



Violation Detection in Videos

Swapnil Pakhare
Ritvik Sharma
Ayush Anand
Vikrant Deshmukh



Problem Statement

To detect violations in videos and return the timestamps wherein such violations take place.

A violation for us is defined as scenes including smoking, drinking, obscenity or a combination of these.



Motivation

As of now, widespread form of filtering video data for said violations is through manual labour of hours of looking at content to be filtered.

Can this time be reduced, if not eliminated completely?

What approach can we take to build a model towards this purpose? What tools can aid us?



MACHINE LEARNING



memegenerator.net



The Solution

Our problem is to detect smoking, drinking, and obscenity in videos. At its core, this is a **multi-class video classification problem**.

All we need to do now is to build a video classifier and collect enough “well selected” data to train it on.



The Datasets

Perhaps the most important and definitely the hardest part of the project was building datasets.

Challenges :

- A good dataset should generalize well
- Should not be biased
- Minimal noise/junk
- Annotation..




The Model

Can extract images as frames in videos and train it on a CNN (and hope for the best?).

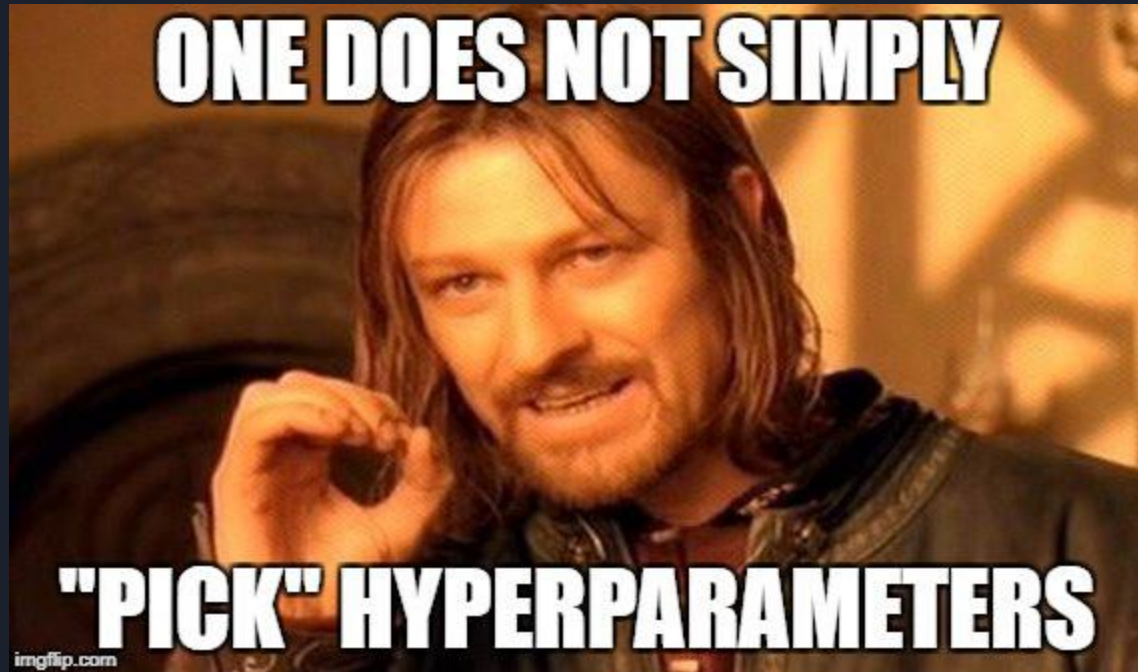
Consider a 60 sec video at 24 fps. Total frames = $60 * 24 = 1440$.

1 video = 1440 frames. Sampling and selection still leaves us with a huge number of frames to annotate and leaves us with a dataset which is more or less similar and is from just a single video.

We can annotate entire videos of the dataset and use each frame with the label of the video, but this will leave us with the problem of junk data leading us to be more careful in choosing our model architecture and hyperparameters.



