TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS

THESIS NO: 075MSICE016

Video Summarization using Spatio-Temporal Features by Detecting Representative Content based on Supervised Deep Learning

by
Ramesh Kumar Sah

A THESIS
SUBMITTED TO THE DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN INFORMATION AND COMMUNICATION ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING
LALITPUR, NEPAL

August, 2021

Video Summarization using Spatio-Temporal Features by Detecting Representative Content based on Supervised Deep Learning

by

Ramesh Kumar Sah

075MSICE016

Thesis Supervisor

Mr. Sharad Kumar Ghimire

A thesis submitted in partial fulfillment of the requirements for the degree of Masters of Science in Information and Communication Engineering

Department of Electronics and Computer Engineering

Institute of Engineering, Pulchowk Campus

Tribhuvan University

Lalitpur, Nepal

August, 2021

COPYRIGHT©

DECLARATION

I declare that the work hereby submitted for Master of Science in Information and Communication Engineering (MSICE) at IOE, Pulchowk Campus entitled **"Video Summarization using Spatio-Temporal Features by Detecting Representative Content based on Supervised Deep Learning"** is my own work and has not been previously submitted by me at any university for any academic award.

I authorize IOE, Pulchowk Campus to lend this thesis to other institution or individuals for the purpose of scholarly research.

Ramesh Kumar Sah
075MSICE016
Date: August, 2021

RECOMMENDATION

The undersigned certify that they have read and recommended to the Department of Electronics and Computer Engineering for acceptance, a thesis entitled **"Video Summarization using Spatio-Temporal Features by Detecting Representative Content based on Supervised Deep Learning"**, submitted by **Ramesh Kumar Sah** in partial fulfillment of the requirement for the award of the degree of **"Master of Science in Information and Communication Engineering"**.

.....

Supervisor: Mr. Sharad Kumar Ghimire,
Department of Electronics and Computer Engineering,
Institute of Engineering, Tribhuvan University

.....

External Examiner: Dr. Pradip Paudyal,
Deputy Director,
Nepal Telecommunication Authority (NTA)

.....

Committee Chairperson: Dr. Basanta Joshi,
Program Coordinator,
Information and Communication Engineering,
Institute of Engineering, Tribhuvan University

Date: August, 2021

DEPARTMENTAL ACCEPTANCE

The thesis entitled **"Video Summarization using Spatio-Temporal Features by Detecting Representative Content based on Supervised Deep Learning"**, submitted by **Ramesh Kumar Sah** in partial fulfillment of the requirement for the award of the degree of **"Master of Science in Information and Communication Engineering"** has been accepted as a bonafide record of work independently carried out by him in the department.

.....

Prof. Dr. Ram Krishna Maharjan

Head of the Department,

Department of Electronics and Computer Engineering,

Pulchowk Campus,

Institute of Engineering,

Tribhuvan University,

Nepal.

ACKNOWLEDGEMENT

I would like to express my deep gratitude to Department of Electronics and Computer Engineering for incorporating Thesis as a part of our syllabus.

It is my immense pleasure to mention the thesis supervisor **Mr. Sharad Kumar Ghimire** and the programme coordinator **Dr. Basanta Joshi** for their valuable feedback and guidance.

I would like to present my acknowledgement to respected Dean, **Prof. Dr. Shashidhar Ram Joshi** and Head of the Department **Prof. Dr. Ram Krishna Maharjan** and to **Dr. Dibakar Raj Pant, Dr. Surendra Shrestha, Prof. Dr Subarna Shakya Dr. Sanjeeb Prasad Panday, Dr. Aman Shakya, Dr. Nand Bikram Adhikari, Dr. Arun Kumar Timilsina, Mr. Babu Ram Dawadi, Mr. Daya Sagar Baral** and other faculties for their precious guidance and constant encouragement.

I am thankful to faculty members of the Department of Electronics and Computer Engineering for providing suitable platform to prepare the thesis.

Sincerely,
Ramesh Kumar Sah
075MSICE016

ABSTRACT

Video Summarization is the approach to generate the compact version of video keeping relevant content intact and eliminating redundancy. In this work, a framework has been proposed which makes use of the spatial and temporal features with self attention from the video sequences to identify the representative content by generating temporal proposals and supervised learning from the data manually created by humans or users. Existing Supervised methods don't deal with the temporal interest and its consistency. For that temporal uniformity is also necessary which can be addressed by predicting the temporal proposals of the video segment. The proposed work treats it as temporal action detection which predicts importance score and location of the segments simultaneously by developing the anchor based method which generates anchors of varying lengths to identify interesting proposals. Moreover the extensive quantitative and qualitative analysis on TVSumm and SumMe datasets augmented with OVP and YouTube datasets justify the effectiveness of the method.

Keywords: Video Summarization, Self Attention, Deep Learning

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CBVRS	Content Based Video Retrival System
VAE	Variational Autoencoder
CNN	Convolution Neural Network
VSUMM	Video Summarization
OVP	Open Video Projects
FPS	Frame Per Second
RNN	Recurrent Neural Network
LSTM	Long Short Term Memory
Bi-LSTM	Bidirectional LSTM
IOU	Intersection of Union
FC	Fully Connected
GT	Ground Truth
NMS	Non-Maximum Suppression
KTS	Kernel Temporal Segmentation

CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

The exponential growth of the video consumption has brought up the new challenges for the browsing and navigating through the video more effectively and efficiently. Video has become the arguably the primary source for the data consumption with the emergence of the data centers and warehouses. Trend of streaming sites and distribution of the videos have become the mainstream in the world of the social media. Though, the consumer might not have enough time to watch the whole video and has to go through complete video to extract the information from it. In those cases, the consumer might just need to get overview of the video without watching the complete video which bestows more relevant event occurred in the video. The conventional news and media distribution methods are quickly replaced by video streaming sites such as YouTube, which are themselves compelled to accustom the rise of uploading videos rather than text and images.

The surge of video content as main source of data consumption for information, the automation of the video summary process has turned predominant. In these times, video summarization has appeared as a daunitng task for machine learning approach, which targets at automatically analyzing the content in video. Video summary application can be helpful in producing highlights for sporting events, movie previews, and generally shortening video to the most appropriate sub sequences, enabling humans to efficiently search vast video repositories. There are a huge number of algorithm for summarizing video, most of which are computationally costly, separately handle video shots, and some of them caters local definition significance. Local definition measured around interest points has been implemented in this suggested technique called keyframe extractions. A moving-image abstract is also called Video Skimming.

One of the most fundamental measures in the field of video summarisation is the main frame video description. This method provides users with an accurate and portable representation of original video content. The basic concept of keyframe extraction converts the entire video frames to a lessser frames that represent most of the frames. Video synopsis greatly decreases details that must be reviewed for the Video Recovery Framework Based on Content (CBVRS). The majority of works extract keyframes after detecting videoshots in the sense of video summarization.

1.2 Problem Definition

There has been huge number of the researches held in the domain of the video summarization recently [3],[4]. These video summarization approaches can be divided into three broad categories: 1)Unsupervised methods [5],[6] , 2) Weakly Supervised methods [7], and 3) Supervised methods [8],[9]. Unsupervised Methods mostly dwells in identifying heuristic criteria like representatives, diversity and sometimes sparsity. Some of the methods are cluster based [36], subset selection,dictionary learning, adversarial learning based[10] and reinforcement learning[11]. Weakly supervised method deals with the additional information includes web priors[12][3], video categories and titles. Even though unsupervised and weakly supervised methods are good performing they lack learning from human summaries which are manually created. This issue is addressed by supervised methods [13],[14]. Supervised Methods comprise sumarization of video based on long short term memory, diverse sequential subset selection, attention based encoder decoder networks. Existing Supervised methods dont deal with the temporal interest and its consistency.

However the issue with these approaches is that for the same contextual segment, frame scores of the video alone cannot be sufficient enough to represent the semantic content. For that temporal uniformity is also necessary which can be addressed by predicting the temporal proposals of the video segment on the basis of action recognition task.

In this proposed work these research gaps have been addressed by adopting a new perspective to video summarization techniques. It has been treated as temporal action detection which predicts importance score and location of the segments simultaneously by developing the anchor based method which generates anchors of varying lengths to identify interesting proposals. Moreover the extensive quantitative and qualitative analysis will justify the effectiveness of the method. The contribution of this thesis works are as follows:

- A new perspective for the framework has been proposed in the domain of video summarization which treats it as temporal representative portion detection problem which detects representative content from the video by generating temporal proposals learned supervised by human created summaries.
- Anchor based mechanism has been followed to generate temporal proposals that can handle variable length of the representative portions and learns the importance scores of that particular portions.
- Extensive analysis and experiments have been performed to investigate effectiveness of the approach.

1.3 Objectives

Video summarization is the task of generating the short synopsis of the original video which eliminates redundancy without loosing the important content of the same. The objective of this thesis work are:

- To make video summarizing framework considering temporal interest selection by predicting important score and segment locations simultaneously.
- To conduct extensive experiments and analysis on different data sets to demonstrate the effectiveness if this proposed work.

1.4 Scope of the Work

Automatic Video summarization has been gaining more popularity's with the rise of the deep learning approaches. This is high in demand where shorter version of the video is required. The scope of the video summarization are listed below.

- Short egocentric videos summarised with the better content selected automatically by this framework can be eye pleasing.
- Web videos with wide range of content can be summarised with only important content to show.
- Summarizing news reports, sports highlights, Movies will allow the user to quickly look through the content and patterns.
- Advent of drones and robots have increased the amount of video recording and analysis. Thus video summarization can ease the analysis and interpretation.
- Surveillance videos can be summarised if the model is trained with labeled data related to videos from CCTV.

CHAPTER 2

LITERATURE REVIEW

Video Summarization has allured a lot of consideration. Identifying and extracting the relevant information from the trivial content is the most daunting challenge in video summarization. There have been lot of researches in the domain of video summarization till date. These can be broadly classified into three categories.

2.1 Unsupervised Video Summarization

K-Means clustering approaches has been prevailed in the video summarization in early works which utilizes the low level features and motion cues to leverage the summary[6]. These methods were able to achieve good performance although with the highly motion camera and varying illumination condition causes degradation in the performance. Unsupervised approaches can further be divided into four different categories.1) Dictionary based learning[15],[16] takes the approach of formulating video by optimizing the loss function. Elhamifar et al. [17] is dictionary based approach. Similarly Roy et. al [18] forms representative method for summarization.2) Subset based selection methods selects the representative frames from the original videos. Elhamifar et al. [19] exploits this subset selection approach to determine the similarity between source and actual sets.3) Reinforcement learning has become one of emerging approach in the domain of video summarization which rewards and punish the agent based on action. It uses discrete sampling of action which gives the generated summaries. Zhou et al. [11] formulated deep network for the summarization based on diversity Representative reward. 4) Adversarial learning based methods uses the learning from the ground truth values and then discriminate the input and output accordingly. Mahasseni et al. [10] formed the network based on adversarial network i.e. on LSTM networks in which generated video is compared with the ground truth in the discriminator to get the summaries video. Rochan and Wang [20] developed the video summarization network using unpaired data. Yuan et al. [21] takes advantages of cycle consistent adversarial

network to make summaries from corresponding videos. Although unsupervised approaches able to give good result it lacks the the integrity of human summaries ground truth data.

2.2 Weakly Supervised Video Summarization

These methods mainly focuses on the additional information which includes web priors[3],[22],video categories[23],[1], Video titles[24]. Khosla et al. [22] takes advantage of the web prior images for summarising the videos. Cai et al. [12] uses the variational autoencoder (VAE) [23] to train the web videos to get summaries of videos. Cai et al. [12] captured the key shots which has more visual contents based on the title of image search. s. Potapov et al. [1] developed a summarization method based on the categories of videos. Panda et al. [23] takes the derivative of classification loss to select the key segments in original videos.

2.3 Supervised Video Summarization

Recent advancement in deep learning and presence of abundant human created and annotated summaries supervised approaches have taken the huge step in performance. Gygli et al. [25] fomulated the video summarizing model which leverage the spatial and temporal information. . Gong et al. [26] as well as Sharghi et al. [27] developed the Detriminantal Point Process [28] as the video summarization model that is a non parametric approach to transfer the strucure from training videos to testing videos. Zhang et al. [29] takes advantage of deep network bidirectional LSTM that estimates importance score of each frames in video. Zhao et al. [30], [2] made use of fixed length hierarchical RNN to discover hierarchical structure of the videos. Yao et al. [61] incorporates spatial information along with the temporal information to build the pairwise ranking model based on deep network. Video summarization is formed as sequence to sequence learning by Zhang et al. [31]. Hussain et al. [32] leverage the advantage of both CNN and Bi-LSTM to compute the multi-view approach for video summarization. In the recent advancement attention models [54] have taken significant approach in this

domain.Ji et al. [33] combined the encoder decoder architecture with attention model. Further Fajtl et al. [8] uses self attention model which is extended version of attention based models.

2.4 Anchor-Based Models

The comprehensive progress in the computer vision specially in the object detection aid in the action localization tasks. Candidate segments are identified using multi scale segments-CNN by Shou et al.[50]. Gao et al. [34] combined the approach of spatial and temporal to form the proposals simultaneously. Xu et al. [36] with the help predefined anchors it predicts the variable length proposals. Chao et al. [35] developed the model that is able to generate multi scale anchor segment for localizing the actions.

