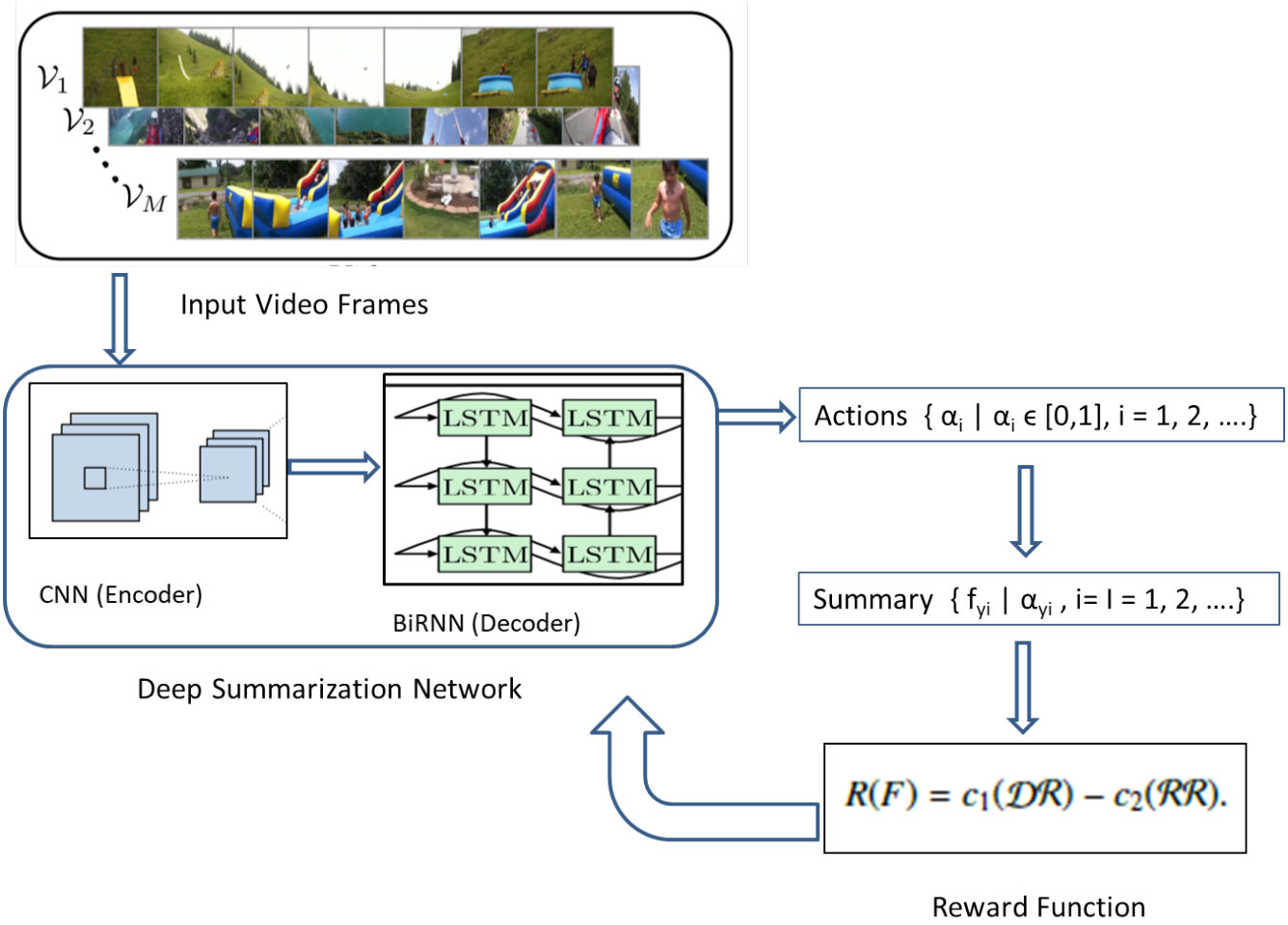


Figure 1. Deep Summarization Network (DSN) with reinforcement learning; DSN receives input video frames and takes actions as binary vector corresponding to which frames to select as the output summary. The diversity and representativeness of the summary is computed based on feedback reward function.

$$d(y_i, y_j) = 1 - \frac{y_i^\intercal y_j}{||y_i||_2 ||y_j||_2}. \quad (4)$$

The agent's diversity reward will be higher when the frames selected for the video summary exhibit wider diversity as well as more dissimilarity between each other.