ONE DOES NOT SIMPLY



"PICK" HYPERPARAMETERS

imgflip.com

Architecture

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 240, 240, 32)	4640
activation_1 (Activation)	(None, 240, 240, 32)	0
max_pooling2d_1 (MaxPooling2D)	(None, 120, 120, 32)	0
conv2d_2 (Conv2D)	(None, 120, 120, 32)	9248
activation_2 (Activation)	(None, 120, 120, 32)	0
max_pooling2d_2 (MaxPooling2D)	(None, 60, 60, 32)	0
dropout_1 (Dropout)	(None, 60, 60, 32)	0
conv2d_3 (Conv2D)	(None, 60, 60, 64)	18496
activation_3 (Activation)	(None, 60, 60, 64)	0
max_pooling2d_3 (MaxPooling2D)	(None, 30, 30, 64)	0
conv2d_4 (Conv2D)	(None, 30, 30, 64)	36928
activation_4 (Activation)	(None, 30, 30, 64)	0
max_pooling2d_4 (MaxPooling2D)	(None, 15, 15, 64)	0
dropout_2 (Dropout)	(None, 15, 15, 64)	0
flatten_1 (Flatten)	(None, 14400)	0
dense_1 (Dense)	(None, 512)	7373312
activation_5 (Activation)	(None, 512)	0
dropout_3 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 2)	1026
activation_6 (Activation)	(None, 2)	0
Total params: 7,443,650		
Trainable params: 7,443,650		
Non-trainable params: 0		

CNN Results

		prediction outcome		total
		p	n	
actual value	p'	TP 0.46	FN 0.16	p'
	n'	FP 0.28	TN 0.10	N'
total		P	N	

Accuracy: 0.56

Precision: 0.62

Recall: 0.74

F1-score: 0.67

Why doesn't this work well? Time dependency of data not being captured! Solution? RNNs!



WE'RE GONNA NEED

A GPU



LSTMs

Models that work beautifully for sequential data.

Takes into account the past context of frames while training.

Can extract features from frames and feed to LSTMs as input?



VGG16

Very good at extracting key features from images.

Features extracted from VGG16 fed into LSTM for final classification.

Architecture

Layer (type)	Output Shape	Param #
=====	=====	=====
lstm_1 (LSTM)	(None, 512)	3098624
dense_1 (Dense)	(None, 512)	262656
Activation='relu'		
dropout_1 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 2)	1026
activation_1 (Activation='softmax')	(None, 2)	0
Loss='categorical_crossentropy'		
Optimizer='rmsprop'		
=====	=====	=====
Total params: 3,362,306		
Trainable params: 3,362,306		
Non-trainable params: 0		

VGG16-LSTM Results

	p	n	total
p'	TP 0.44	FN 0.10	p'
n'	FP 0.30	TN 0.16	N'
total	P	N	

Accuracy: 0.6

Precision: 0.59

Recall: 0.81

F1-score: 0.68

Still not good enough? Have some spare time? Explore!



Bidirectional LSTM (BLSTM)

LSTMs take into consideration only past context of a sequence.

Bidirectional LSTMs take both past and future context, thus have more context.

The input sequence is fed as is as well as in a reversed format.



Architecture

Layer (type)	Output Shape	Param #
=====		
bidirectional_1 (Bidirection	(None, 4, 1024)	6197248
bidirectional_2 (Bidirection	(None, 20)	82800
dense_1 (Dense)	(None, 512)	10752
Activation = 'relu'		
dropout_1 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 2)	1026
activation_1 (Activation='softmax')	(None, 2)	0
Loss='categorical_crossentropy'		
Optimizer='rmsprop'		
=====		
Total params: 6,291,826		
Trainable params: 6,291,826		
Non-trainable params: 0		

BLSTM Results

	p	n	total
p'	TP 0.48	FN 0.04	p'
n'	FP 0.26	TN 0.22	N'
total	P	N	

Accuracy: 0.7

Precision: 0.64

Recall: 0.92

F1-score: 0.76



Scope for Improvement

- Current model fails for cases where scene depicts a smoker using means of smoking apart from cigarettes.
- Dataset can be improved.
- Our obscenity classifier is a nudity classifier, with larger datasets we can ensure it classifies any and all forms of obscenity.



Scope

- Model can find widespread use in the film industry among others.
- Model can be extended to any sort of video classification and hence can find use in Video Surveillance to detect threats, crimes etc.

Thank You