CHAPTER 3

METHODOLOGY

3.1 System Block diagram

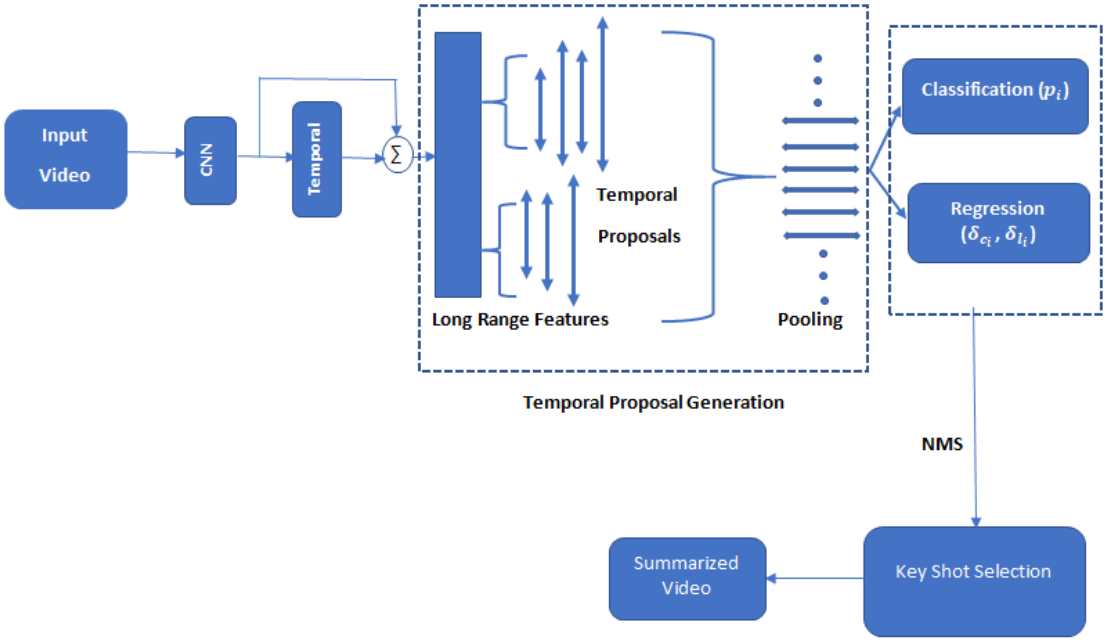


Figure 3.1: Block diagram

The Figure 3.1 depicts working architecture for this proposed algorithm. It can be divided into basic individual steps which can be termed as Extraction of Spatio-Temporal Feature , Generation of Temporal Action Proposals, proposals classification and regression and selection of keyshots.

3.2 Dataset Description

The algorithm will be experimented on the two standard Datasets i.e SumMe dataset and TVSum. These are currently the only datasets appropriatly annotated dataset which can be used for video summarization based on keyshots and these data cannot be sufficient enough to train deep learning models to overcome this, additional datasets viz: OVP and YouTube are used which augment the training

dataset. These datasets are labelled using keyframes but we need keyshot as the annotation for that these keyframes are converted into frame level scores and finally to binary keyshot summaries.

3.2.1 SumMe Dataset

The SumMe dataset is created by [25], the benchmark for evaluating the automatic summary for present and future approaches used for summarization of videos. It contains 25 videos with varying length ranging from one to five minutes.It includes summaries provided by various users, and the length of video is limited to 5% to 15%. By crowd sourcing, summaries were compiled. Length of the summaries produced by humans is limited to within 15% of the original video.

3.2.2 TVSum Dataset

The TVSum Dataset[24] has 50 videos sequences which are downloaded from YouTube. It contains videos like changing vehicle tyre,dog show etc. The ground truth segments are required for training Purpose.

Datasets	No. of Videos	User Number	Contents	Annotation Type	Duration(avg)
SumMe	25	15-18	User generated Videos	Frame-Level Score	146s
TVSum	50	20	Web Videos	Frame-Level Score	235s
OVP	50	5	Various Genre Videos	KeyFrames	98s
YouTube	39	5	Web Videos	KeyFrames	196s

Table 3.1: Descriptions of datasets which will be used in this work

Table 3.1 consists of the datasets with the number of videos constituted by each of the datasets along with the number of users used for the annotations. The content of videos are mostly web videos and egocentric videos. Similarly the annotations are frame level score.

3.3 Extraction of Spatio-Temporal Features

In case of video sequences long range temporal information can be captured using CNN in order to recognize the characteristic frames and gives basic idea of video content. In addition to that long range representations are helpful for getting more context information. For that GoogleNet will be used for feature extraction avoiding last three layers. Given the Video Sequences V of F Frames we will get the features vectors $v_j, j \in i, .., F$. For the long term temporal features attention based mechanism will be used which will give the feature vector as w_j. Thus the final representations of the feature vector will be obtained as the concatenation of the two feature vectors as $x_j = w_j + v_j$.

3.3.1 GoogleNet

GoogleNet is the 22 layered deep learning architecture which prevails in the area of computer vision which has been developed by Google. It has performed well comparing it with its predecessors based on computational efficiency with the error rate of 6.67%. This is used in this work for spatial feature extraction from the most representative frames of the videos. To do that feature vector is extracted excluding the last three layers of GoogleNet architecture. Figure 3.2 shows the architecture of GoogleNet with pool5 layer along the inception layers shown in the figure. Visual features are taken from pool5 layer after softmax applied on the outputs.

3.3.2 Temporal Features

In case of videos sequences of the video frames are as important as the individual frames because they retains the flow of action in the video sequences and gives more contextual information about the event. Thus to capture long range features Temporal features are extracted using attention mechanism. Moreover other models like LSTM, Bi-LSTM , Graph convolution will also be investigated for their analysis. Figure 3.3 depicts the multihead self attention architecture which is used for this work where 8 head are used for attention mechanism.

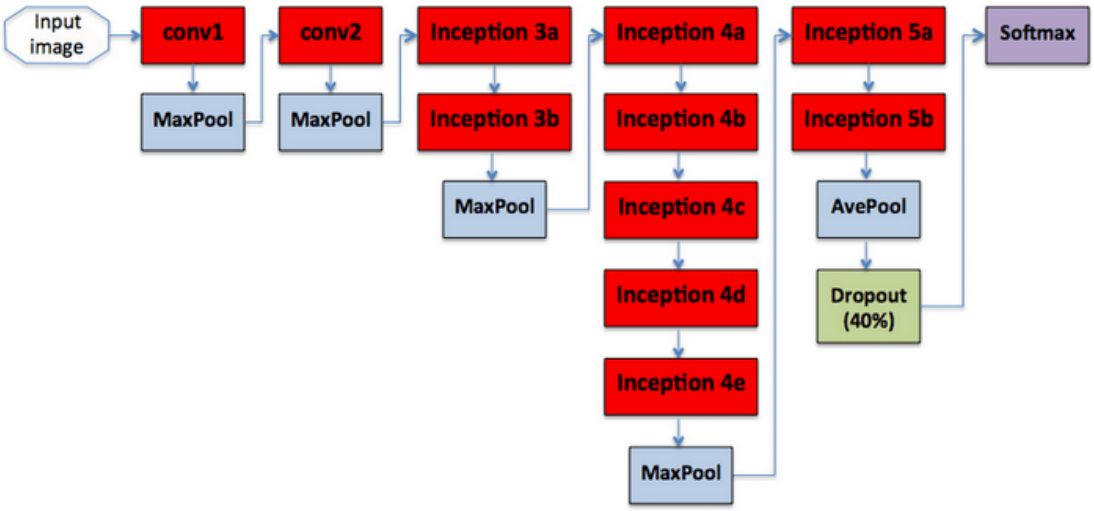


Figure 3.2: Architecture of GoogleNet [1]

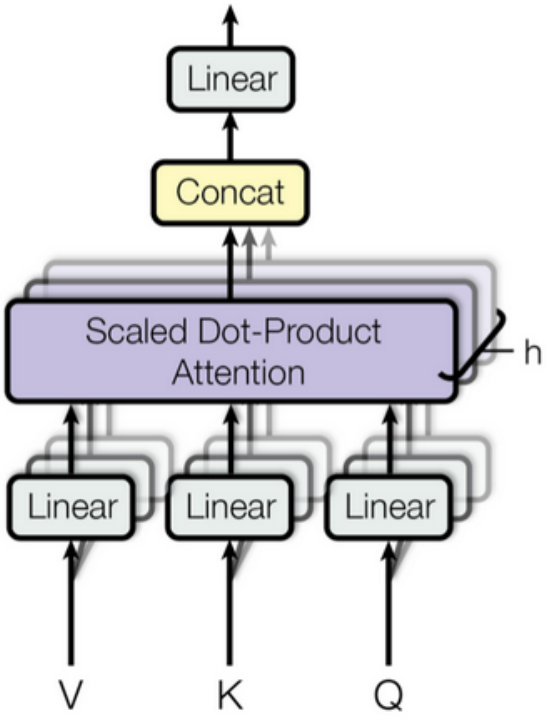


Figure 3.3: Multi-Head Attention[2]

3.4 Generation of Temporal Action Proposals

Video sequences has mostly the variable length duration that raise the concern for video summarizations when the temporal features are not taken into account that leads to problem of incomplete segmentation and irrelevant frames getting importance. In training process binary class labels will be assigned to the interest proposals. For that we will be calculating temporal Intersection over Union (tIoU) and compare with the threshold value to assign the binary labels either positive or negative. If greater that threshold positive value will be assigned and if less than threshold negative value will be assigned.

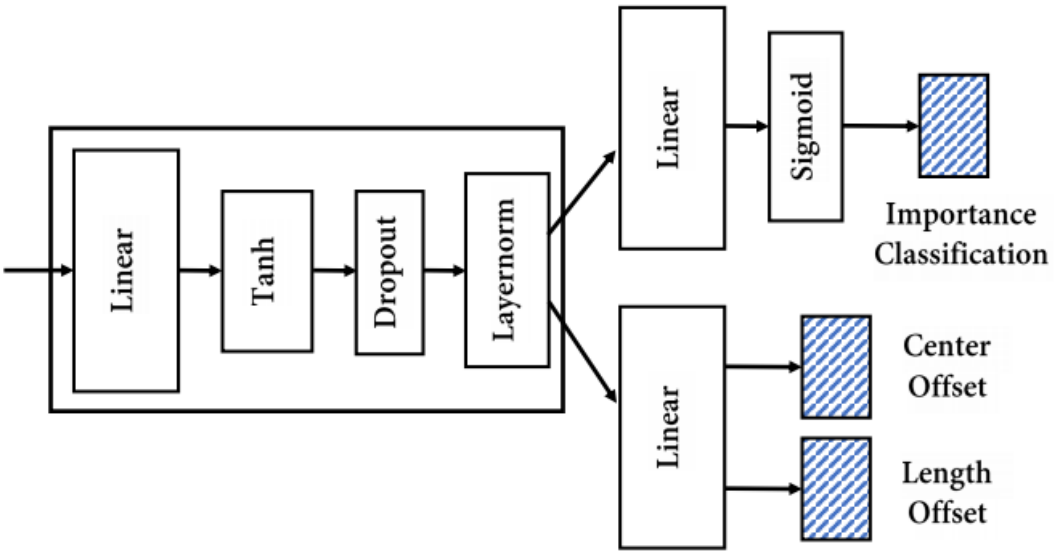


Figure 3.4: Detail Layers of Classification and Regression

3.5 Proposal Classification and Regression

Temporal Features are average pooled and then fed next module as in the Figure 3.3. It bifurcates into two different smaller module i.e Classification and Regression each of these contain FC Layers as shown in figure 3.4. Classification gives the significance score and second output gives the center and length offset.
The Loss Function for training the network is defined as L and mathematically can be expressed below equation:

$$L(p, p^*, t, t^*) = \frac{1}{N}\sum_i L_{cls}(p_i, p^*) + \frac{\lambda}{N_{pos}}\sum_i p_i^* L_{reg} \times (t_i, t_i^*) \tag{3.1}$$

where the λ is the hyper-parameter that balances the loss of classification and regression. N_{pos} denotes proposals with positive labels and N is the total labelled proposals.Similarly the p_i and p_i^* are importance score of predicted and GT respectively for the i^{th}proposals. Likewise L_{cls} is representation of cross entropy loss.
L_{reg} is the regression loss and it can be defined by the smooth absolute mean square loss function and mathematically can be written as:

$$L_{reg}(t_i, t_i^*) = \frac{1}{Q}\sum_{q=1}^{Q} L_{ismooth}(t_{iq} - t_{iq}^*) \tag{3.2}$$

$$\begin{aligned} L_{ismooth}(x) &= 0.5x^2 \quad if \quad |x| < 1 \\ &= |x| - 0.5 \quad otherwise \end{aligned} \tag{3.3}$$

These are the smooth absolute loss taken to consider the regression loss. In the equation 3.2 t_{iq} is the q^{th} loss for the element t_i. These losses are generated by comparing the predicted center offsets and length offset with that of ground truth offsets.

3.6 Selection of the Keyshots

Finally after obtaining the importance score for each of the frames and offsets implementing classification and regression module refined segments are generated which is done as testing phase of the network. To eliminate the overlapping of low confidence segments Non-Maxima Suppression (NMS) will be performed which will mitigate the redundancy and the segment with low quality. Since the importance score are assigned as the frame level, shots are need to be identified round the frames with high importance score. For the same Kernel Temporal Segmentation (KTS) algorithm will be implemented to get key shots which will take the consideration of both importance score of frames as well as the temporal features.