The representativeness reward evaluates how representative the selected frames are of the other frames through centrality in the features space. Gygli et al. pose assessment of representativeness as the k-medoids problem [28]. The agent's representativeness reward will be higher when the the mean squared errors between video frames and their nearest medoids is higher. Hence, we define $\mathcal{RR}$ as

$$\mathcal{RR} = \exp(-\frac{1}{N}\sum_{i=1}^{N}\min_{j\in\mathcal{F}}||y_i - y_j||_2). \quad (5)$$

Finally, we combine $\mathcal{DR}$ and $\mathcal{RR}$ into a single reward to drive the training of DSN:

$$R(F) = c_1(\mathcal{DR}) - c_2(\mathcal{RR}). \quad (6)$$

It is important to note that the representativeness is treated as a penalty for the overall reward. Scenes that are essential to the video's plot and may contain spoilers will score highly in representativeness. As such, we wish to train the agent to generate video summaries that exclude these scenes. However, we do not want the agent to weigh diversity and representativeness equally, so we multiply each reward by c_1, c_2. We set different constants for c_1, c_2 such that the the agent favors a diverse summary that is partially representative over a non-diverse summary that is non-representative. During training, we found best results with $c_1 = 0.75$ and $c_2 = 0.25$. More on this will be discussed in the evaluation section.

The goal of the agent is to learn the DSN's optimum policy function γ_ϕ with parameters ϕ by maximizing the expected rewards

$$O(\phi) = \mathbb{E}_{p_\phi(\alpha_{1:N})}[R(F)], \tag{7}$$

where $p_\phi(\alpha_{1:N})$ denotes the probability distributions over all actions.

As proposed by Williams [29], we implement the REINFORCE algorithm to compute the derivative of the objective function $O(\phi)$ w.r.t. the parameters ϕ:

$$\nabla_\phi O(\phi) = \mathbb{E}_{p_\phi(\alpha_{1:N}}[R(F)\sum_{i=1}^{N}\nabla_\phi \log \gamma_\phi(\alpha_i|s_i)] \tag{8}$$

where α_i is the action taken by DSN at time i and s_i is the output hidden state from the BiRNN.

However, as it is difficult to directly compute the expectation over the high-dimensional action sequences, the episodic REINFORCE algorithm is deployed to approximate the gradient by computing the average gradient of N runs over the same video as follows:

$$\nabla_\phi O(\phi) \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T} R_n \nabla_\phi \log \gamma_\phi(\alpha_i|s_i), \tag{9}$$

where R_n is the reward computed at the n^{th} episode.

This approximation of the gradient may contain high variance, so it may be difficult for the network to converge. To remedy this, we decrement the reward by a constant baseline l which is computed as the moving average of rewards received so far. Our approximate calculation of the gradient becomes

$$\nabla_\phi O(\phi) \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{T}(R_n - l)\nabla_\phi \log \gamma_\phi(\alpha_i|s_i)]. \tag{10}$$

Naturally, a video summary that selects more frames will produce a higher reward, therefore, we apply a regularization term on the probability distribution $\rho_{1:T}$ in order to limit the number of selected frames in the summary. Similar to the regularization done by Mahasseni [30], we minimize the following during training:

$$\mathcal{L}_{\text{pt}} = \|\frac{1}{T}\sum_{=i1}^{T}\rho_i - \epsilon\|^2, \tag{11}$$

where ϵ denotes the percentage of frames to be selected for the summary. Furthermore, we minimize the risk of overfitting by adding the following regularization term on the weight parameters ϕ

$$\mathcal{L}_{\text{wt}} = \sum_{x,y}\phi_{x,y}^2. \tag{12}$$

Further, we combine the gradients computed from Eq. (10), Eq. (11), and Eq. (12) and update ϕ via stochastic gradient-based method to optimize the policy function's parameters ϕ.

