

Deep Fake Detection in Videos using Convolutional and Recurrent Strategies

*Project report submitted to
Indian Institute of Information Technology, Nagpur, in
partial fulfillment of the requirements for the award of
the degree of*

**Bachelor of Technology
In
Department of Computer Science and Engineering**

By

**B. Sree Charan
(BT16CSE024)**

Under the guidance of

**Dr. Jitendra V. Tembhurne,
Assistant Professor**



**Department of Computer Science and Engineering
Indian Institute of Information Technology, Nagpur 440 006(India)**

2016-2020

Deep Fake Detection in Videos using Convolutional and Recurrent Strategies

*Project report submitted to
Indian Institute of Information Technology, Nagpur, in
partial fulfillment of the requirements for the award of
the degree of*

**Bachelor of Technology
In
Department of Computer Science and Engineering**

By

**B. Sree Charan
(BT16CSE024)**

Under the guidance of

**Dr. Jitendra V. Tembhurne,
Assistant Professor**



**Department of Computer Science and Engineering
Indian Institute of Information Technology, Nagpur 440 006(India)**

2016-2020

© Indian Institute of Information Technology, Nagpur (IIIT) 2020


Declaration

I, **B. Sree Charan**, hereby declare that this project work titled “**Deep Fake Detection in Videos using Convolutional and Recurrent Strategies**” is carried out by me in the Department of Computer Science and Engineering in Indian Institute of Information Technology, Nagpur. The work is original and has not been submitted earlier whole or in part for the award of any degree/diploma at this or any other Institution/University.

Date: 22/06/2020

Name : **B. Sree Charan**

Enrollment No. : **BT16CSE024**

Signature : 


Declaration

I, **B. Sree Charan**, Enrollment No (**BT16CSE024**), understand that plagiarism is defined as any one or the combination of the following:

1. Uncredited verbatim copying of individual sentences, paragraphs or illustrations (such as graphs, diagrams, etc.) from any source, published or unpublished, including the internet.
2. Uncredited improper paraphrasing of pages or paragraphs (changing a few words or phrases, or rearranging the original sentence order).
3. Credited verbatim copying of a major portion of a paper (or thesis chapter) without clear delineation of who did or wrote what. (Source: IEEE, the institute, Dec.2004) I have made sure that all the ideas, expressions, graphs, diagrams, etc. that are not a result of my own work, are properly credited. Long phrases or sentences that had to be used verbatim from published literature have been clearly identified using quotation marks.

I affirm that no portion of my work can be considered as plagiarism and I take full responsibility if such complaint occurs. I understand fully well the guide of the thesis may not be in a position to check for possibility of such incidences of plagiarism in this body of work.

Date: 22/06/2020

Name : **B. Sree Charan**
Enrollment No. : **BT16CSE024**
Signature : 
Department of CSE,
IIIT, NAGPUR.



भारतीय सूचना प्रौद्योगिकी संस्थान, नागपुर

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, NAGPUR

"An Institution of National Importance by an Act of Parliament"

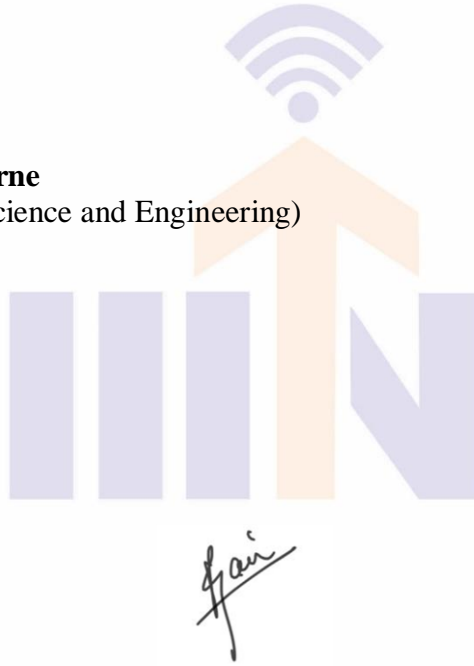
RTTC, BSNL Near TV Tower, Beside Balaji Temple, Seminary Hills, Nagpur - 440 006

Website: www.iiitn.ac.in, Email: director@iiitn.ac.in, registrar@iiitn.ac.in Phone: 0712 - 2985010

Thesis Approval Certificate

This is to certify that the project titled "**Deep Fake Detection in Videos using Convolutional and Recurrent Strategies**", is submitted by **B. Sree Charan** with enrollment number **BT16CSE024** in the partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering**. The work is found to be fit for final evaluation.