3.7 Evaluation Metrics

For the evaluation of the result obtained from the experiments on the datasets, F- Measure will be used as the quantitative metrics. To evaluate how the user summaries and summaries created by the model, F-Score is computed which is the harmonic mean of recall and precision that gives the quantitative measurement of the model efficiency. F-score is calculated in percentages. The F Measure is calculated by the following equations:

$$F - Measure = \frac{2P_i * R_i}{P_i + R_i} \tag{3.4}$$

Where $P_i$is the Precision and calculated by:

$$P_i = \frac{length(gs_i \cap gt_i))}{length(gs_i)} \tag{3.5}$$

Similarly $R_i$is recall and it is calculated by:

$$R_i = \frac{length(gs_i \cap gt_i))}{length(gt_i)} \tag{3.6}$$

gs_i is the generated summaries for i^{th} summary and gt_i is the annotated summary. The F measure is the quantitative measure for evaluating the results obtained from the experiments on the two datasets viz: SumMe and TVSum. True and false positives and false negatives for the F-score are considered as similarity in the model summaries and ground truth which is calculated per frame basis. It can be seen in figure 3.5.

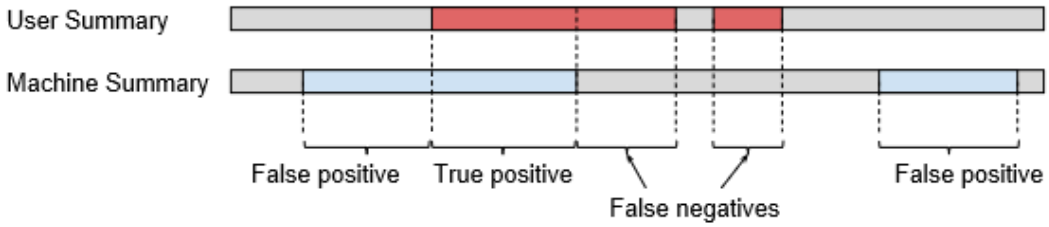
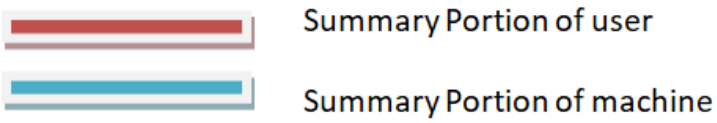


Figure 3.5: True Positive, False Positive and False Negative representation on per frame basis between ground truth and generated summary by the model



3.8 Tools used

- Python Language
- HDF Compass
- Google Colab
- PyTorch

3.9 Implementation

3.9.1 Dataset Preparation

Dataset used for this work are TVSum, SumMe, OVP, and YouTube. Pre processed dataset is available[11]. It has preprocessed data with the following data structure:

- features : 2D-array with shape(n_steps,feature-dimension)
- gtscore : 1D-array with shape (n_steps), stores ground truth importance score (used for training, e.g. regression loss)
- user_summary : 2D-array with shape (num_users, n_frames), each row is a binary vector (used for test)
- change_points : 2D-array with shape (num_segments, 2), each row stores indices of a segment
- n_frame_per_seg : 1D-array with shape (num_segments), indicates number of frames in each segment
- n_frames : number of frames in original video
- picks : positions of subsampled frames in original video
- gtsummary : 1D-array with shape (n_steps), ground truth summary provided by user (used for training, e.g. maximum likelihood)
- n_steps : number of subsampled frames
- video_name : original video name, only available for SumMe dataset

The datasets contain these attributes which have significant role for the training and testing model. Since these datasets contain labeled ground truth it will make the evaluation of the model more reliable and validated. The figure 3.6 and 3.7 show the visualization of the ground truth frame importance score and the ground truth summary of one of the video from SumMe dataset. GT score and GT Summary are used for training the model.
X-axis is the frame indices and Y-axis is the score ranging from 0-1.

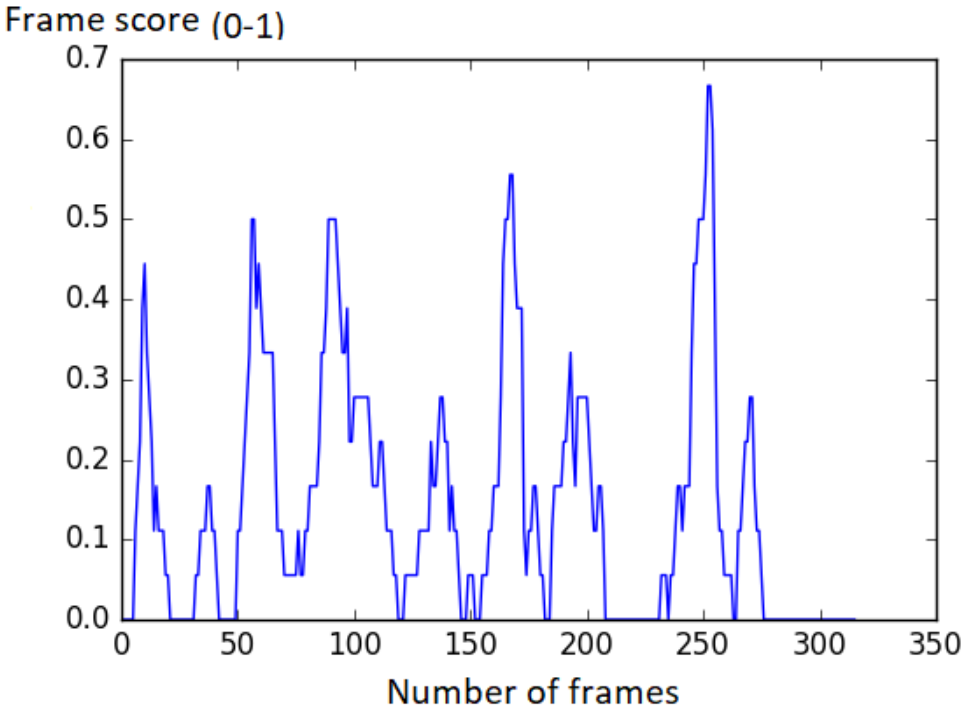


Figure 3.6: GT Score of the video frame importance

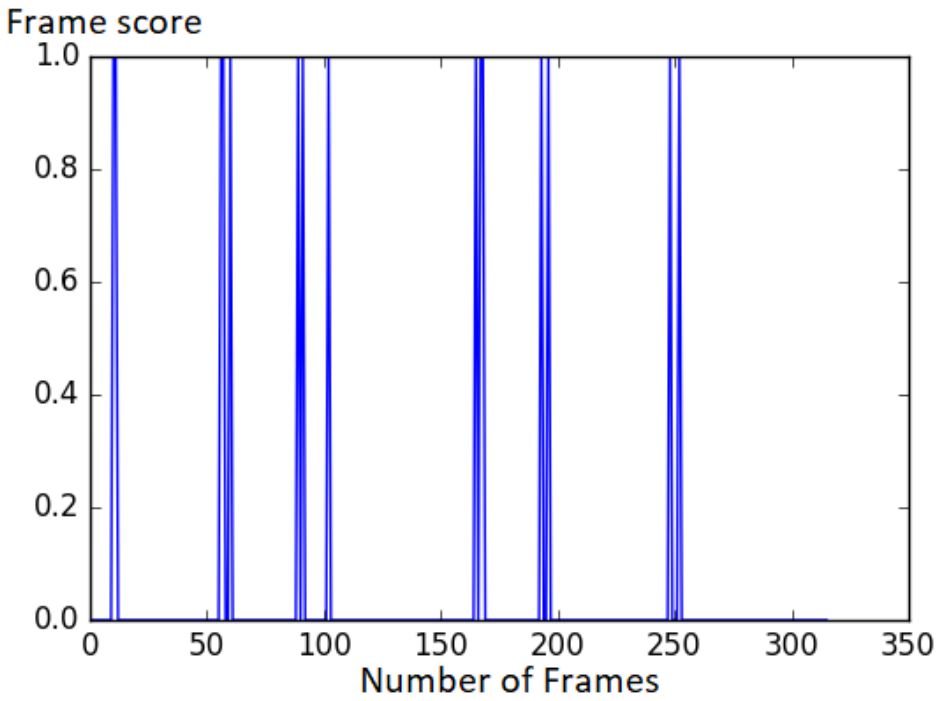


Figure 3.7: GT summary/Segment annotated by the User

3.9.2 Preparation of the Ground Truth for Training and Evaluation

To train the model frame level importance score annotated in the dataset is used while binary keyshot summaries is used to evaluate the model. For that dataset needs to be converted into keyshot summaries before training and evaluation. Annotation in SumMe has both keyshot level as well as frame level which can interchanged using averaging the framelevel scores. Similarly TVSum dataset comes with frame level score and it needs to be converted into shot level scoring by taking the average over the frames using the equation 4.1.

$$s_i = \frac{1}{l_i} \sum_{a=1}^{l_i} y_{i,a} \tag{3.7}$$

where l_i is the that particular i^{th} shot and s_i is the shot level score for the y_a^{th} frame.

This is done to do the direct comparison of the training and testing ground truth which is even used by [11][29]. The Prepossessed datasets are publicly available by [11] which includess CNN features and frame level importance score along with the change points generated by KTS and Keyshots. Features obtained from pretrained on ImageNet networks possess the dimensions of 1024 and respective features are extracted using GoogleNet from pool5 layer. Following works from literature [8], dataset videos which are of 24fps are down-sampled to 2 fps. This is done to alleviate the computational complexities and reduce the redundancy.

3.9.3 Experimental setups

There are two basic dataset mainly used in this work i.e TVSum and SumMe. TVSum contains 50 videos mostly of web videos while SumMe contains 25 videos of similar genre. Moreover these datasets are augmented with other two datasets viz: OVP and YouTube. Each of these videos are of 30fps so following the works in the literature and equivalent evaluation with other methods the videos are down-sampled to 2fps which also reduce the complexity of computation and take care of temporal redundancy. There are three different experimental setups that are

Canonical , Augmented and Transfer setting of dataset. Moreover the environment to run the python code is google colab. Pytorch library has been used for the implementation. The model is trained on google colab for 300 epochs with Adam optimizers and learning rate of $5x10^{-5}$.

Training Parameters	**Values**
Epochs	300
Learning Rate	$5x10^{-5}$
Optimizer	Adam
Drop Out	0.5
weight_decay	$1x10^{-5}$
num_feature	1024
anchor_scales	[4, 8, 16, 32]

Table 3.2: Training Parameters

3.10 Training of the Model

There are basically two modules which are needed to be trained end to end. They are temporal proposal networks and the classification and regression module. Spatio-Temporal Features extraction module gives the long range features on which the proposals are generated using anchor based mechanism. These proposals are learnt from the ground truth and then adjust themselves with offsets. Loss function calculates the total loss and back propagates to optimize the weights accordingly. During the training of the proposal notworks binary labels i.e positive and negative labels are alloted to the generated proposals. Ratio of positive to negative labels assignment is 1:3 as there will be more number of negative samples which might create class imbalance. In order to assign the labels to the proposals, Intersection of Union(IOU) are calculated and compared it with that of ground truth. Threshold for the IOU is selected as 0.6 that means if the IOU between generated proposals and ground truth is greater than 0.6 positive label is assigned and if it is less than 0.6 and greater than 0.3 negative labels is assigned. For the samples falling

between 0 - 0.3 are considered as incomplete and unimportant proposals. Later in classification and regression module multi task loss is calculated using equation 3.1 and learning process continues for each epoch.

3.10.1 Training on Canonical Setting

Datasets(TVSum and SumMe) are divided into 5 random splits. Model is trained using 80% of the data while remaining 20% of the data are for evaluation in canonical setting. The result obtained from this setup is tabulated in given table : It can be observed that model has been converging as the epoch number increases. In the figure 3.8 and figure 3.9 after the epoch 11 there is drastic change in loss and from there loss has been decreasing gradually with the epoch number. From the table 3.3 we can validate that value of loss is similar with every splits that shows that model has not over-fitted the data for canonical settings. Even though for the SumMe dataset there are lots of variation seen in the data per the splits.

