



Army Institute of Technology, Pune

“Video Summarization using Deep Neural Network”

Team Members:

Ravi Raj

Aman Kumar Singh

Varad Bhatnagar

Sneha Mane

Group No: 11

Guide: Prof Nilima Walde

Date: 20 December 2018

Overview


- Introduction
- Project Objective
- Platform and Technology
- System Architecture
- Algorithms and procedures
- Diagrams
- Implementation Till Now
- Papers
- Conclusion
- References

Introduction

Due to research in the domains of Machine Learning and Neural Networks, many new concepts and algorithms have come up in the 21st century which aim to solve problems in an out of the box way.

One such problem is the Video Summarization. People are moving towards utilisation of video based content day by day. Video Sharing platforms such as Dailymotion and Youtube are getting very popular. Videos available on these platforms are of varying lengths and genres. The project aims to generate an accurate summary of the main events happening in a video. This summary will be another video / collection of important frames. Using Neural Networks and soccernet dataset, the classification of frames into important / non important is being done.

Approach



Our approach enjoys recent advent of deep neural networks (DNNs). Our approach segments the original videos into short video segments, for each of which we calculate deep features in a high-dimensional, continuous semantic space using a DNN. We then sample a subset of video segments such that the sampled segments are semantically representative of the entire video content and are not redundant. For sampling such segments, we define an objective function that evaluates representativeness and redundancy of sampled segments. After sampling video segments, we simply concatenate them in the temporal order to generate a video summary



Project objectives

- 1 Tool for Summarization of a lengthy video will be prepared
- 2 Caption of the summarized video will also be prepared if time permits.
- 3 Accuracy will be obtained through comparison with other supervised and unsupervised techniques.

Applications of Video Summarization

Saves Time

In order to see the important details of the events in the video, with shortened video user can easily see all the important details/events in the video in less time.

Content Flagging

Videos having adult content and other objectionable content can be flagged easily instead of manually viewing the whole video.

Storage Friendly

As with this project we will have shortened video with all the important details/events in in, hence will save storage.

Sharing Friendly


With shortened video, sharing would be easy as the important details would be compressed in shortened video.

Target audience

- 01 | Social Media sites
- 02 | Commentators
- 03 | Newspapers
- 04 | News Channels
- 05 | People who have less time, storage space



Deliverables

- 
1. Product design
 2. Test plan
 3. SRS document
 4. UML Diagrams
 5. Source code
 6. Results
 7. Software Application

Platform



We aim to create a software application with a backend comprising of the required neural networks.

The backend may be pluggable based on the accuracy and precision that is obtained based on different genres of videos (i.e. There might be different backend models for different videos based on their genre). This will be decided at a later stage in the project based on the testing results.

Since there is a lot of image processing involved, use of personal PCs is infeasible. Hence we have used EC2 instance of Amazon Web Services where machine having 122 GB RAM and 16 Cores was chosen.



Use Cases



Use Case 1 : Football Highlights

Events :

1. Substitution
2. Goal
3. Yellow Cards



SoccerNet

1. The dataset is composed of 500 complete soccer games from six main European leagues, covering three seasons from 2014 to 2017 and a total duration of 764 hours.
2. A total of 6,637 temporal annotations are automatically parsed from online match reports at a one minute resolution for three main classes of events (Goal, Yellow/Red Card, and Substitution).
3. The events are present in JSON format for each and every video.

JSON FORMAT

~/Desktop/img/Labels1.json (img) - Sublime Text (UNREGISTERED)

imageconcat.py x Labels1.json x extract.py x

```
1 [
2   "UrlLocal": "england_epl/2016-2017/2016-12-31 - 20-30 Liverpool 1 - 0 Manchester City/",
3   "UrlYoutube": "",
4   "annotations": [
5     {
6       "gameTime": "1 - 06:14",
7       "label": "y-card",
8       "team": "home"
9     },
10    {
11      "gameTime": "1 - 07:37",
12      "label": "soccer-ball",
13      "team": "home"
14    },
15    {
16      "gameTime": "2 - 18:22",
17      "label": "substitution-in",
18      "team": "home"
19    },
20    {
21      "gameTime": "2 - 29:19",
22      "label": "y-card",
23      "team": "home"
24    },
25    {
26      "gameTime": "2 - 40:30",
27      "label": "substitution-in",
28      "team": "away"
29    },
30    {
31      "gameTime": "2 - 43:05",
32      "label": "substitution-in",
33      "team": "away"
34    },
35    {
36      "gameTime": "2 - 43:46",
37      "label": "substitution-in",
38      "team": "home"
39    },
40    {
41      "gameTime": "2 - 47:26",
42      "label": "y-card",
43      "team": "away"
44    }
45  ],
46  "gameAwayTeam": "Manchester City",
```

Line 1, Column 1

Spaces: 4 JSON





Use Case 2 : Archery Highlights

Events :

1. 10 pointer
2. 9 pointer
3. 8 pointer



Download from
Dreamstime.com

This watermarked comp image is for previewing purposes only.




ID 58488173

© Detlev Voss | Dreamstime.com



ALGORITHMS USED

DNN

A horizontal bar with a gray segment on the left and an orange segment on the right.