To perform the video summarization task for a test video, we predict the importance scores of frames by applying the trained DSN. Then we apply Kernel Temporal Segmentation [31] to bin multiple consecutive frames into shots. The importance scores of shots are calculated as an average of importance scores of frames in respective shot. The shot that maximizes the total scores are selected to be included in the summary. Note that we also constrain the length of a summary to 15% of the original video's length. We iteratively added shots into the summary by rank. We found this to produce the best results for spoiler-sensitive summarization, but we also experiment with the Knapsack algorithm via dynamic programming by He [32].

In the Experiments section, we evaluate some generated summaries as well as analyze the performance of different hyperparameters.

$$\phi = \phi - \lambda\nabla_\phi(-O + \beta_a\mathcal{L}_{\text{pt}} + \beta_b\mathcal{L}_{\text{wt}}), \tag{13}$$

where α is the learning rate is denoted by , and the weighting hyperparamters for the regularization terms are denoted by β_a and β_b. Practically, the optimization is performed using the the Adam optimization algorithm [33] which increases the log-probability for actions that produce high rewards and decreases the log-probability of actions that resulted in low rewards.

5. Evaluation

Firstly, we have compared the performance of our spoiler-sensitive model, abbreviated as DR-DSN$_{SS}$ with the reinforcement learning based video summarization model presented in [1], abbreviated as DR-DSN. Then, we compared our results against current state-of-the-art reinforcement learning approaches for the traditional video summarization task.

We compare the performances of DR-DSN$_{SS}$ and DR-DSN models based on multiple parameters:

- number of epochs
- β (weight for summary length penalty term)
- shot selection method (knapsack/rank)

- constants associated with the representativeness and diversity rewards i.e. c_1 and c_2

Model	Epochs	Selection Method	F_1 Score
DR-DSN	5	Knapsack	18.2
DR-DSN	5	Rank	21.3
DR-DSN$_{SS}$	5	Knapsack	22.3
DR-DSN$_{SS}$	5	Rank	15.1
DR-DSN	10	Knapsack	17.2
DR-DSN	10	Rank	19.9
DR-DSN$_{SS}$	10	Knapsack	18.7
DR-DSN$_{SS}$	10	Rank	12.6
DR-DSN	15	Knapsack	15.9
DR-DSN	15	Rank	20.7
DR-DSN$_{SS}$	15	Knapsack	19.6
DR-DSN$_{SS}$	15	Rank	12.9

TABLE I. Comparison with respect to number of epochs and the shot selection method ($\beta = 0.01$ and $c_1 = c_2 = 1$).

Table I compares the performance of the two models with respect to different number of epochs (5, 10 and 15) as well as against the two different shot selection methods. Table I shows that with equal magnitude c_1, c_2, our model achieves the highest score with 5 epochs under the Knapsack selection method. From table I we see that the "Rank" shot selection method performs better than the "Knapsack" method for equal number of iterations of the same model. We also see that the model's performance starts to deteriorate after 5 epochs.

From table II we see that after adjusting the value of the parameter β to 0.1 our model performs better than before. Using these two results, we decided to move forward and try different values for the coefficients for the diversity reward and the representativeness reward.

Table III includes information regarding the different constants which we tried for the coefficients for the two rewards. The constant c_1 is the coefficient of the diversity reward whereas the constant c_2 is the coefficient of the representativeness reward. We tried different values in order to find the optimal parameters such that we could penalize summaries with high representativeness but reward the agent for a summary for with medium representativeness at the same time. We can see that our best model has an F_1 score of 26.5 and it has the following parameters:

Model	Epochs	Selection Method	F_1 Score
DR-DSN	5	Knapsack	17.5
DR-DSN$_{SS}$	5	Knapsack	19.1
DR-DSN	15	Knapsack	19.7
DR-DSN$_{SS}$	15	Knapsack	17

TABLE II. Model performance when parameters are $\beta = 0.1$ and $c_1 = c_2 = 1$.