Dataset splits	TVSum	SumMe
split 0	0.4673	0.5344
split 1	0.5558	0.5407
split 2	0.4936	0.5592
split 3	0.4730	0.5633
split 4	0.4830	0.5729

Table 3.3: Total Loss on each splits on TVSum and SumMe for Canonical Setting

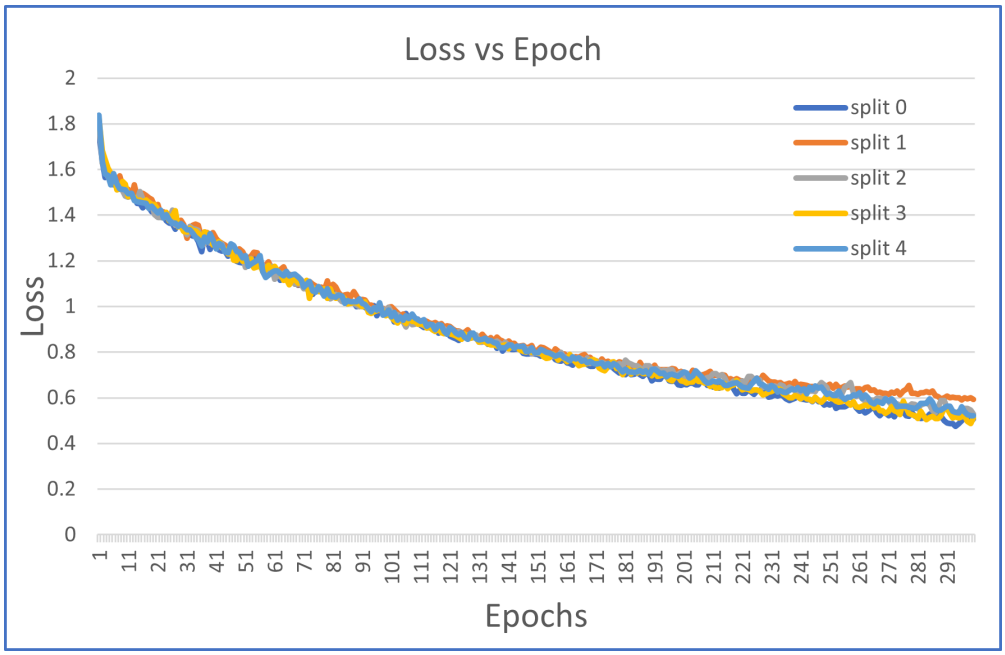


Figure 3.8: Loss Curve on Canonical TVSumm Dataset

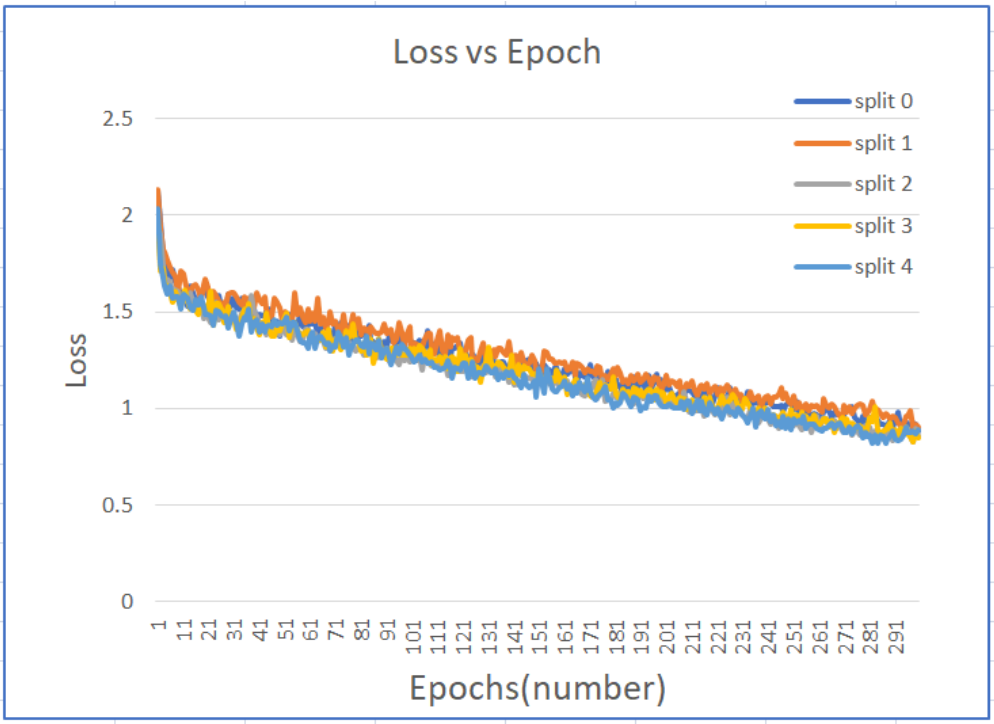


Figure 3.9: Loss Curve on Canonical SumMe Dataset

3.10.2 Training on Augmented Setting

In this setup again 5 random splits is taken for cross validation but in addition for training 80% of the data is augmented with other three dataset are used. For Example, to train SumMe in the augmented setting all samples from TVSum, OVP, and Youtube and 80% of the SumMe are taken as training sample while remaining 20% is used as evaluation set. Same goes for TVSum. Figure 3.10 and 3.11 show that there is smooth change in loss curve as the epoch number increases. There is now earlier convergence or late convergence observed because in the augmented setting all four datasets have been used for the training that makes loss curve smooth for every epochs. Convergence can be observed after the certain epochs as shown in the fig 3.10 and fig 3.11.

Dataset splits	TVSum	SumMe
split 0	0.3752	0.3709
split 1	0.4095	0.4202
split 2	0.3859	0.3947
split 3	0.3814	0.3738
split 4	0.4021	0.4178

Table 3.4: Total Loss on each splits on TVSum and SumMe for Augmented Setting

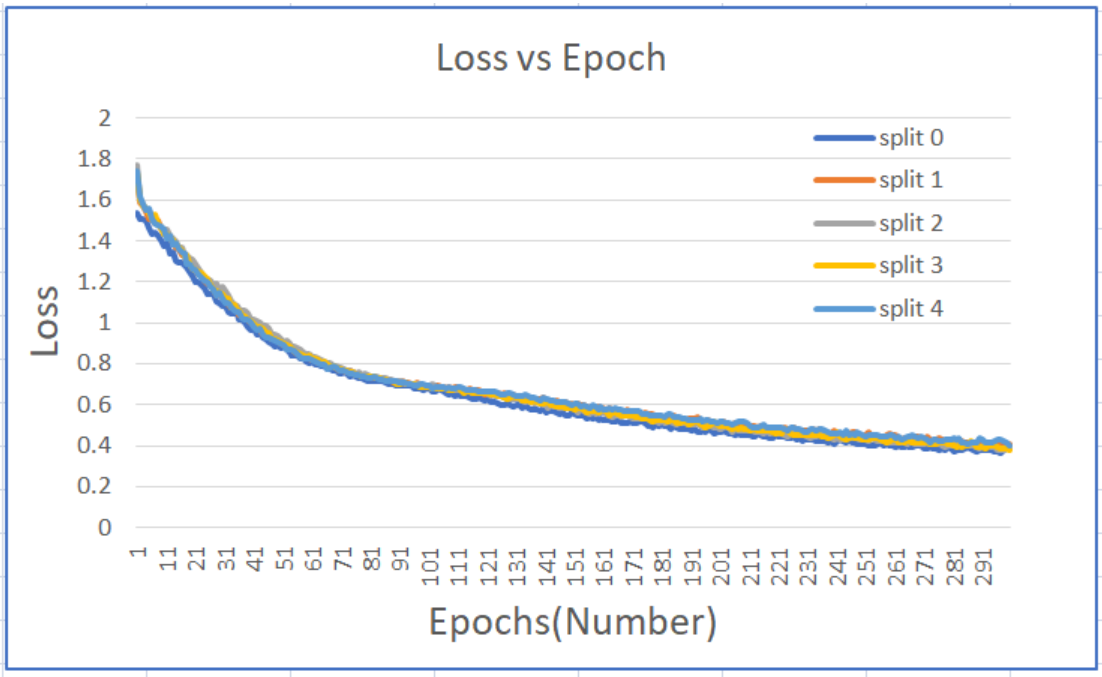


Figure 3.10: Loss Curve on Augmented TVSumm Dataset

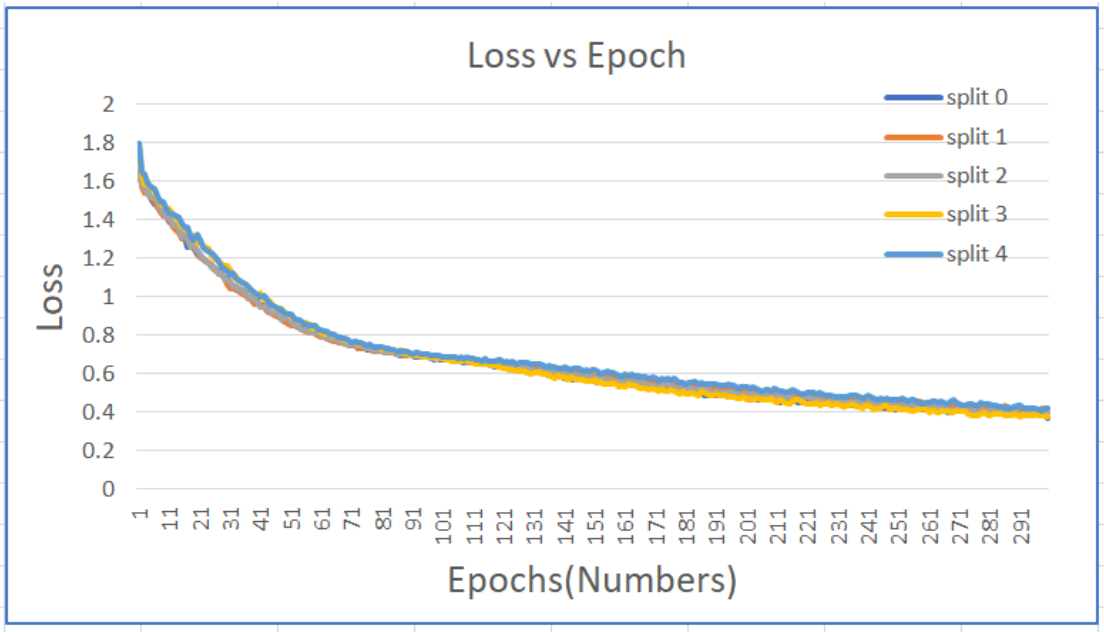


Figure 3.11: Loss Curve on Augmented SumMe Dataset

3.10.3 Training on Transfer Setting

In this setting, Model is trained using three of the datasets and rest of one dataset is used for testing the modelIn this setting, Model is trained using three of the datasets and rest of one dataset is used for testing the model. For Example, If i take SumMe as evalutaion dataset then other three viz. TVSum, OVP and Youtube are used as training set. Similarly when SumMe , OVP, YouTube, are used as training set, TVSum is used as evaluation set. The Table 3.5 is loss table for the dataset TVSum and SumMe on the different dataset splits. Each split is trained on 300 epochs. Training on transfer setting have similar loss curve as that of canonical with better loss since it makes use of all four datasets for trianing.

Dataset splits	TVSum	SumMe
split 0	0.4009	0.3497
split 1	0.4054	0.3665
split 2	0.4025	0.3854
split 3	0.3950	0.3699
split 4	0.3912	0.3882

Table 3.5: Total Loss on each splits on TVSum and SumMe for Canonical Setting

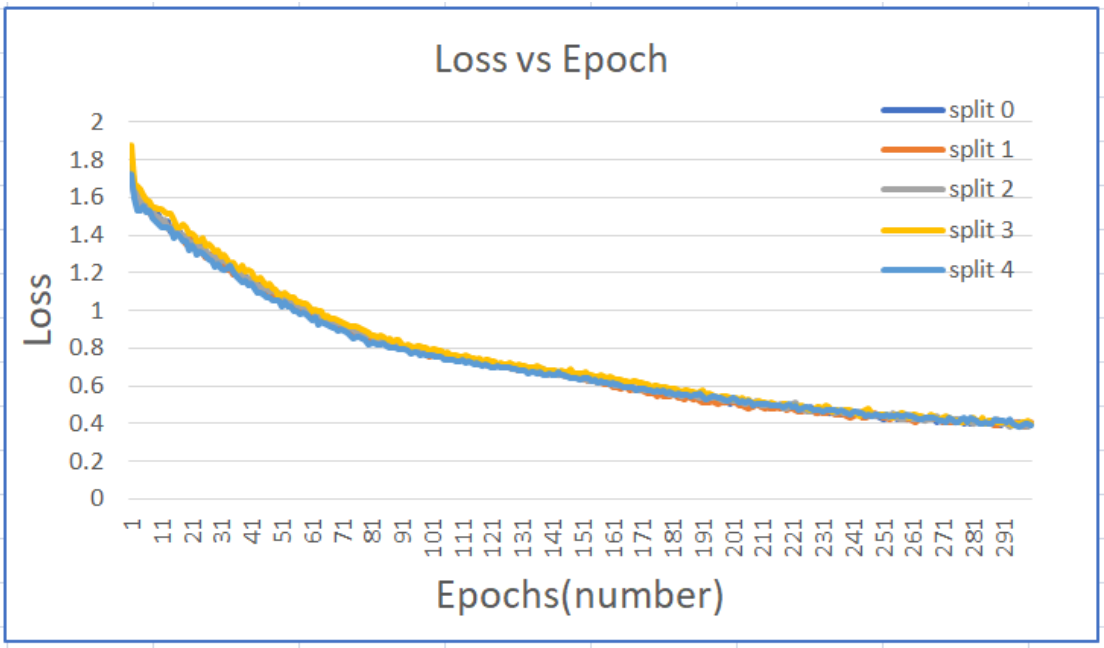


Figure 3.12: Loss Curve for Transfer TVSumm Dataset

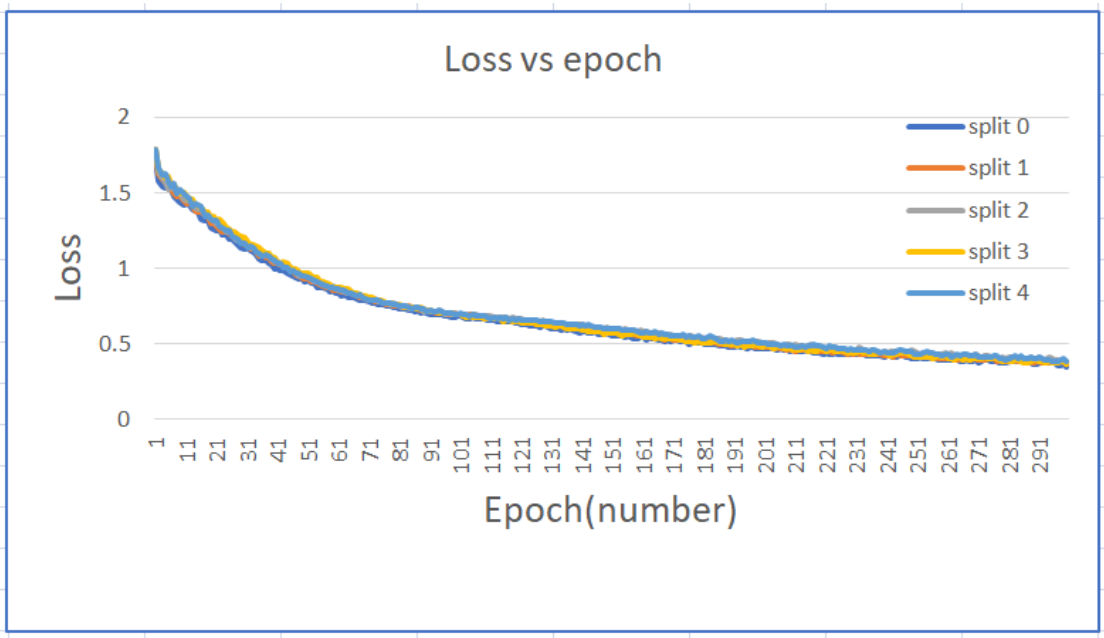


Figure 3.13: Loss Curve for Transfer SumMe Dataset

The model has been trained on three different setting with splitting the datasets into five splits for the cross validation. From the loss curve and tables it can be inferred that there model has been trained properly with the available dataset with three different experimental setups. After the training there are positive and negative labeled proposals generated from the long range of features based on the calculation of IOU. Samples of positive and negatives are classified in 1:3 ration to eliminate the class imbalance problem that might occur because of high number of negative samples. There are reason behind labelling the samples of proposals with the anchor based mechanism. Firstly it helps the model to choose the consecutive frames of the high IOU value compared to the ground truth as well as ignores selecting the irrelevant portions. Secondly it also handles the irrelevant portions and segments by labeling it as negative samples for those samples which have poor overlaps with ground truth.