Deep learning (also known as deep structured learning or hierarchical learning) is part of a broader family of machine learning methods based on learning data representations, as opposed to task-specific algorithms. Learning can be supervised, semi-supervised or unsupervised.

Deep learning architectures such as deep neural networks, deep belief networks and recurrent neural networks have been applied to fields including computer vision, speech recognition, natural language processing, audio recognition, social network filtering, machine translation, bioinformatics, drug design and board game programs, where they have produced results comparable to and in some cases superior to human experts.

RNN


A recurrent neural network (RNN) is a class of artificial neural network where connections between nodes form a directed graph along a sequence. This allows it to exhibit temporal dynamic behavior for a time sequence. Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs. This makes them applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition. The term "recurrent neural network" is used indiscriminately to refer to two broad classes of networks with a similar general structure, where one is finite impulse and the other is infinite impulse. Both classes of networks exhibit temporal dynamic behavior. A finite impulse recurrent network is a directed acyclic graph that can be unrolled and replaced with a strictly feedforward neural network, while an infinite impulse recurrent network is a directed cyclic graph that can not be unrolled.

CNN



In machine learning, a convolutional neural network (CNN, or ConvNet) is a class of deep, feed-forward artificial neural networks, most commonly applied to analyzing visual imagery. CNNs use a variation of multilayer perceptrons designed to require minimal preprocessing. They are also known as shift invariant or space invariant artificial neural networks (SIANN), based on their shared-weights architecture and translation invariance characteristics. Convolutional networks were inspired by biological processes in that the connectivity pattern between neurons resembles the organization of the animal visual cortex. Individual cortical neurons respond to stimuli only in a restricted region of the visual field known as the receptive field. The receptive fields of different neurons partially overlap such that they cover the entire visual field.