Model	c_1	c_2	Selection Method	F_1 Score
DR-DSN$_{SS}$	0.75	-0.25	Knapsack	17.6
DR-DSN$_{SS}$	0.75	-0.25	Rank	20.8
DR-DSN$_{SS}$	0.75	0.25	Knapsack	22.2
DR-DSN$_{SS}$	0.75	0.25	Rank	26.5
DR-DSN$_{SS}$	0.9	-0.1	Knapsack	18.4
DR-DSN$_{SS}$	0.9	-0.1	Rank	19
DR-DSN$_{SS}$	0.9	0.1	Knapsack	21.3
DR-DSN$_{SS}$	0.9	0.1	Rank	21.7

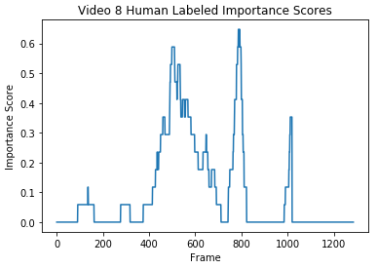
TABLE III. Comparing model performance with respect to c_1 and c_2 and the two shot selection methods ($\beta = 0.1$).

- $c_1 = 0.75$
- $c_2 = 0.25$
- $\beta = 0.1$
- Number of Epochs = 5
- Shot Selection Method = "Rank"

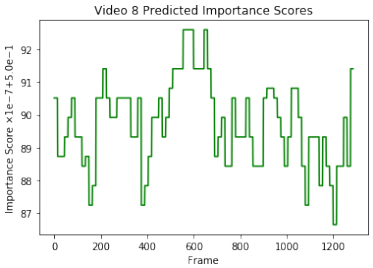
Figure 2 shows the ground truth, the predicted scores and the selected frames for video 8. We can see that there is a huge spike in the importance scores around frame number 800, which is the frame in which the chef pours the oil in the onion volcano and proceeds to light it with a matchstick. A few of the frames of video_8 can be seen in figure 3. We can see that from frames numbered 100 to 400, the chef prepares the volcano (a scene which is important towards the plot of the video but it is not a spoiler). As mentioned before, we consider the spoiler in this video, the scene in which the chef lights the onion volcano on fire. Thus, we can see that our model does a decent job in selecting the frames which are not spoilers and at the same time, incorporates other important frames in the video summary too.

Similarly, we show the ground truth, the predicted scores and the selected frames for video_24 in figure 4. The spikes in the ground truth of the importance scenes correspond to the times in which the person jumps off the cliff into the water (See figure 5). The longest spike is seen around frame number 1100 and we can see that this frame corresponds to the time when the person has jumped and is currently in the air. For this video, we can say that this scene is a spoiler, since we would not want the viewer to know about this scene while they watch the trailer. We can see that the predicted scores are not as high for the scene in which the person jumps and in-fact, it is higher for the scenes which have an importance score in the middle range, thus, preserving the video summary from including the spoiler, yet, including other important scenes.

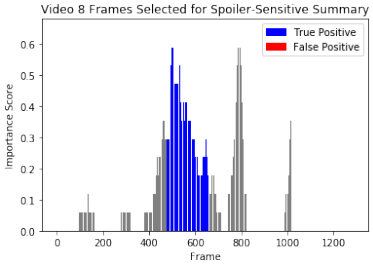
For a more qualitative comparison of our model (DR-DSN$_{SS}$) with the DR-DSN model, we compared the F_1 score of the 5 videos which were present in the test set. The results can be seen in table IV. We see that for every video, our model has a higher F_1 score. However, for video_12, both of the models have an F_1 score of 0.0. We speculate that this is the case because the duration of the video is



(a) Ground Truth: Human Labelled Importance Scores



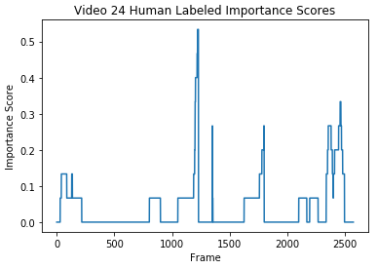
(b) Predicted scores by our model DR-DSN$_{SS}$



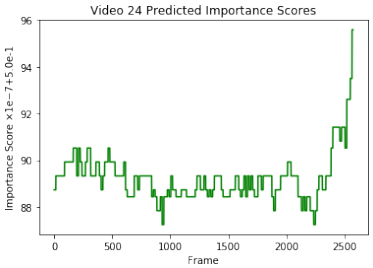
(c) Frames selected for generating summary