Total number of the 1024 dimensions of visual features are extracted using GoogleNet trained on imagenet from pool5 layers. In the addition to that 8 heads attention mechanism are used with hidden layers of 128 layers. Morover NMS threshold has been set to 0.5 and the hyper parameter is 1. These are the parameters for the training of the model.

3.11 Evaluation of the Model

Evaluation is done with test datasets on the trained model for every setting of the dataset. Each of the results are shown below as tables and graphs for each of the set ups. Each tables shown the F-Score measured with respect to the user summaries and the generated summaries on each splits of the dataset for different set ups. And Mean value for every splits has been taken as a reference for that particular set up. Similarly graphs show the curve between F-Score on every splits for 300 epochs which show the as epochs increase the value of F-Score also increased.

3.11.1 Evaluation on Canonical Settings

In the canonical setting 20% of each dataset is used for the evaluation purpose. It is evaluated on the model trained on 80% of the dataset. F-score has been calculated for each splits as shown in the Table 3.6 on TVSum and SumMe datasets. It is seen that similar F-score on every splits. Moreover the figure 3.14 and 3.15 are the F-score vs epoch curves for TVSum and SumMe datasets respectively for each of each splits of datasets. It can be observed that F-score becomes better as the number of epochs increases which shows that the result shows better as it gets evaluated with the more data with more number of epochs.

Dataset Splits	TVSum	SumMe
split 0	0.6242	0.5344
split 1	0.5871	0.5407
split 2	0.6436	0.5592
split 3	0.6152	0.5633
split 4	0.6404	0.5729

Table 3.6: F-Score on each split on TVSum and SumMe for Canonical Setting

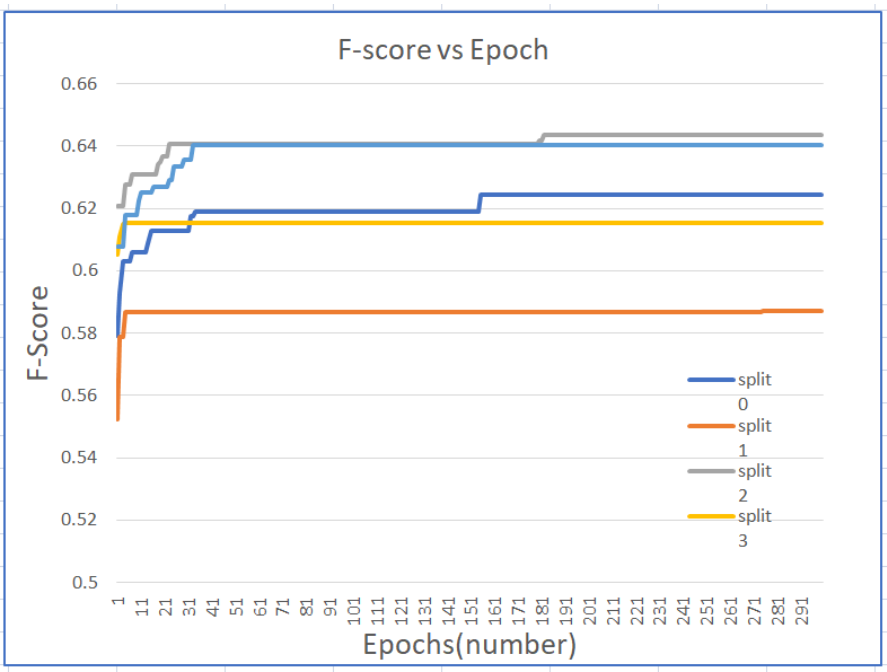


Figure 3.14: F-Score Curve for Canonical TVSumm Dataset

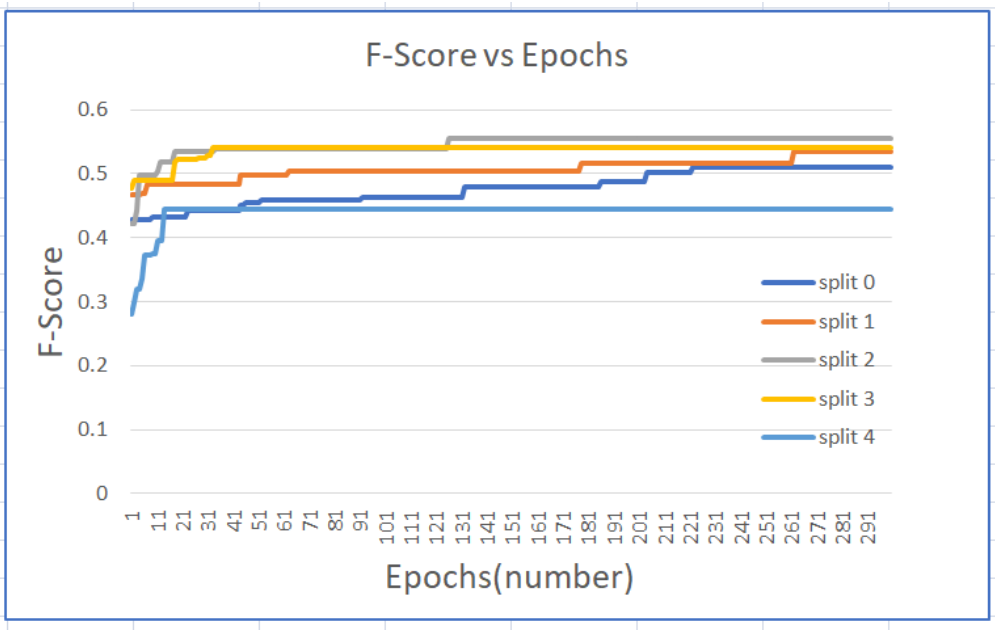


Figure 3.15: F-Score Curves for canonical SumMe Dataset

3.11.2 Evaluation on Augmented Setting

In this experimental setting the 80% of three dataset is used for training and only 20% remaining data is used from one of the dataset for the evaluation. So there are two evaluation one for TVSum and another for SumMe dataset. Table 3.7 shows that evaluation metrics F-score for each of the datasets and it can be seen that data are better for the split 4 in TVSum while for SumMe split 3 gives better result. This show that the data distribution and number of data has more influence on the result which shown the model learns better with more number of data. Moreover Figure 3.16 and 3.17 shows the curve for every epoch and can be inferred that after certain epoch there is no any improvement in the F-score value as the model reaches saturation.

Dataset splits	TVSum	SumMe
split 0	0.6381	0.4726
split 1	0.6083	0.4731
split 2	0.6437	0.5333
split 3	0.6347	0.5659
split 4	0.6487	0.4535

Table 3.7: F-Score on each splits on TVSum and SumMe for Augmented Setting

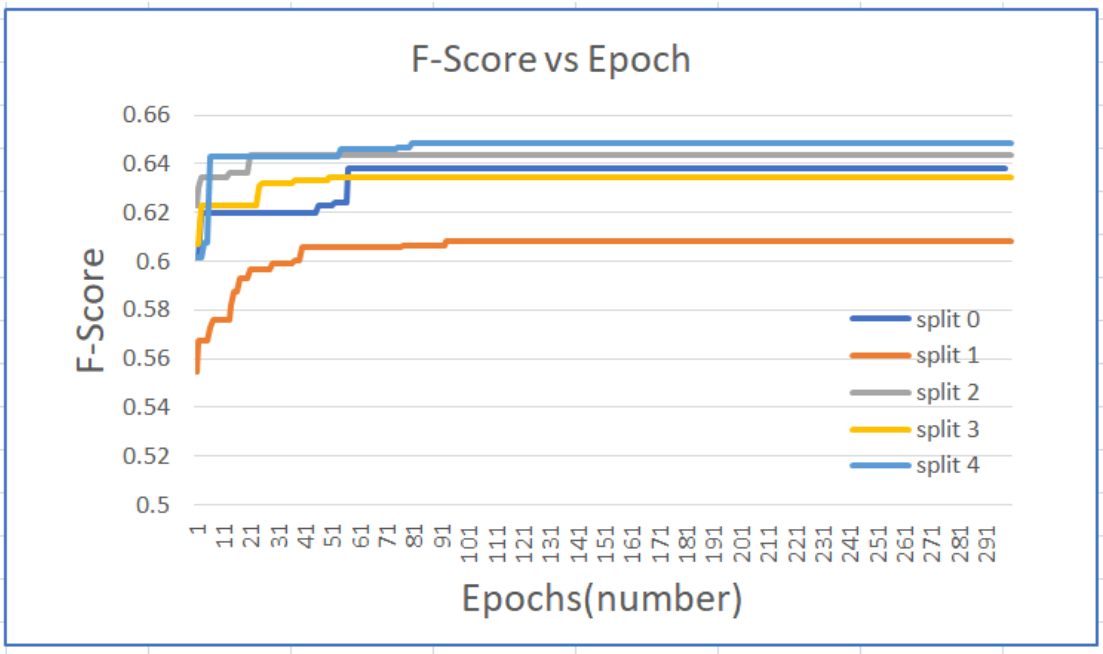


Figure 3.16: F-Score Curve for Augmented TVSumm Dataset

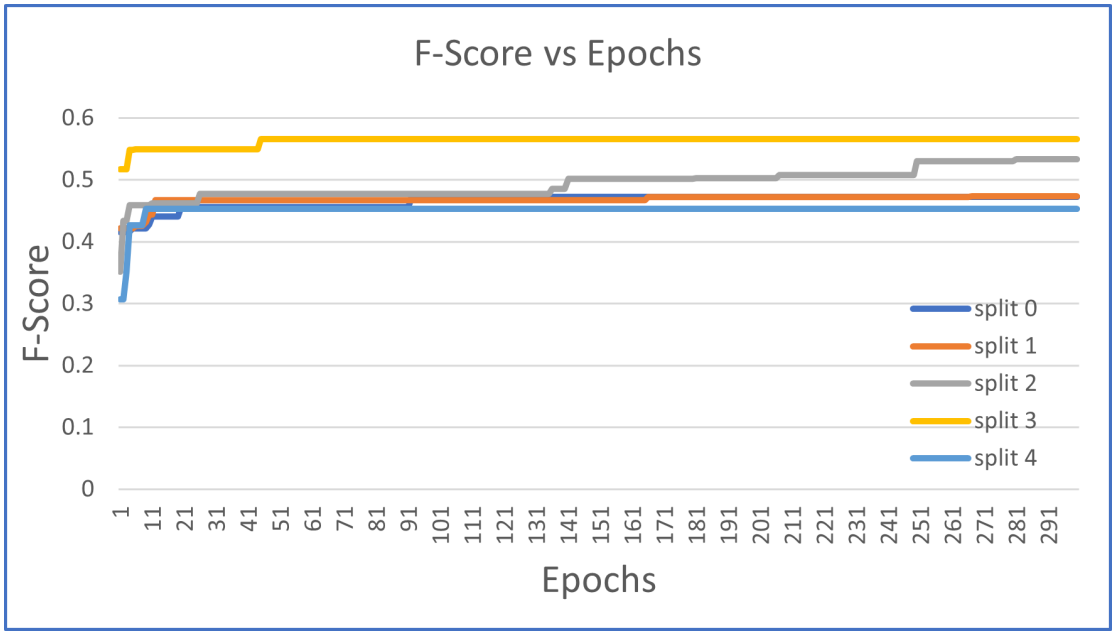


Figure 3.17: F-Score Curve for Augmented SumMe Dataset

3.11.3 Evaluation on Transfer Setting

In the transfer setting of dataset, three of the datasets are used as training and one of remaining dataset is used for evaluation purpose. Table 3.8 shows the evaluation value as F-Score for every splits of the datasets. The value signifies that the result is not much better as compared to the canonical and augmented settings for any of the datasets. This shows that transfer setting, model has difficult learning from data because of the variation in datasets. Using different dataset for training tunes the model in different ways that makes the model learns adverse that is why the F-score is low as compared to previous settings.