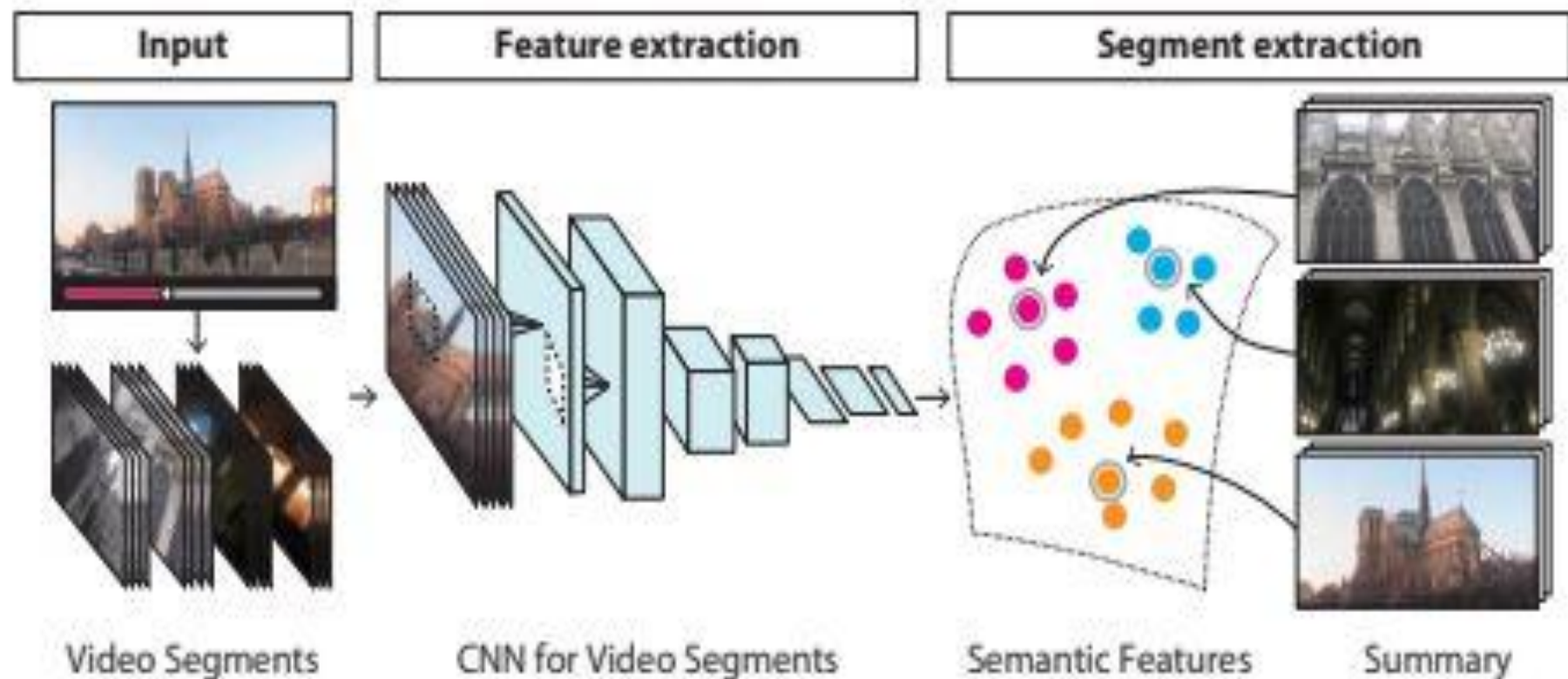
Sliding Window

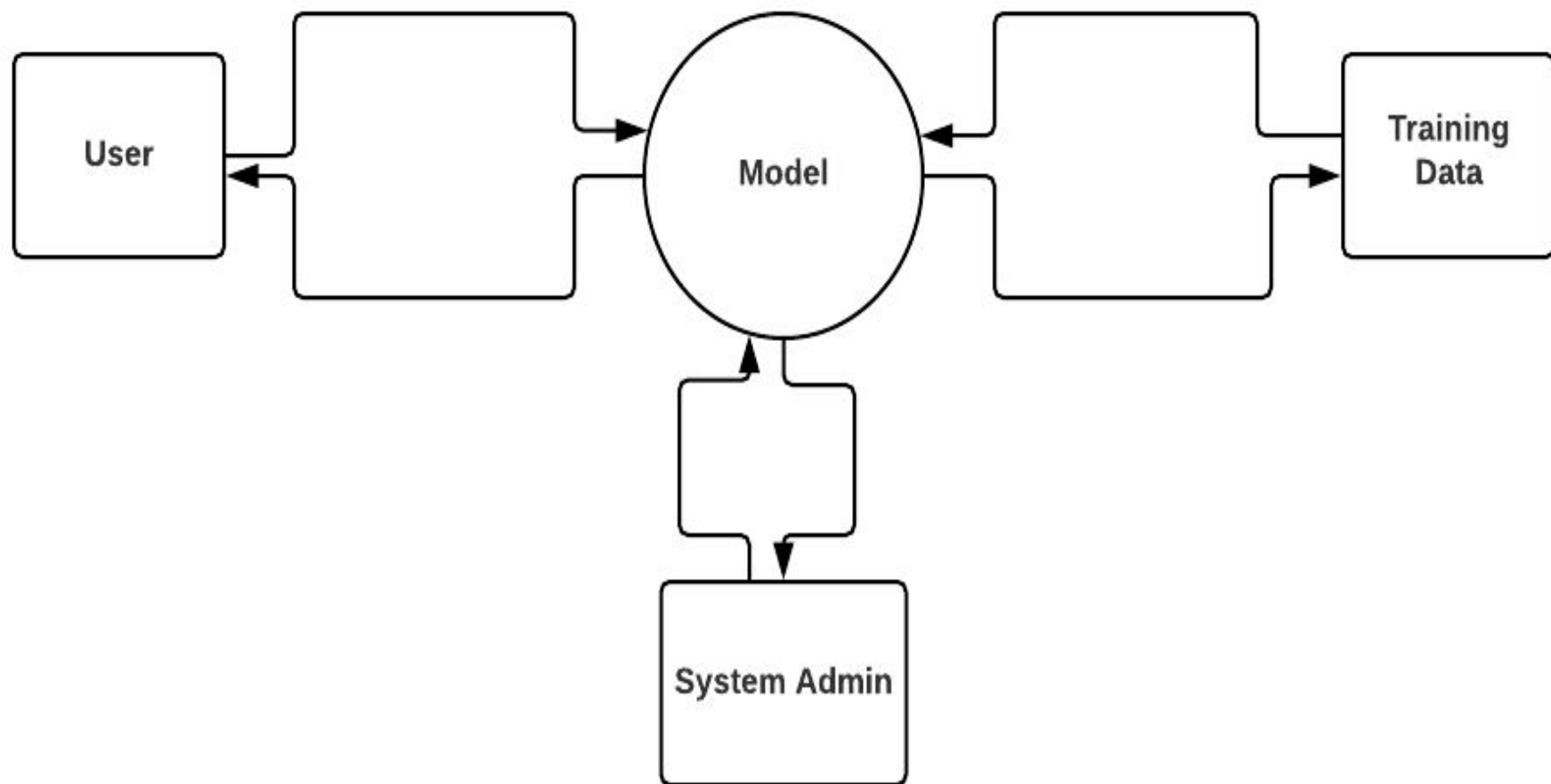


Sliding window algorithm is the concept of considering only the part of data to be taken into account for evaluation, which qualifies the criteria and discards the data that doesn't qualify to be in window of data to be evaluated.