Figure 2. Scores for Video 8: Hibachi Volcano Onion Trick

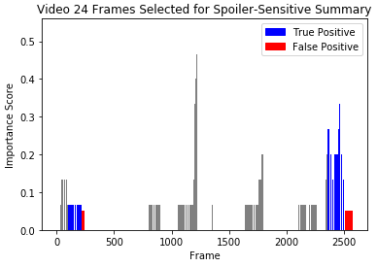
Figure 3. Frames 100, 400, 800 and 1000 for Video 8



(a) Ground Truth: Human Labelled Importance Scores



(b) Predicted scores by our model DR-DSN$_{SS}$



(c) Frames selected for generating summary

Figure 4. Scores for Video 24: Paluma Jump

Figure 5. Frames 300, 900, 1100 and 2200 for Video 24

quite small (39 seconds) and while manually annotating an importance score of 0 to the spoiler, we might have assigned a score of 0 to a large fraction of the original frames since the spoiler-sensitive summary which gets generated for this video is only of 3 seconds which correspond to the last 3 seconds of the video.

We also computed the cross-correlation between the ground truth and the outputs of the two models (V). We decided to use another metric for evaluation besides the F_1 score as it is a poor indicator for evaluating video summaries [34]. It has been observed that randomly generated summaries achieve comparable or better performance to the state-of-the-art when evaluated using an F_1 metric. [34] suggests an alternative approach for assessing the predicted importance scores by finding the cross-correlation between the predictions and the human annotations. From table V, we observe that when using the cross-correlation metric, the DR-DSN model performs better than the DR-DSN$_{SS}$ model. For all the videos we see that the predictions of the DR-DSN model have a higher positive correlation as compared to the predictions of the DR-DSN$_{SS}$ model.

For further evaluation of the proposed method DR-

Video ID	DR-DSN$_{SS}$	DR-DSN
video_8	48.5	12.2
video_24	39.5	23.6
video_17	33.2	13.9
video_2	19.6	0.0
video_12	0.0	0.0

TABLE IV. F_1 Score Comparison on Test Videos.

Video ID	DR-DSN$_{SS}$	DR-DSN
video_8	58.37	61.94
video_24	48.92	51.58
video_17	314.32	332.66
video_2	244.65	258.67
video_12	45.12	47.74

TABLE V. Cross-Correlation Comparison on Test Videos.

DSN$_{SS}$, we have compared its performance with recent state of the art unsupervised video summarization methods using the entire set of available videos of the SumMe dataset. We used GoogLeNet [26] pretrained on ImageNet [27] to extract the visual features of the video frames from the final layers. We used the method of standard 5-fold cross validation which is suggested in [25] for evaluating the methods i.e. we used 80% of videos for training and the rest for testing. Note that during our experiments, we performed hyperparameter tuning for our method as detailed in previous paragraphs to achieve optimal performance.

For comparative evaluation, we compute the F_1 score to measure the similarity between the selected key frames in the video summary and the ground-truth annotations. Table VI shows the resultant average F_1 scores achieved by competitive methods. As is evident from the table, DR-DSN$_{SS}$ has a higher F_1 score compared to all other methods showcasing its better performance.

6. Discussion

In this paper, we use a label-free, diversity-representativeness reward function to train a reinforcement learning model for spoiler-sensitive video summarization. The agent receives a reward calculated based on the generated summary's level of diversity and representativeness, and updates the policy's parameters such that diverse and non-representative frames are more likely to be selected for the spoiler-free video summary.

Method	F_1 score
DR-DSN$_{SS}$	48.2
AuDSN [4]	47.7
ACGAN [5]	46.0
PCDL [6]	42.7
EDSN [7]	42.6
DR-DSN [1]	41.4

TABLE VI. F_1 Score Comparison of DR-DSN$_{SS}$ with State of the Art.