Dataset splits	TVSum	SumMe
split 0	0.5945	0.4790
split 1	0.5951	0.4578
split 2	0.5869	0.4666
split 3	0.5954	0.4525
split 4	0.5888	0.4605

Table 3.8: F-Score on each splits on TVSum and SumMe for Transfer Setting

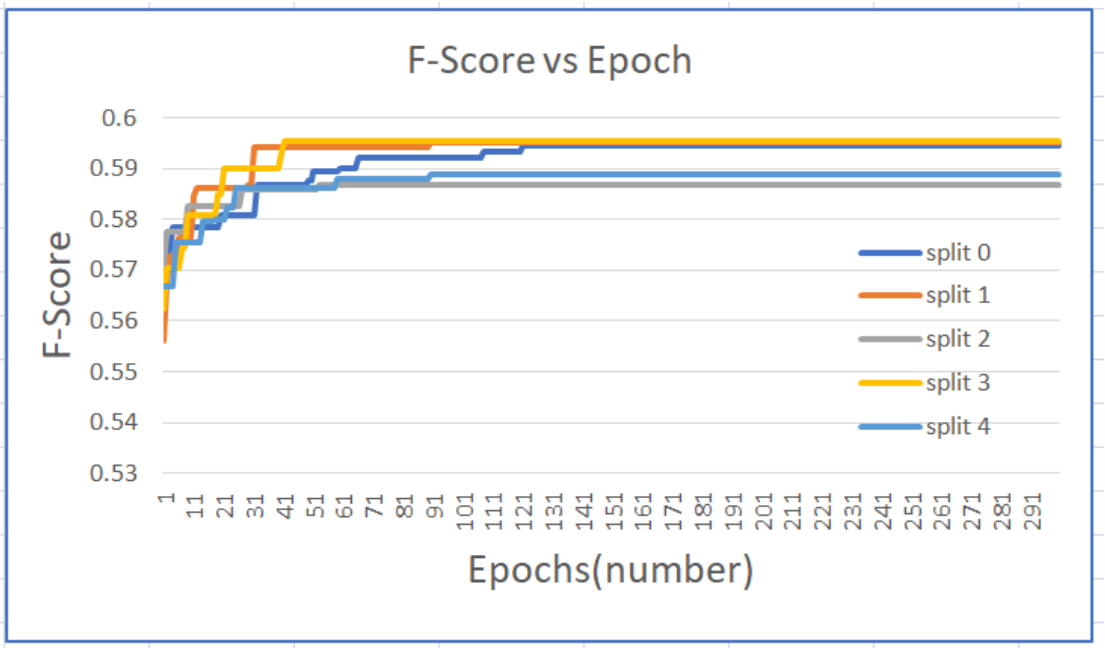


Figure 3.18: F-Score Curve on Transfer TVSumm Dataset

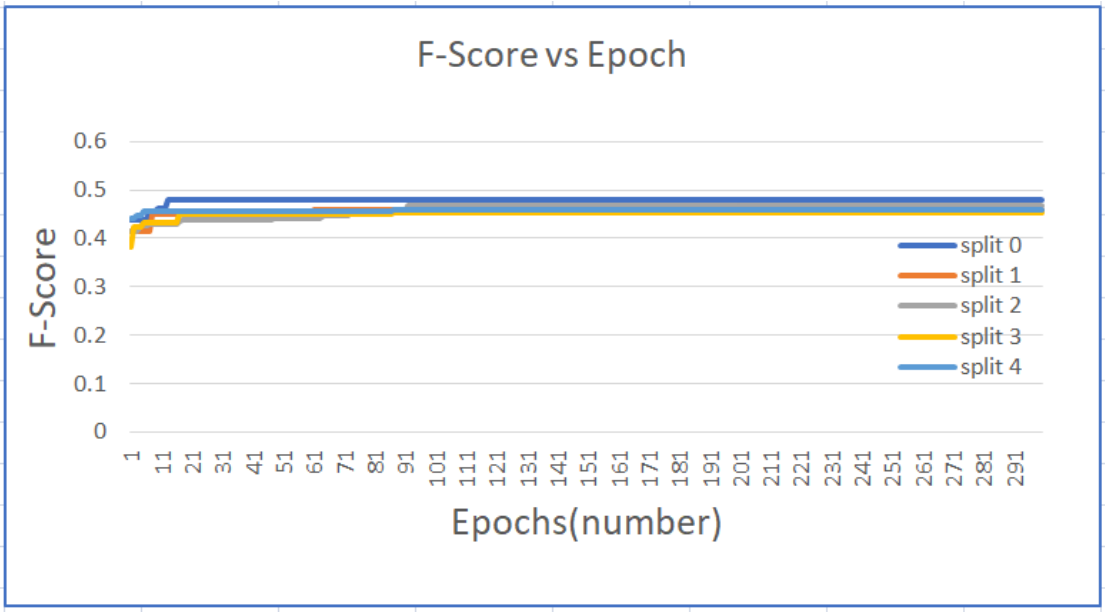


Figure 3.19: F-Score Curve on Transfer SumMe Dataset

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Results

Temporal Features have been extracted using self attention for the model while it has also been investigated with other temporal models like LSTM and BiLSTM to find out the effectiveness of the attention mechanism over others. Table 4.2 shows the F-Score of the each models based on the dataset while Figure 4.1 and 4.2 show the graphical representations of F-score on each split of TVSumm and SumMe dataset respectively. It signifies that attention mechanism is working better for TVSumm Dataset while for SumMe dataset all models have similar performance.

From the Table 4.1 it can be inferred that model performs well on Augmented Setting on TVsumm dataset while on SumMe dataset model performs well on canonical set up as compared with other set up. This can also be validated by looking at loss curve which shows that loss is minimum for the TVSum dataset in Augmented Setup.

Datasets	Canonical	Augmented	Transfer
TVSum	64.36	**64.87**	59.54
SumMe	**57.29**	56.59	47.90

Table 4.1: F-Score comparisons on TVSum and SumMe for different Settings

Datasets	LSTM	BiLSTM	Attention
TVSum	60.01	58.90	**64.36**
SumMe	51.21	52.56	**57.29**

Table 4.2: F-Score comparisons on TVSum and SumMe for different Temporal Models

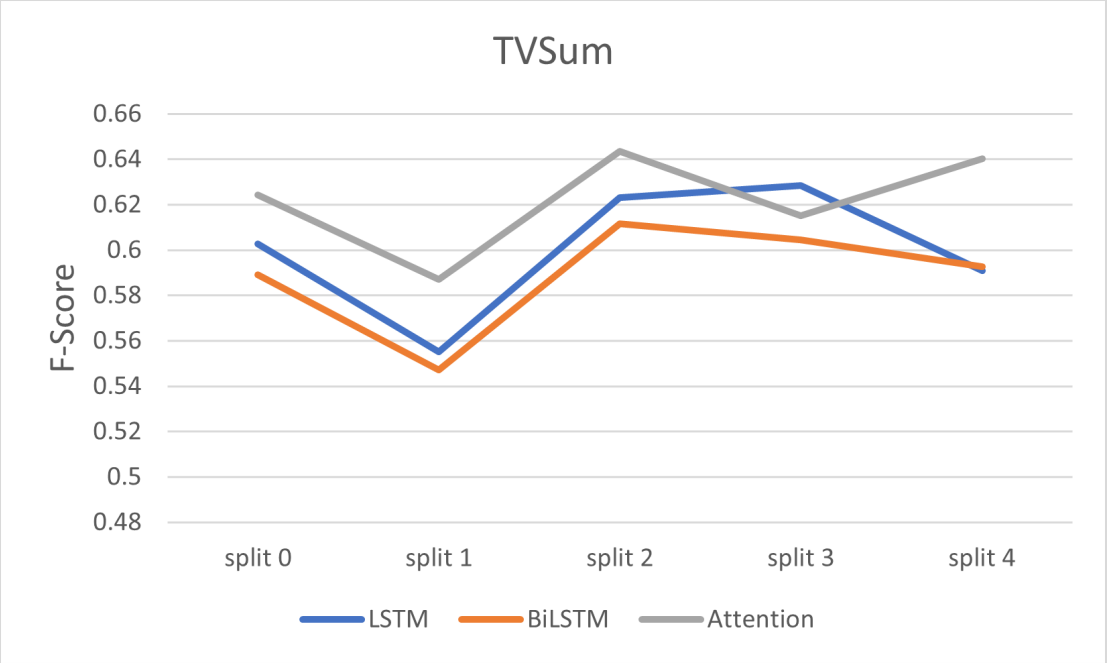


Figure 4.1: Comparison of Performance of LSTM,BiLSTM and Attention models on TVSumm Dataset

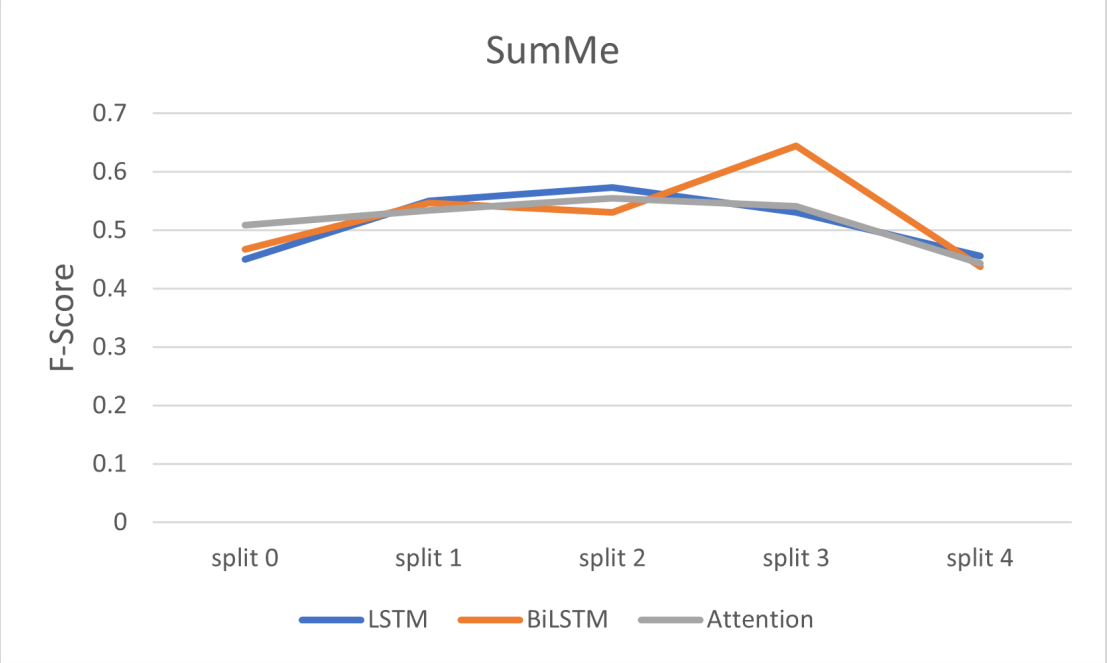


Figure 4.2: Comparison of Performance of LSTM,BiLSTM and Attention models on SumMe Dataset

4.1.1 Frame Comparison with different models

Temporal features selection models are most important when it comes to capture sequence of the videos. Here self attention model has been used primarily for capturing temporal features along with the LSTM and BiLSTM. It can be observed that each of the models have their own capabilities to select important frames based on their learning from ground truth. Provided that the F-score for the attention mechanism leads quantitatively we can verify it from the frame comparison as well. Figure 4.3 shows the sample of selected frames using self attention mechanism model. Figure 4.4 shows the sample of selected frames using LSTM model and Figure 4.5 shows the sample of selected frame using BiLSTM model as the temporal features selection models. It can be seen that there is huge difference in the frame selection on each of the models.

Figure 4.3: sample of selected Frames using self attention model

Figure 4.4: sample of selected Frames using LSTM model

Figure 4.5: sample of selected Frames using BiLSTM model

4.1.2 Diversity in the generated summaries

One of features of good summaries is that it should include diverse content which can be measured diversity score measurement.The degree of diversity of a generated summary is evaluated by measuring the dissimilarity among the selected frames in the feature space [11]. The diversity score is used to evaluate the diversity in the summaries generated by this algorithm on SumMe and TVSum dataset. Table 4.3 shows comparison of the diversity score of dppLSTM and DR-DSN.

Datasets	dppLSTM [29]	DR-DSN [11]	Proposed Method
SumMe	0.591	0.594	**0.6549**
TVSumm	0.463	0.464	**0.4748**

Table 4.3: Diversity Score comparison

4.1.3 Analysis of Recall vs Proposals

In order to ensure that there is high recall with respect to the ground truth, recall vs refined proposals can be represented as in Figure 4.6 which shows the recall values based on the number of proposals for each of the models used for comparison. This is for TVSum dataset which contains minimum of 83 sec length videos and 166 frames.In this method 4 proposals are being generated on each temporal locations making it total of 664 proposals. Figure 4.6 shows that it can achieve about 95% of the recall.