Name: Dr. Jitendra V. Tembhurne
(Assistant Professor, Computer Science and Engineering)



Name: Dr. Pooja Jain
(Head, Computer Science and Engineering)

Date: 22 / 06 / 2020

Place: Nagpur.

Abstract

As of late, rapid mounting of artificial intelligence and deep learning, machines have become very powerful that one can practically synthesize voices and faces, which leaves barely any hints of being fake, called 'Deepfakes'. There are many forms of Deepfakes, such as -- Deepfakes, Face2Face, FaceSwaps etc. Close to recent, with the increase in availability of computational power, masses can synthesize Deepfakes, which results in fraud and other malpractices like making fakes to blackmail public figures, to manipulate opinions and creating controversies. Due to the increase in spread of digital misinformation there is an urgent need of a model to distinguish the difference between real and fake digital data. This project is an attempt to detect such misinformations to a level of sheer accuracy. We proposed a system which makes use of a Convolution Neural Network to identify the unique features and characteristics of a Deepfakes. These characteristics are then analyzed by a Recurrent Neural Network to identify if an external medium has orchestrated the video. This system trains on the subject of a large collection of videos and we have used a dataset named Celeb-DF-v2, for training the model which is a vast collection of 6528 original and synthesized videos made by Deepfake Forensics and achieved an accuracy precisely to 97%.

Keywords : Deepfakes, misinformation, auto-encoders, Generative adversarial networks, Convolution neural networks, Long Short Term Memory, temporal dependencies.

Table of Contents

Sr. No.		Page No.
	<i>Abstract</i>	<i>i</i>
	<i>Table of Contents</i>	<i>ii</i>
	<i>List of Figures</i>	<i>iv</i>
	<i>List of Tables</i>	<i>vi</i>
	<i>List of Abbreviations</i>	<i>vii</i>
1	Introduction	1
2	Literature Review	4
3	Methodology	12
	3.1. Proposed system	12
	3.1.1. Network architecture	14
	3.1.2. Dataset	18
	3.1.3. Data preprocessing	18
	3.1.4. Parameter settings	20
	3.2. Requirements	21
	3.2.1. Hardware requirements	21
	3.2.2. Software requirements	21
	3.2.2.1. Keras	21

	3.2.2.2. TensorFlow	22
	3.2.2.3. Anaconda	23
	3.2.2.4. Kaggle Kernel (Online)	25
4	Results and Discussion	26
	4.1. Results	26
	4.2. Summary	31
5	Conclusion and Further Enhancement	33
6	References	34
7	Acknowledgement	37

List of Figures

Figure No.	Description	Page No.
1.1(a)	Depiction of real images	2
1.1(b)	Depiction of synthesized images	2
1.2	Method of synthesizing Deepfakes	2
3.1	Proposed system for Deepfake detection	13
3.2	Standard LSTM network	16
3.3	Bi – directional LSTM	17
3.4	Proposed model using LSTM	17
3.5	Proposed model using Bi - LSTM	18
3.6	Depiction of data preprocessing	19
3.7	Functionalities of Keras	22
3.8	Dataset/model is trained and deployed using TensorFlow	23
3.9	Collection of features by Anaconda	24
3.10	Kaggle Kernels	25