Future improvements to this current approach can explore other reward functions. Since our approach was largely based off another for traditional video summary, different reward functions may better train the agent to select important frames that do not contain any spoilers. Specifically, instead of maximizing negative representativeness, perhaps a reward can be built around semantics or climax, and our agent would instead be trained to generate video summaries that are minimal in semantic significant or climactic level.

Furthermore, our DSN encoder only extract visual features from each video frames; as such, crucial information within each frame is potentially lost during training. We can expand our encoder to incorporate multimodal signals that may be available during each frame for more accurate feature extraction. For instance, audio, which can help the agent learn to identify diverse and (non-)representative frames for the video summary. Additionally, with the case of live-streamed videos like sports games, there are often live chat logs that contain viewers' comments for each frame. This is another channel that can help the agent generate more accurate spoiler-free video summaries.

For testing, many of our ground truth labels were highly subjective and involved the labelers to determine both the level of significance and presence of spoilers for each particular frame. Without a large sample of labels, it is possible that our labels are biased, and thus our evaluations an imperfect measure of how well our agent performed. Conducting large user studies and experiments to better label positive frames for the evaluation of the generated summaries would allow us to be more confident in the performance of our model.

The evaluation of our results is largely founded on the performance of the F-score in evaluating video summarizations; however, Otani [16] has found that the F-score may be a poor metric for video summarization as state-of-the-art video summarization models only achieve an average F-score of about 40%. For more robust evaluation metrics, it may be worth exploring other metrics like cross- correlation.

In conclusion, we have trained a reinforcement learning model to learn to generate spoiler-free video summaries with a diversity and representativeness reward. Experiments on the SumMe dataset showed that using reinforcement learning with a diversity and non-representativeness reward to generate spoiler-free video summaries out performs other reinforcement learning-based approaches for more traditional video summarization. Application of this model can useful for generating movie trailers and sports highlights.

REFERENCES

[1] K. Zhou, Y. Qiao, and T. Xiang, "Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, Apr. 2018.

[2] M.-W. Dictionary, "Spoiler definition," https://www.merriam-webster.com/dictionary/spoiler.

[3] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in *ECCV*, 2014.

[4] X. Wang, Y. Li, H. Wang, L. Huang, and S. Ding, "A video summarization model based on deep reinforcement learning with long-term dependency," *Sensors*, vol. 22, no. 19, 2022.

[5] X. He, Y. Hua, T. Song, Z. Zhang, Z. Xue, R. Ma, N. Robertson, and H. Guan, "Unsupervised video summarization with attentive conditional generative adversarial networks," in *Proceedings of the 27th ACM International Conference on Multimedia*, ser. MM '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 2296–2304.

[6] B. Zhao, X. Li, and X. Lu, "Tth-rnn: Tensor-train hierarchical recurrent neural network for video summarization," in *IEEE Transactions on Industrial Electronics*, 2020.

[7] B. M. N. Gonuguntla and N. B. Puhan, "Enhanced deep video summarization network," in *30th British Machine Vision Conference (BMVC), Workshop on Applications of Egocentric Vision Workshop (EgoApp)*. BMVC, 2019.

[8] E. E. Apostolidis, E. Adamantidou, A. I. Metsai, V. Mezaris, and I. Patras, "Video summarization using deep neural networks: A survey," *Proc. IEEE*, vol. 109, no. 11, pp. 1838–1863, 2021.

[9] V. Tiwari and C. Bhatnagar, "A survey of recent work on video summarization: approaches and techniques," *Multimedia Tools and Applications*, vol. 80, no. 18, pp. 27 187–27 221, 2021.

[10] L. Lebron Casas and E. Koblents, "Video summarization with lstm and deep attention models," in *MultiMedia Modeling*, I. Kompatsiaris, B. Huet, V. Mezaris, C. Gurrin, W.-H. Cheng, and S. Vrochidis, Eds. Cham: Springer International Publishing, 2019, pp. 67–79.

[11] L. Ping, Y. Qinghao, Z. Luming, Y. Li, X. Xianghua, and S. Ling, "Exploring global diverse attention via pairwise temporal relation for video summarization," *Pattern Recognition*, vol. 111, p. 107677, 2021.