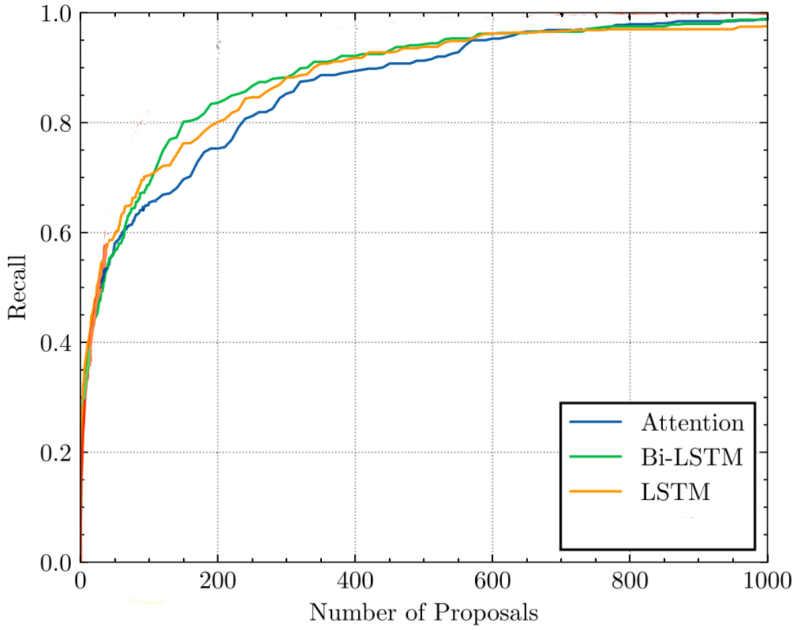


Figure 4.6: Recall vs Proposal for TVSumm Dataset

4.1.4 Comparison with various video summarization methods

Results obtained from this work has been compared with other methods of the video summarization which has utilized deep learning methods and tested on the SumMe and TVSum dataset. It has been compared with those methods which has similar methodolgies and used deep learning approaches to get summarised results. Some of the methods are vsLSTM [29], dppLSTM [29], DR-DSN [11], SUM-GAN [10], VASNET [8], AVS [33]. Table 4.4 signifies that there is significant improvement in this works compared to other methods. It has also been compared with the baseline model like random summaries, uniform sampling and clustering methods. From the F-score comparison it can be inferred that summaries generated by this method is far better than that of random and uniform sampling summaries. Moreover to demonstrate how effectively this proposed methods learned from data labeled by users and human performance is compared, for that we calculate F-score of all users and ground truth. In table 4.4 it can be observed this proposed method are better than human performance for TVSumm while for SumMe it is very close. This shows that our method has learned enough from the user annotations and summaries created by are similar to that human would have generated.

Methods	SumMe	TVSumm
Random Selection [24]	-	32
Uniform Sampling [24]	15	36
Clustering	17.5	39
vsLSTM [29]	37.6	54.2
dppLSTM [29]	38.6	54.7
SUM-GAN [10]	41.7	56.3
DR-DSN [11]	42.1	58.1
AVS [33]	43.9	59.4
VASNET [8]	49.71	61.42
Human [24]	64.2	63.7
Proposed Method	**57.29**	**64.36**

Table 4.4: F-Score comparison of other methods and this work

4.1.5 Validation of the Result

In order to compare this result with the human summaries, it is necessary to validate the human consistency among the participants of the crowd sourcing. Since the dataset is prepared by crowd sourcing.Human consistency of selecting the frames similar to other users f-score among the human selection is calculated. We use similar consistency measurement that evaluate the model summaries performance[9]. The SumMe dataset has a mean of F=0.31(min.0.18,max. 0.51) [25] while this work is able to get higher than this value.

$$\bar{F}_i = \frac{1}{N-1} \sum_{j=1, j\neq i}^{N} 2\frac{p_{ij} r_{ij}}{p_{ij} + r_{ij}},$$

where N is the number of human subjects, p_ij is the precision and r_ij the recall of human selection i using selection j as ground truth.[25]
Additionally, Cronbach alpha is calculated in order to validate the psychometric test as standard measure which gives the reliability value for the dataset result. It is defined as

$$\alpha = \frac{Nr}{1 + (N-1)r} \tag{4.1}$$

where r is the mean pairwise correlation between all human selections. The mean value of the dataset is $\alpha = 0.74(min.0.21, max.0.94)$. In the reality α is around 0.9, while α¿0.7 is the minimum for a good test related to human test[36]. where r denotes mean pairwise correlation between all human selections. The dataset has a mean of $\alpha = 0.74(min.0.21, max.0.94)$. It is applicable for both the dataset like TVSum which also has Cronbach alpha of higher value than 0.7

where r is the mean pairwise correlation between all human selections. The dataset has a mean of $\alpha = 0.74(min.0.21, max.0.94)$. Ideally α is around 0.9, while α¿0.7 is the minimum for a good test [36]. It is applicable for both the dataset like TVSum which also has Cronbach alpha of higher value than 0.81

It is applicable for both the dataset like TVSum which also has Cronbach alpha of higher value than 0.7 which shows the relaibility of the dataset.

4.2 Ablation Study

4.2.1 Influence of the average pooling layer(temporal)

In this model average pooling layer has been implemented in order to handle the variable length of the proposals. This layer has significant influence in the following classification module as well as in the regression module so in order to investigate the significance of this layer. Table 5.5 shows the F-Score variation because of the pooling layer presence and absence on two different datasets. It can be inferred from the table that with pooling layer performance of the model is better as compared that of without pooling layers.

Pooling Layer	SumMe			TVSum		
	Canonical	Augmented	Transfer	Canonical	Augmented	Transfer
×	51.4	52.3	44.9	61.2	61.9	56.7
✓	57.2	56.59	47.90	64.36	64.87	59.54

Table 4.5: Effect of with and without average pooling layer by showing F-Score(%) comparison on TVSumm and SumMe Datasets

4.2.2 Analysis of NMS Threshold

NMS thresholds has been used for the removal of redundant and low quality proposals from the output of classification and regression section which signifies that it directly affects the performance. High qualities proposals can be refined when higher threshold is chosen retaining low qualities proposals. Thus it is necessary to analyze the influence of the NMS threshold value in the model. As shown in Figure 4.8 and Figure 4.9 the value of F-Score corresponding of the different threshold values of Non-Maximum Supression(NMS) on TVSum and SumMe respectively. It can be observed that as threshold increases corresponding f-score increases and after nms 0.5 value it starts decreasing this shows that changes in the value of threshold directly influence the value of F-score thus menthod performance. The default NMS threshold chosen is 0.5.

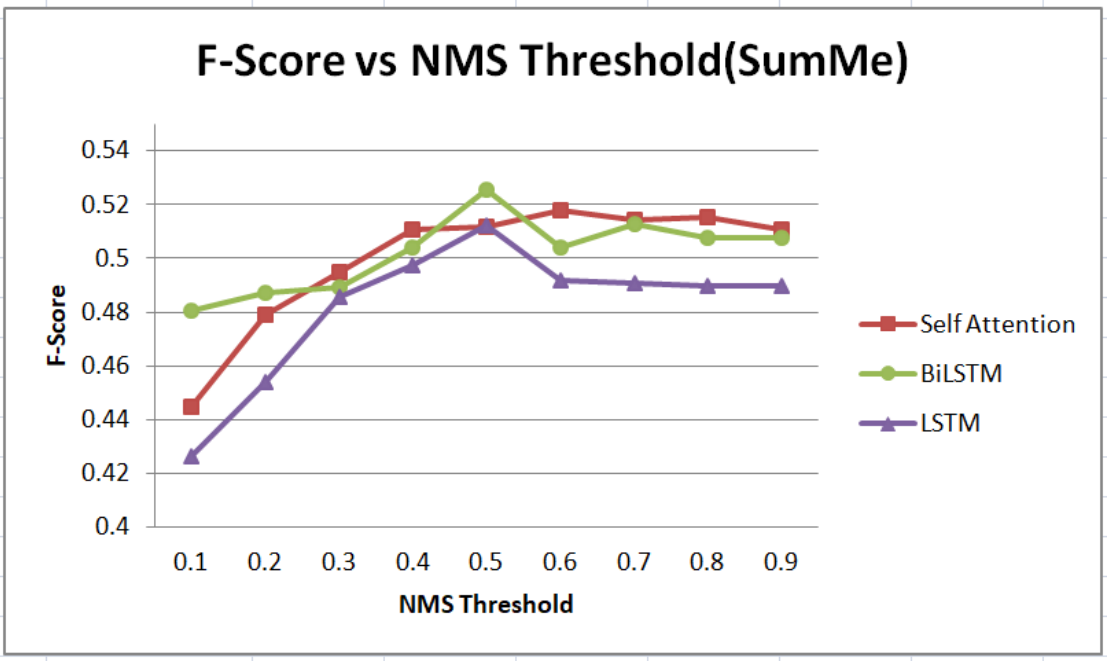


Figure 4.7: NMS Threshold analysis on SumMe dataset (Default is set at 0.5)

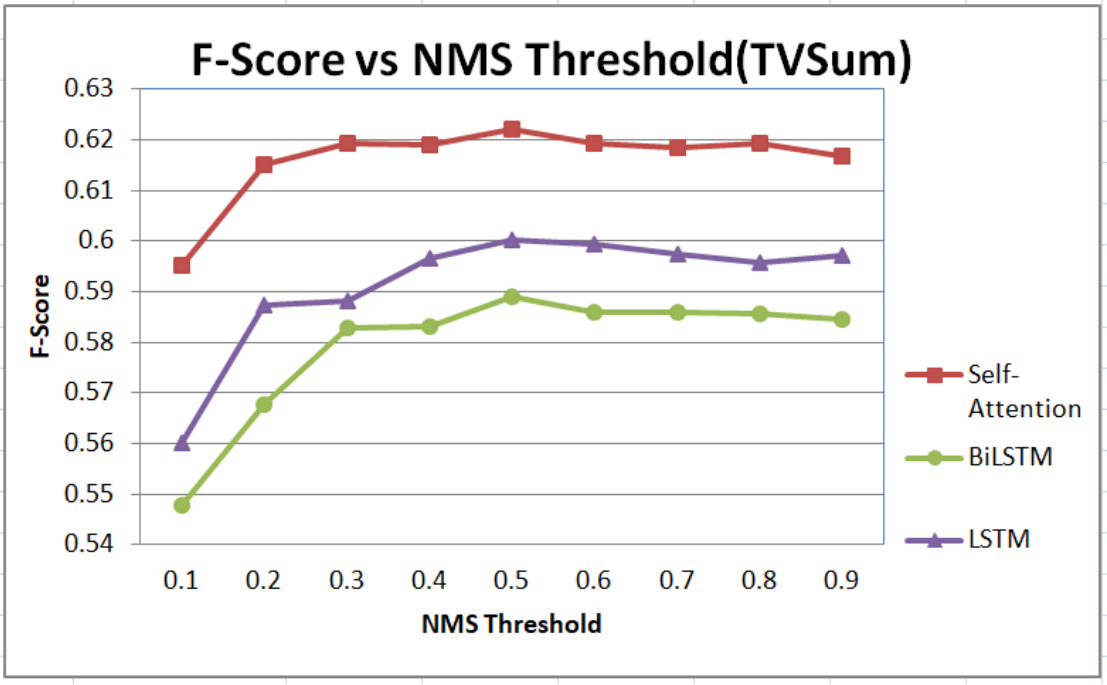


Figure 4.8: NMS Threshold analysis on TVSum dataset (Default is set at 0.5)

4.3 Significance of Temporal Sequence and Continuity

In order to get video summary temporal sequence and the continuity is major concern. Thus to investigate the significance of the temporal continuity some experiments were performed. To ensure the continuity in summary relevant proposals are being selected which are refined by using the regression module. The result has to be compared with the reference values. The reference values are calculated by without generating proposals and importance scores are directly predicted using self attention mechanism. Table 4.6 shows the results obtained from the experiment done using self attention.$\lambda = 0$ shows parameter λ is zero in the loss function in equation 3.1 and proposals are only generated but not refined using regression. It can be observed that the result has been degraded in such case as compared when refining of the proposals are performed. With Refining of proposals the permanence is superior in both the datasets.

Methods	Relevant Proposals	Refined Proposals	SumMe	TVSumm
Reference	×	×	48.8	59.6
$\lambda = 0$	✓	×	47.4	60.7
$\lambda = 1$	✓	✓	57.29	64.36

Table 4.6: F-Score comparison in terms of temporal continuity and refined proposals

Runtime	SumMe	TVSumm
Average Frames(number)	293	470
Average time(ms)	17.25	31.18

Table 4.7: Runtime Analysis(average)

4.4 Runtime Analysis

The inference time has been calculated of the model to identify hwo much time is required. To calculate inference time runtime is calculated after the extraction of features in GoogleNet. Table 4.7 shows the inference time averaged on SumMe and TVSum datasets. It is computed per video basis on the average of 15 frames per second. The runtime has been expressed in the ms and frames is in number.

4.5 Qualitative Results

For the intuitive interpretation of the result some qualitative result analysis has been performed which shows the effectiveness of the framework and visualization of the results. Figure 4.9 shows the result comprising the selected frames from the predicted keyshots of the video_1. It can be observed that predicted keyshots are close enough with the ground truth segments and frames in figure 4.9 are sample of the selected frames most of these frames represent most of the parts of the video and gives quite information about the content of the video.
Moreover figure 4.10 gives the comparison of the result from this proposed method with the ground truth as well as with other methods of previous works along with the human summaries. From the figure we can infer that this method gives precise segments compared with the other methods and human summaries. Score of the individual frames are also shown according to which the keyshots are selected. Similarly it can also be observed that the keyshots from other previous works VASNet[8] and dppLSTM[29] predicts the shots which are dissimilar to that of ground truths and irrespective of the predicted score. It leads to selection of incomplete as well as unimportant segments from the original video.