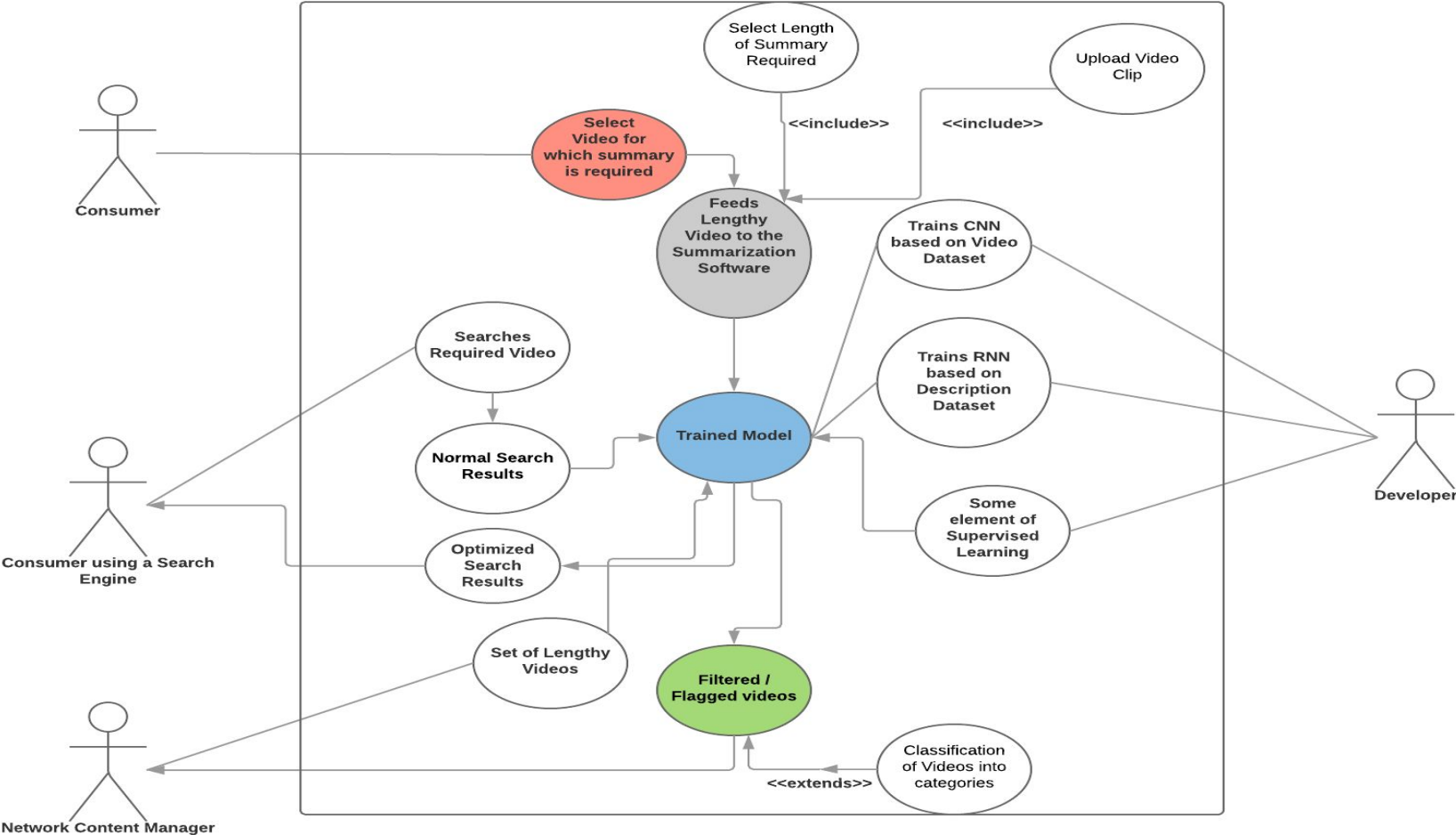
SYSTEM ARCHITECTURE



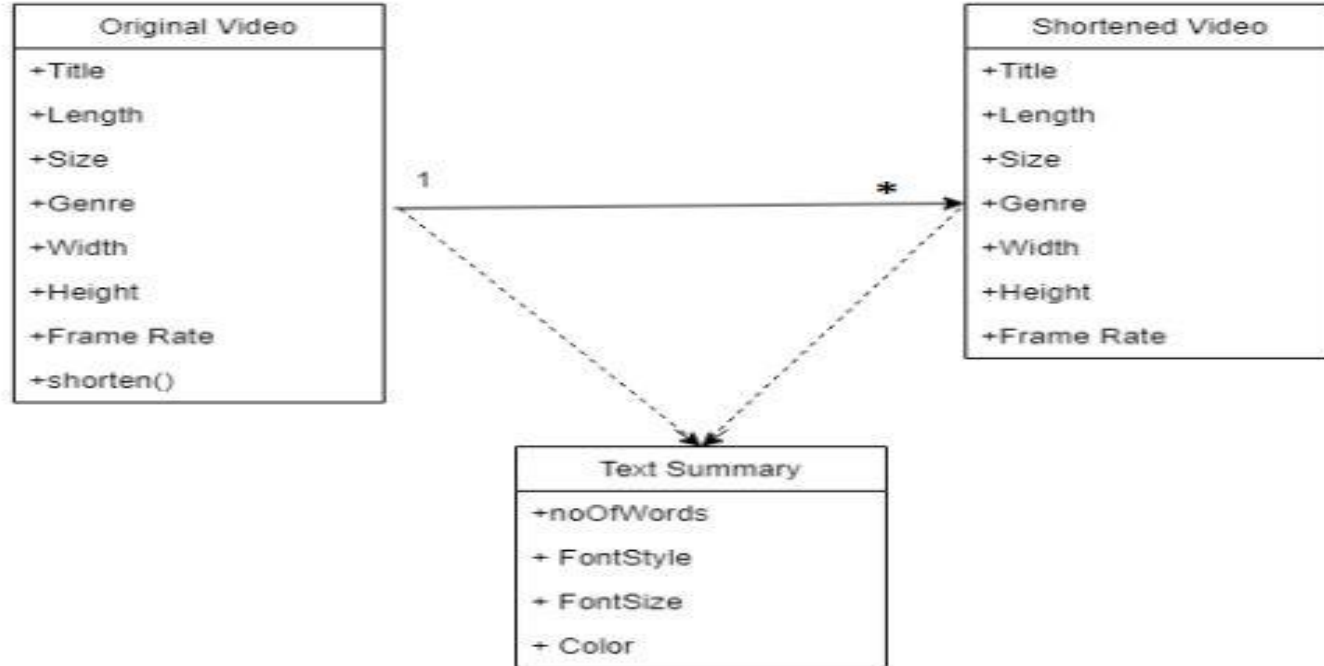




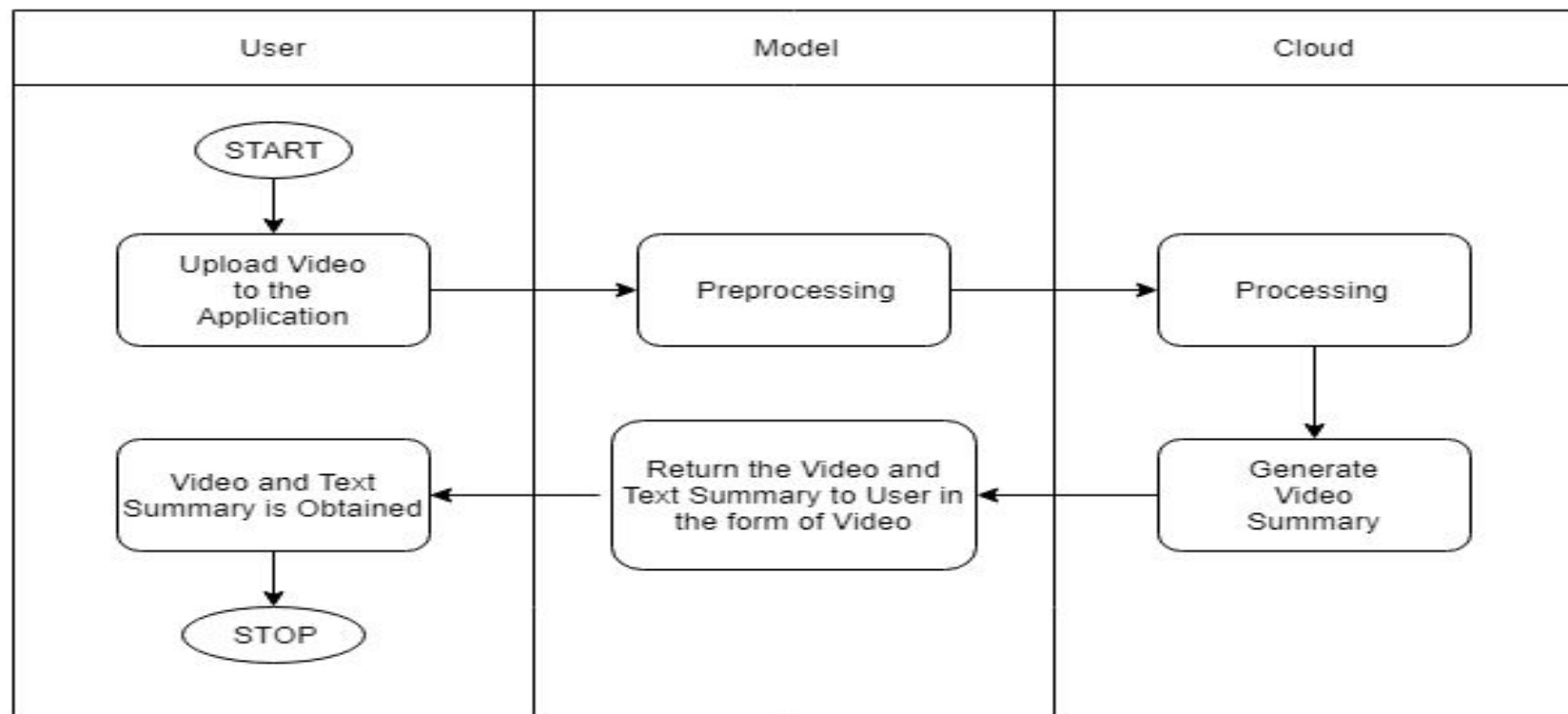
UML DIAGRAM



CLASS DIAGRAM



ACTIVITY DIAGRAM





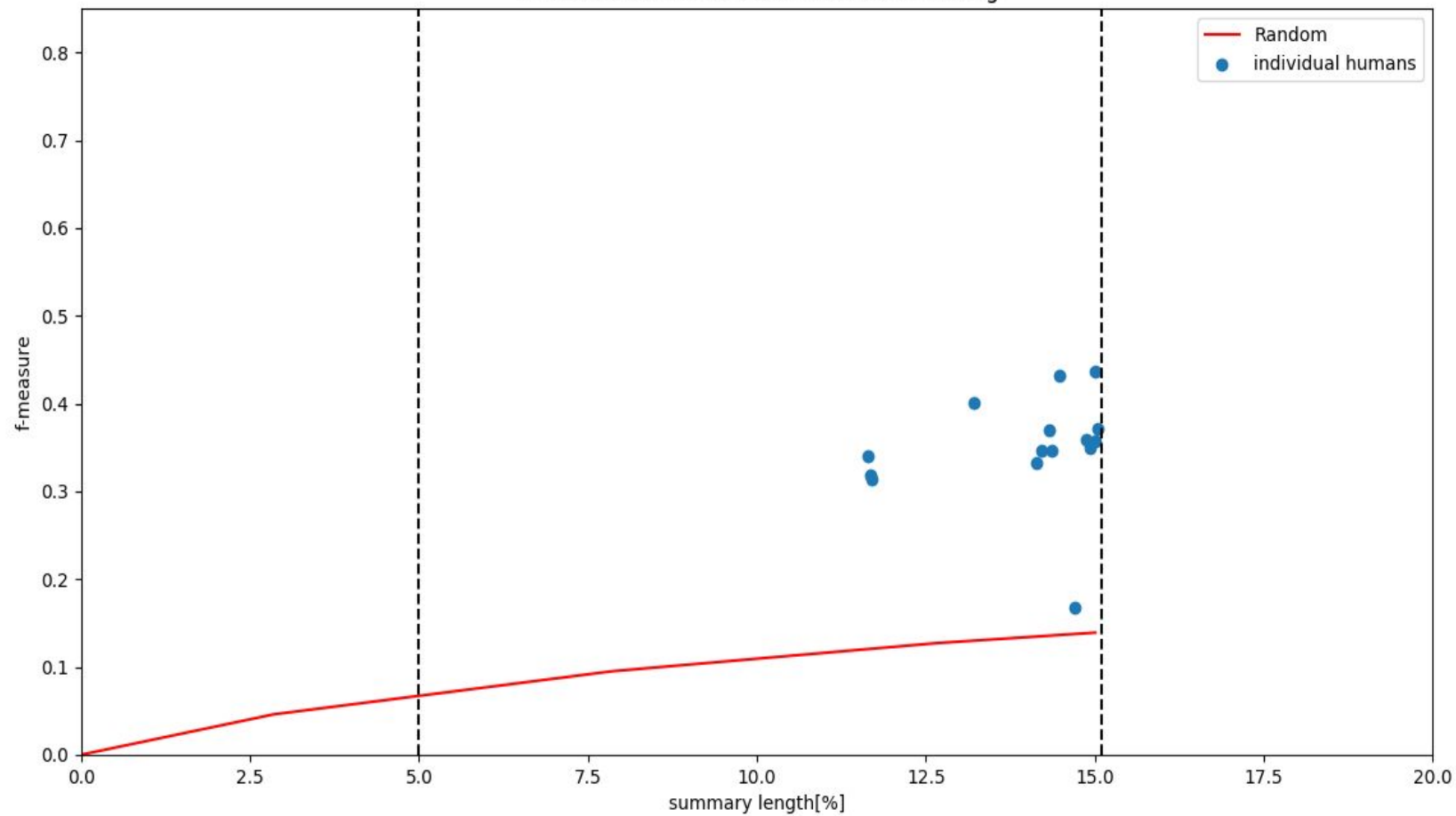
EXPERIMENTAL RESULTS

```
21/21 [=====] - 6s 299ms/step - loss: 1.3443 - acc: 0.6988 - val_loss: 1.3006 - val_acc: 0.6974
Epoch 7/25
21/21 [=====] - 6s 298ms/step - loss: 1.2447 - acc: 0.7223 - val_loss: 1.2090 - val_acc: 0.6974
Epoch 8/25
21/21 [=====] - 6s 298ms/step - loss: 1.1301 - acc: 0.7208 - val_loss: 1.1205 - val_acc: 0.7105
Epoch 9/25
21/21 [=====] - 6s 298ms/step - loss: 1.0527 - acc: 0.7223 - val_loss: 1.0729 - val_acc: 0.7105
Epoch 10/25
21/21 [=====] - 6s 298ms/step - loss: 0.9994 - acc: 0.7301 - val_loss: 1.0157 - val_acc: 0.6974
Epoch 11/25