4.1	Epoch categorical accuracy of the VGG-16 + LSTM model	27
4.2	Epoch loss of the VGG-16 + LSTM model	27
4.3	Epoch categorical accuracy of the VGG-19 + LSTM model	28
4.4	Epoch loss of the VGG-19 + LSTM model	28
4.5	Epoch categorical accuracy of the VGG-16 + BiLSTM model	29
4.6	Epoch loss of the VGG-16 + BiLSTM model	29
4.7	Epoch categorical accuracy of the VGG-19 + BiLSTM model	30
4.8	Epoch loss of the VGG-19 + BiLSTM model	30

List of Tables

Table No.	Description	Page No.
3.1	Architecture of VGG16 and VGG19	16
4.1	Proclaiming the accuracies on our dataset splits for LSTM and BiLSTM models	26
4.2	Comparison of performance of our best working model with some of existing models	31

List of Abbreviations

Short Form	Abbreviation
GANs	Generative Adversarial Networks
CNN/CNNs	Convolutional Neural Network / Networks
RNN/RNNs	Recurrent Neural Network / Networks
LSTM	Long Short Term Memory
Uni – LSTM / UniLSTM	Uni Directional Long Short Term Memory
Bi – LSTM / BiLSTM	Bi Directional Long Short Term Memory
Neural-ODE	Neural Ordinary Differential Equation
SITS	Satellite Image Time Series
TCNNs/TempCNNs	Temporal Convolutional Neural Networks
AFRS	Automatic Face Recognition System
STN	Spatial Transformer Network
VGG	Visual Geometry Group
SIANN	Shift Invariant or Space Invariant Artificial Neural Network

SGD	Stochastic Gradient Descent
GUI	Graphical User Interface
API	Application Program Interface
GPU	Graphics Programming Unit
TPU	Tensor Programming Unit

Introduction

Deepfake is a technique with which we can make a fake image of an individual by superimposing one image over the other. Deep learning models like auto-encoders and GANs can be trained to synthesize fake images. These networks are trained over a large set of images which specifically involve all kinds of face expressions, thereby creating a model which can properly decode the face of the target onto the face of the person in the photo/video. This is very powerful because we can't identify the difference between a fake and a real image using traditional methods because of the complexity of the image generation. These techniques are commonly used to target public figures, like politicians, movie stars and other celebrities. The first Deepfakes were made in 2017, where most of the targets were popular celebrities. Their faces were superimposed over the bodies of porn stars. The problem with this technology is that, it can cause many threatening situation to world peace. These models can be used to make fakes of politicians and make controversial fake statements which might lead to chaos in the world or probably start a world war. Fig. 1.1(a) shows the original images of famous personalities and Fig. 1.1(b) shows the Deepfake images generated by using synthesis and overlapping.



Fig. 1.1(a): Depiction of real images



Fig. 1.1(b) Depiction of synthesized images

In Fig. 1.2. we have two images of different celebrities. We make two encoder-decoder networks, one for each image and train the networks with the corresponding images. Once both the networks are trained, the encoder trained on image A is extracted and connected to the decoder trained on image B. When we input the image A to this mixed network, the resulting output is an overlapping of both image A and B. This is the standard way of synthesizing Deepfakes.