Figure 5.11 in the appendix A consists of the sample of the selected keyframes of the plane landing video of TVSum Dataset. It can be noticed that the selected keyframes are sufficient enough to get the idea of about the original video that the content in video is supposed to be the plane landing on beach side. Most of the selected keyframes are the representative content of the original video without going through the complete video.

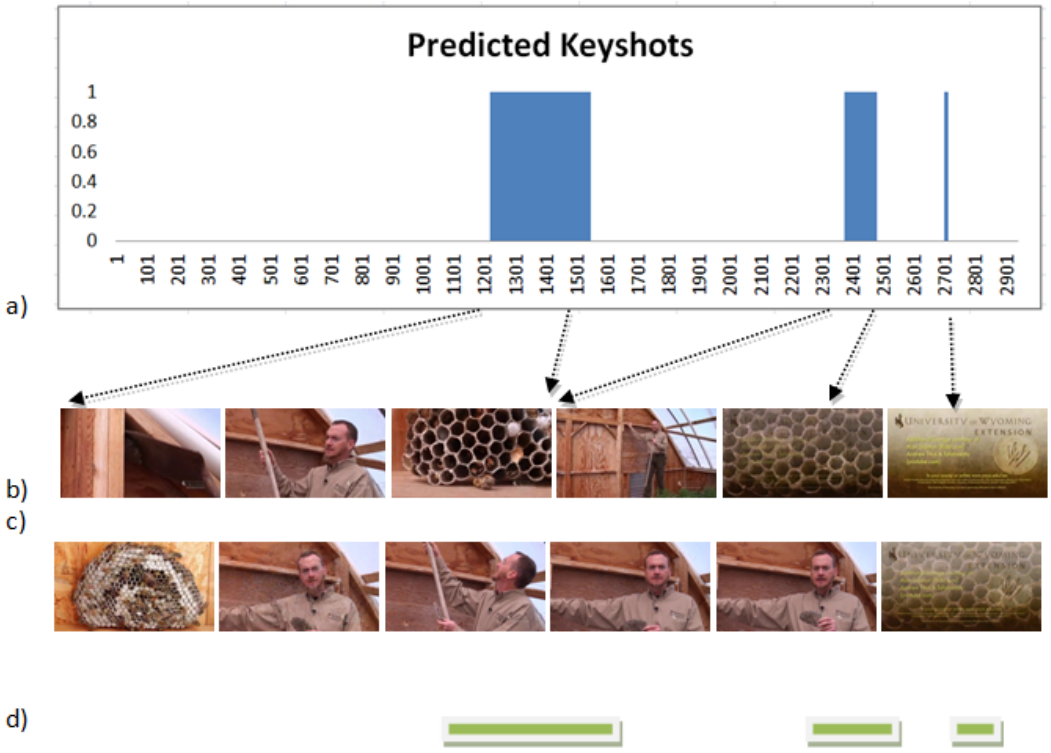


Figure 4.9: Sample of selected frames compared with predicted segments and ground truth. a) Predicted segments of the video_1 (from TVSumm dataset), b) sample of selected frames from predicted segments, c) sample of selected frames using uniform sampling(every 7th Frame) and d) Ground Truth Segments, X-axis denotes the frame indices.

Figure 4.10: Comparison of frames and keyshots selected using different methods and ground truth of playing ball video from SumMe dataset. Horizontal axes of above graphs show frames indices while vertical axes show importance score of the respective frames

CHAPTER 5

CONCLUSION AND DISCUSSIONS

5.1 Conclusion

In this thesis work a video summarization framework has been formulated which can identify the representative content from the video with the help of spatial and temporal features from each frames by generating anchor based temporal proposals on each frame time stamps. In contrasts with previous works end to end supervised training is performed to classify importance score and predict the segments simultaneously which handles the variable length of ground truth segments and prevents selection of the irrelevant and incomplete segments. The framework has novel approach of using the temporal proposal generation technique concatenated with the classification and regression making a end to end training model. Model is trained on two dataset viz: TVSumm and SumMe and these two datasets are augmented with YouTube and OVP dataset and loss is calculated with respect to ground truth to make the model trainable. Evaluation of the trained model for the video summaries is using the F-Score for quantitative measurements of the result and the performance. It can be noted that proposed work has successfully outperformed the earlier works in the same domain.

5.2 Limitations and Future Enhancements

Although the video has been summarised using visual features, audio has not been included because audio in the summarised video will not give any semantic meaning to it and there will be the mismatch between summary and audio file. The dataset which are used in this work are intentionally used without the audio because audio can make the users to make summaries biased to the audio perspective which contradicts the objective of this work which is solely based on representative content and visual stimuli of the video [24]. Since the training and evaluation of the model are performed without audio the resulted summaries are expected to be only summarised video. Moreover the lack of the labelled dataset in the different domains constrain to make deep learning model more effective.

There are lots of research going in the domain of video summarization to get summaries which can represent original video more appropriately. This work can also be further researched to train the model with varieties of labeled datasets in wide range of domain. Training the model with video datasets which are annotated by many expertise from different domains and application can make the model more precise and accurate with wide range of application.

REFERENCES

[1] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[3] Wen-Sheng Chu, Yale Song, and Alejandro Jaimes. Video co-summarization: Video summarization by visual co-occurrence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3584–3592, 2015.

[4] Cheng Huang and Hongmei Wang. A novel key-frames selection framework for comprehensive video summarization. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(2):577–589, 2019.

[5] Sandra Eliza Fontes De Avila, Ana Paula Brandao Lopes, Antonio da Luz Jr, and Arnaldo de Albuquerque Araújo. Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method. *Pattern Recognition Letters*, 32(1):56–68, 2011.

[6] Youssef Hadi, Fedwa Essannouni, and Rachid Oulad Haj Thami. Video summarization by k-medoid clustering. In *Proceedings of the 2006 ACM symposium on Applied computing*, pages 1400–1401, 2006.

[7] Bo Xiong, Yannis Kalantidis, Deepti Ghadiyaram, and Kristen Grauman. Less is more: Learning highlight detection from video duration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1258–1267, 2019.

[8] Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. Summarizing videos with attention. In *Asian Conference on Computer Vision*, pages 39–54. Springer, 2018.

[9] Michael Gygli, Helmut Grabner, and Luc Van Gool. Video summarization by learning submodular mixtures of objectives. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3090–3098, 2015.

[10] Behrooz Mahasseni, Michael Lam, and Sinisa Todorovic. Unsupervised video summarization with adversarial lstm networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 202–211, 2017.

[11] Kaiyang Zhou, Yu Qiao, and Tao Xiang. Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[12] Sijia Cai, Wangmeng Zuo, Larry S Davis, and Lei Zhang. Weakly-supervised video summarization using variational encoder-decoder and web prior. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 184–200, 2018.

[13] Tsu-Jui Fu, Shao-Heng Tai, and Hwann-Tzong Chen. Attentive and adversarial learning for video summarization. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1579–1587. IEEE, 2019.

[14] Siyu Huang, Xi Li, Zhongfei Zhang, Fei Wu, and Junwei Han. User-ranking video summarization with multi-stage spatio–temporal representation. *IEEE Transactions on Image Processing*, 28(6):2654–2664, 2018.

[15] Shiyang Lu, Zhiyong Wang, Tao Mei, Genliang Guan, and David Dagan Feng. A bag-of-importance model with locality-constrained coding based feature learning for video summarization. *IEEE Transactions on Multimedia*, 16(6):1497–1509, 2014.

[16] Qiao Luan, Mingli Song, Chu Yee Liau, Jiajun Bu, Zicheng Liu, and Ming-Ting Sun. Video summarization based on nonnegative linear reconstruction.

In *2014 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2014.

[17] Ehsan Elhamifar, Guillermo Sapiro, and Rene Vidal. See all by looking at a few: Sparse modeling for finding representative objects. In *2012 IEEE conference on computer vision and pattern recognition*, pages 1600–1607. IEEE, 2012.

[18] Rameswar Panda and Amit K Roy-Chowdhury. Multi-view surveillance video summarization via joint embedding and sparse optimization. *IEEE Transactions on Multimedia*, 19(9):2010–2021, 2017.

[19] Ehsan Elhamifar, Guillermo Sapiro, and S Shankar Sastry. Dissimilarity-based sparse subset selection. *IEEE transactions on pattern analysis and machine intelligence*, 38(11):2182–2197, 2015.

[20] Mrigank Rochan and Yang Wang. Video summarization by learning from unpaired data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7902–7911, 2019.

[21] Li Yuan, Francis EH Tay, Ping Li, Li Zhou, and Jiashi Feng. Cycle-sum: cycle-consistent adversarial lstm networks for unsupervised video summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9143–9150, 2019.

[22] Aditya Khosla, Raffay Hamid, Chih-Jen Lin, and Neel Sundaresan. Large-scale video summarization using web-image priors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2698–2705, 2013.

[23] Rameswar Panda, Abir Das, Ziyan Wu, Jan Ernst, and Amit K Roy-Chowdhury. Weakly supervised summarization of web videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3657–3666, 2017.

[24] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. Tvsum: Summarizing web videos using titles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5179–5187, 2015.

[25] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. Creating summaries from user videos. In *European conference on computer vision*, pages 505–520. Springer, 2014.

[26] Boqing Gong, Wei-Lun Chao, Kristen Grauman, and Fei Sha. Diverse sequential subset selection for supervised video summarization. *Advances in neural information processing systems*, 27:2069–2077, 2014.

[27] Aidean Sharghi, Boqing Gong, and Mubarak Shah. Query-focused extractive video summarization. In *European Conference on Computer Vision*, pages 3–19. Springer, 2016.

[28] Alex Kulesza and Ben Taskar. Determinantal point processes for machine learning. *arXiv preprint arXiv:1207.6083*, 2012.

[29] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Video summarization with long short-term memory. In *European conference on computer vision*, pages 766–782. Springer, 2016.

[30] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. Hierarchical recurrent neural network for video summarization. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 863–871, 2017.

[31] Ke Zhang, Kristen Grauman, and Fei Sha. Retrospective encoders for video summarization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 383–399, 2018.

[32] Tanveer Hussain, Khan Muhammad, Amin Ullah, Zehong Cao, Sung Wook Baik, and Victor Hugo C de Albuquerque. Cloud-assisted multiview video summarization using cnn and bidirectional lstm. *IEEE Transactions on Industrial Informatics*, 16(1):77–86, 2019.

[33] Zhong Ji, Kailin Xiong, Yanwei Pang, and Xuelong Li. Video summarization with attention-based encoder–decoder networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(6):1709–1717, 2019.

[34] Jiyang Gao, Zhenheng Yang, Kan Chen, Chen Sun, and Ram Nevatia. Turn tap: Temporal unit regression network for temporal action proposals. In

Proceedings of the IEEE international conference on computer vision, pages 3628–3636, 2017.

[35] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A Ross, Jia Deng, and Rahul Sukthankar. Rethinking the faster r-cnn architecture for temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1130–1139, 2018.

[36] Paul Kline. *The handbook of psychological testing*. Psychology Press, 2000.

APPENDIX A

Figure 5.1: Samples of the selected frames of the Landing plane video of SumMe dataset

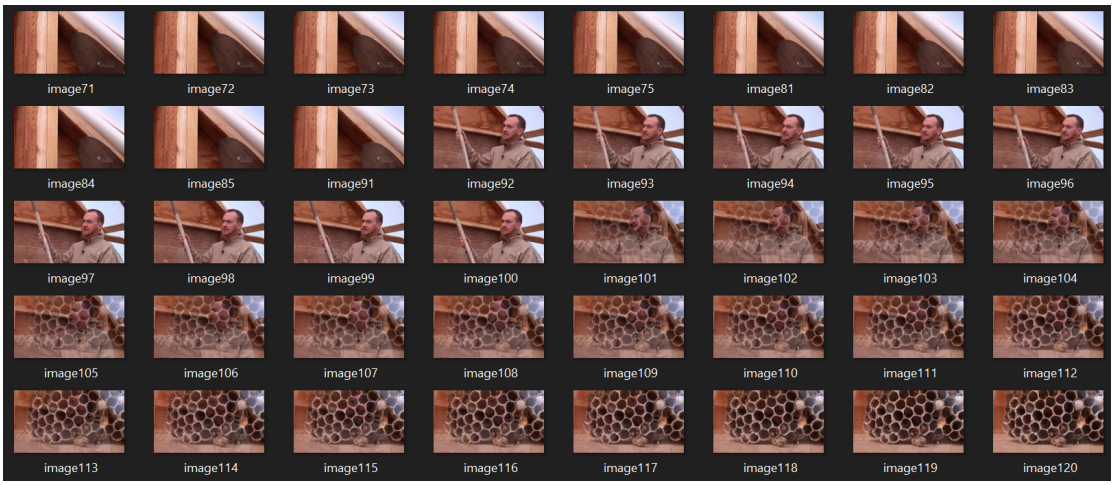


Figure 5.2: Samples of the selected frames of video from TVSum dataset

ORIGINALITY REPORT

12%	9%	8%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	varcity.eu Internet Source	3%
2	Submitted to CSU, San Jose State University Student Paper	2%
3	export.arxiv.org Internet Source	2%
4	"Computer Vision – ACCV 2018 Workshops", Springer Science and Business Media LLC, 2019 Publication	<1%
5	arxiv.org Internet Source	<1%
6	Submitted to The British College Student Paper	<1%
7	docplayer.net Internet Source	<1%
8	Maciej Pęśko, Adam Svystun, Paweł Andruszkiewicz, Przemysław Rokita, Tomasz	<1%