21/21 [=====] - 6s 298ms/step - loss: 0.9916 - acc: 0.6803 - val_loss: 0.9572 - val_acc: 0.7105
Epoch 12/25
21/21 [=====] - 6s 298ms/step - loss: 0.8926 - acc: 0.7343 - val_loss: 0.8961 - val_acc: 0.7237
Epoch 13/25
21/21 [=====] - 6s 297ms/step - loss: 0.8676 - acc: 0.6907 - val_loss: 0.8700 - val_acc: 0.6711
Epoch 14/25
21/21 [=====] - 6s 297ms/step - loss: 0.7886 - acc: 0.7659 - val_loss: 0.8404 - val_acc: 0.7237
Epoch 15/25
21/21 [=====] - 6s 298ms/step - loss: 0.8753 - acc: 0.6982 - val_loss: 0.8654 - val_acc: 0.7237
Epoch 16/25
21/21 [=====] - 6s 298ms/step - loss: 0.7972 - acc: 0.7208 - val_loss: 0.7657 - val_acc: 0.7105
Epoch 17/25
21/21 [=====] - 6s 298ms/step - loss: 0.7721 - acc: 0.7146 - val_loss: 0.7662 - val_acc: 0.7237
Epoch 18/25
21/21 [=====] - 6s 297ms/step - loss: 0.7244 - acc: 0.7387 - val_loss: 0.7158 - val_acc: 0.7105