[12] Z. Ji, K. Xiong, Y. Pang, and X. Li, "Video summarization with attention-based encoder–decoder networks," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 30, no. 6, pp. 1709—-1717, 2020.

[13] G. Yaliniz and N. Ikizler-Cinbis, "Using independently recurrent networks for reinforcement learning based unsupervised video summarization," *Multimedia Tools and Applications*, vol. 80, pp. 17 827—-17 847, 2021.

[14] Z. Lu and K. Grauman, "Story-driven summarization for egocentric video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2714–2721.

[15] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Video summarization with long short-term memory," in *European conference on computer vision*. Springer, 2016, pp. 766–782.

[16] M. Otani, Y. Nakashima, E. Rahtu, J. Heikkilä, and N. Yokoya, "Video summarization using deep semantic features," in *Asian Conference on Computer Vision*. Springer, 2016, pp. 361–377.

[17] Y.-F. Ma, L. Lu, H.-J. Zhang, and M. Li, "A user attention model for video summarization," in *Proceedings of the tenth ACM international conference on Multimedia*. ACM, 2002, pp. 533–542.

[18] R. V. K. Vamsi and D. Subburaman, "A review on video summarization," in *Proceedings of International Conference on Deep Learning, Computing and Intelligence*, G. Manogaran, A. Shanthini, and G. Vadivu, Eds. Singapore: Springer Nature Singapore, 2022, pp. 495–504.

[19] B. Gaikwad, A. Sontakke, M. Patwardhan, N. Pedanekar, and S. Karande, "Plots to previews: Towards automatic movie preview retrieval using publicly available meta-data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, October 2021, pp. 3205–3214.

[20] R. Orest and F. Taras, "movie2trailer: Unsupervised trailer generation using anomaly detection," in *25th Computer Vision Winter Workshop*, February 2020.

[21] H. Xu, Y. Zhen, and H. Zha, "Trailer generation via a point process-based visual attractiveness model," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[22] A. F. Smeaton, B. Lehane, N. E. O'Connor, C. Brady, and G. Craig, "Automatically selecting shots for action movie trailers," in *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*. ACM, 2006, pp. 231–238.

[23] G. Irie, T. Satou, A. Kojima, T. Yamasaki, and K. Aizawa, "Automatic trailer generation," in *Proceedings of the 18th ACM international conference on Multimedia*. ACM, 2010, pp. 839–842.

[24] J. Chou and M. Saxena, "Modified summe dataset," https://github.com/mayank26saxena/spoiler-sensitive-video-summarization.

[25] Z. K., C. W., S. F., and G. K., "Video summarization with long shortterm memory," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016, pp. 766–782.

[26] Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, C. Szegedy, and W. Liu, "Going deeper with convolutions," in *CVPR*, 2015.

[27] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[28] M. Gygli, H. Grabner, and L. Van Gool, "Video summarization by learning submodular mixtures of objectives," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3090–3098.

[29] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, vol. 8, no. 3-4, pp. 229–256, 1992.

[30] B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video

summarization with adversarial lstm networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 202–211.

[31] D. Potapov, M. Douze, Z. Harchaoui, and C. Schmid, "Category-specific video summarization," in *European conference on computer vision*. Springer, 2014, pp. 540–555.

[32] Y. He, J. Song, J. Zhang, and H. Gou, "Research on genetic algorithms for solving static and dynamic knapsack problems," *Appl Res Comput*, vol. 32, no. 4, pp. 1011–1015, 2015.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[34] O. M., N. Y., R. E., and H. J., "Rethinking the evaluation of video summaries," in *arXiv:1903.11328*. ACM, 2019.

Mayank Saxena is a Software Engineer at Amazon, New York. He completed his Masters in Computer Science in 2018 from the Columbia University, New York. His areas of interest include social network analysis, machine learning, big data analytics and deep learning.

Email: ms5736@columbia.edu

Dr. Rakhi Saxena is an Associate Professor in the Department of Computer Science, at Deshbandhu College, University of Delhi. She received her Ph.D. from the Department of Computer Science, University of Delhi, in 2020. Her research areas of interest include social network analysis, graph mining, recommender systems, multilayer networks, and big data analytics.

Email: rsaxena@db.du.ac.in