Epoch 19/25
21/21 [=====] - 6s 298ms/step - loss: 0.7153 - acc: 0.7566 - val_loss: 0.7176 - val_acc: 0.6974
Epoch 20/25
21/21 [=====] - 6s 298ms/step - loss: 0.7031 - acc: 0.7164 - val_loss: 0.7090 - val_acc: 0.6974
Epoch 21/25
21/21 [=====] - 6s 298ms/step - loss: 0.6822 - acc: 0.7238 - val_loss: 0.6856 - val_acc: 0.6974
Epoch 22/25
21/21 [=====] - 6s 303ms/step - loss: 0.6584 - acc: 0.7525 - val_loss: 0.6747 - val_acc: 0.6974
Epoch 23/25
21/21 [=====] - 6s 306ms/step - loss: 0.6474 - acc: 0.7485 - val_loss: 0.6626 - val_acc: 0.7237
Epoch 24/25
21/21 [=====] - 6s 298ms/step - loss: 0.6576 - acc: 0.7205 - val_loss: 0.6559 - val_acc: 0.6974
Epoch 25/25
21/21 [=====] - 6s 298ms/step - loss: 0.6215 - acc: 0.7614 - val_loss: 0.6552 - val_acc: 0.7237
(tensorflow_p36) ubuntu@ip-172-31-31-44:~$
```

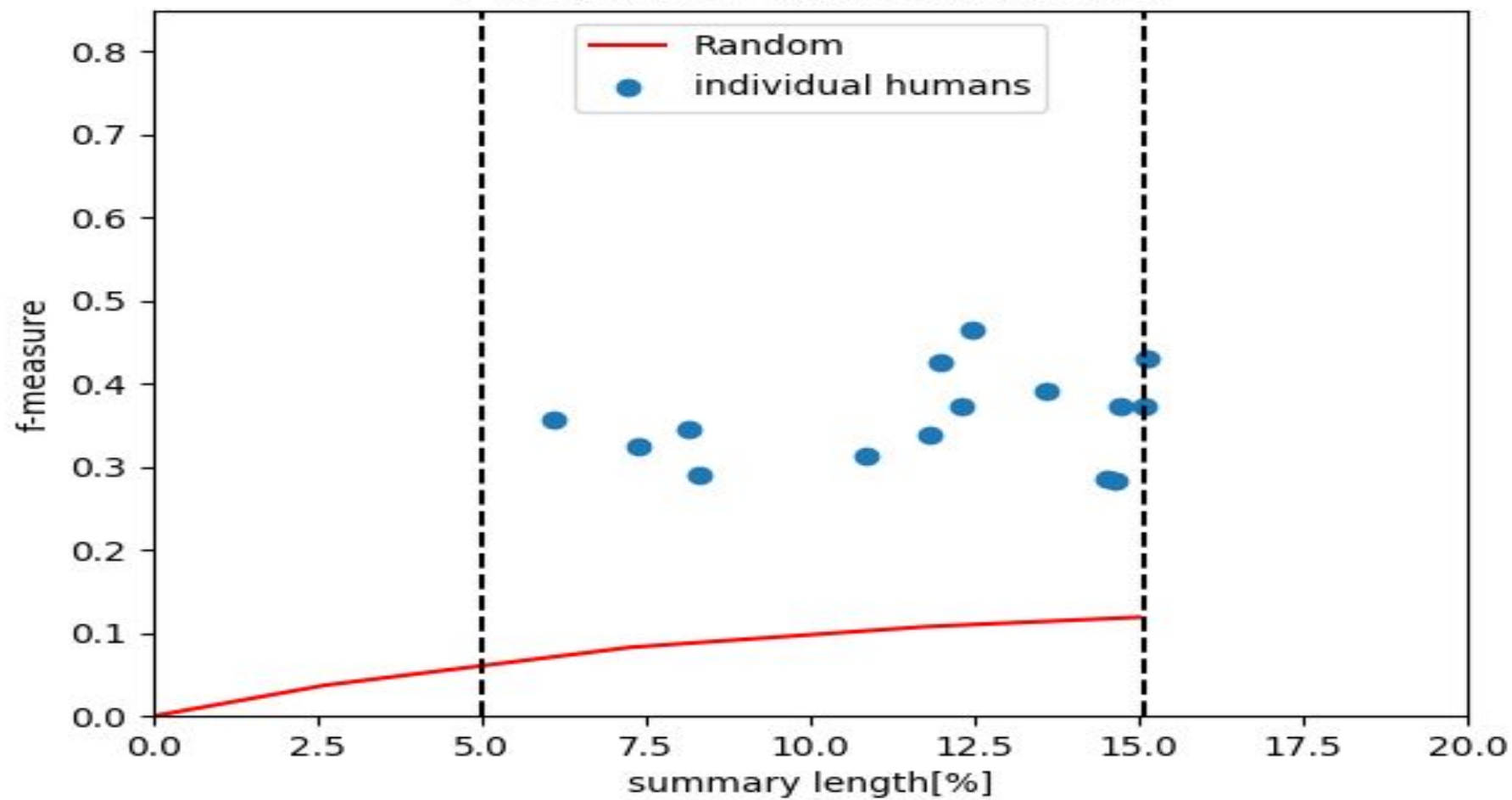


Expected Results in Future


f-measure for video Excavators river crossing



f-measure for video Eiffel Tower



Work done till date




We extracted the images from a dataset of long videos provided by soccernet, built the model which increased the accuracy of summarisation. We have been using many data cleaning techniques to increase the accuracy of the results by removing the inappropriate images. For every event data cleaning is being done, for eg: For yellow card detection in the images we used computer vision. For goal detection we will be using commentary and for substitution we will use object detection.

Deliverables completed till date

1. Python script for Data Extraction from Data.
2. Python script for Data Cleaning [OpenCV].
3. Python script for Model Training on AWS.
4. Python Script for Image Processing and Dataset Manipulation.
5. Python Script for Concatenation of frames into highlight.

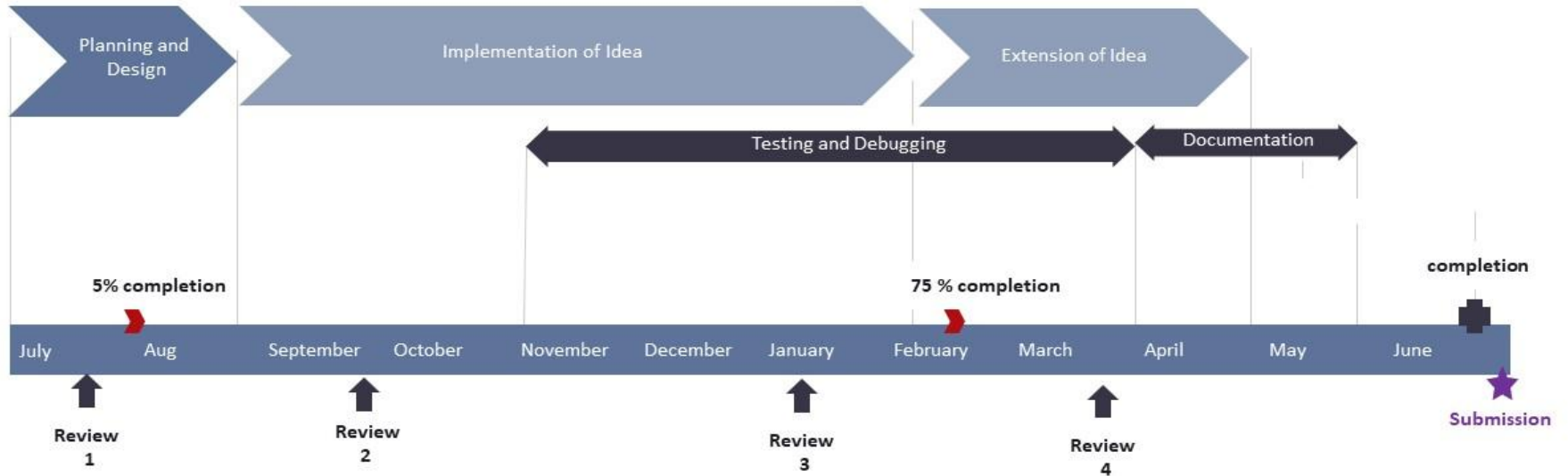
Work proposed next semester



We have an aim of detecting all the goals through commentary, substitution through object detection, missed opportunities, penalties missed and eventually give the entire summary of a long video within few minutes. We will move on to archery after football and summarise the match of archery with this technique. A typical game of archery consist of a lot of time when no work is being done so our technique will help us in saving a lot of our essential time.

Project Timeline

Video Summarization using DNN Timeline




Survey Paper



Link of paper


<https://docs.google.com/document/d/144Ee9VboHTVOL8KEL8sTlrIrFMzhmiKDYNVUp2l7dUY/edit>

Conclusion



We were given with a video. We took frames from the video, labelled it and found out enough information to summarize the entire video using the model. With the help of this model a 10 hour video is can be easily monitored within minutes and we would get all the minute information about the video using the labels. So more the labels more is the accuracy of the summarization.

References

- 
- [1]Gong, Y., Liu, X.: Video summarization using singular value decomposition. In: Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR). (2000) 174–180
 - [2]YouTube.com: Statistics–YouTube.
<https://www.youtube.com/yt/press/enGB/statistics.html> (2016)
 - [3]Gong, B., Chao, W.L., Grauman, K., Sha, F.: Diverse sequential subset selection for supervised video summarization. In: Proc. Advances in Neural Information Processing Systems (NIPS). (2014) 2069–2077



Thank you.