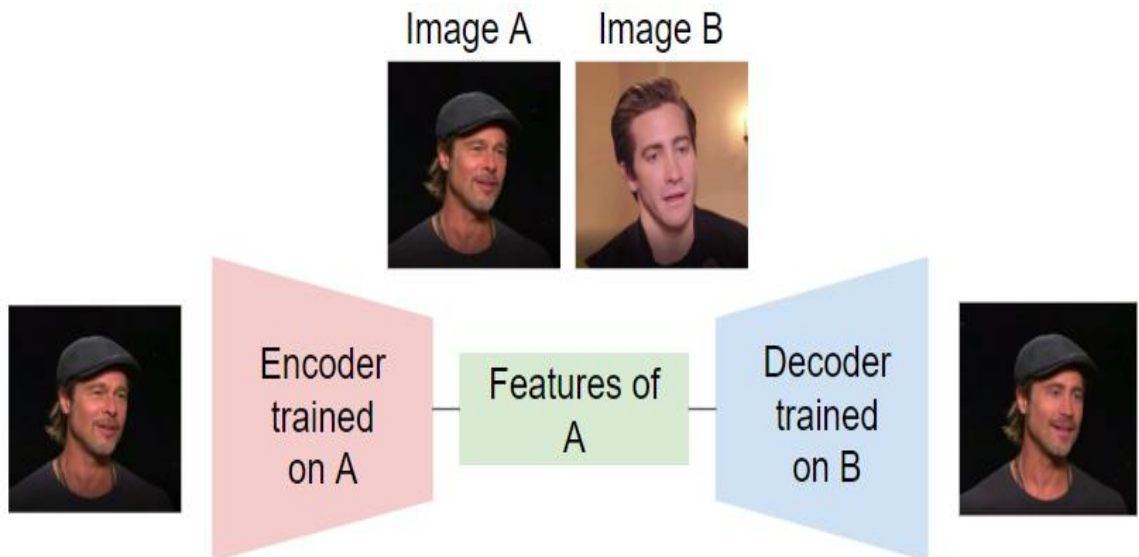


Fig. 1.2: Method of synthesizing Deepfakes

Hence, it becomes paramount to make a detection system which can correctly identify the fake images or videos and save us from any conspiracies created by them. This has motivated us to come up with a system which is an effort to improve the precision of the existing identifiers. Initial models came up with identification of handcrafted features from artifacts and finding inconsistencies in the fake images. These methods include -- using edge detection and recognizing vital parts of a face to identify fakes which didn't work out because of the complexity involved in the generation of Deepfakes itself. The features which were extracted from the artifacts were much simpler than the features of the generated fakes. So, the newer methods have opted, applied deep learning methods to find fakes. This path has given fruitful results and even our method of addressing the problem is using Deep Learning. We identify much more dense characteristics and features using deep learning and can also identify temporal dependencies in video.

Literature Review

This work is carried out to differentiate real and fake products, in the category where neglecting is not accepted and affordable. There are two types of products namely -- consumable, non-consumable. Even if the sales depend on the quality of the items, sometimes, products with less quality are manufactured and marketed to create good business. Here, they dealt by examining real and fake beedi packages (image, hologram, logo, brand) to establish genuineness. Normally both the packages i.e., real and fake are examined under Video Spectral Comparator. Essentially, this method is not enough to draw conclusions. So, thereby the proposed system is a continuation to the existing process where the logo and hologram are analyzed using histogram and edge detection techniques [4] (which is an image processing technique used for image segmentation and data extraction for finding boundaries of objects within image) by the slightest of color using MATLAB software (version 7.10) to see if effective results are drawn or not.

Progress in AI let to another misinformation/Deepfakes called Face Swap – digitally swapping the face of one object with another that barely leaves any hints of being fake resulting in political pain, extorting somebody and many more. The idea of fake forensics arised from a work that they have gone through, that includes – Digital Media Forensics with RNN. In this paper, their framework utilizes Convolutional LSTM [5] which comprises of two basic parts – 1) CNN for outline highlight extraction 2) LSTM for transient grouping investigation. Here, initially a dataset is trained and for better results Deepfakes are generated. From the complete dataset now a video is considered as input or source for CNN which subjects to removal of outline level highlights and lot of features for each casing are produced. The outputs from CNN then act as inputs for LSTM for further examination. Finally, a gauge of probability is produced which distinguishes if a video is subjected to control or not. This model is applied on large collection of video both fake and real. Inspite of it being a basic design, the main motive

is to show how well the recorded results are when compared to others. Finally, here, they have presented a time-based aware system to automatically detect fake videos. The simple LSTM that they have used can accurately predict if a video is subjected to manipulation or not (within 2 seconds of video data). The paper also aims to show how competitive outcomes are accomplished using a basic pipeline architecture. The future works can include how to build robustness of the proposed framework against controlled recordings using obscure systems during preparation.

A series of recent incidents led to the inspection of online fib. Fib can be demonstrated in many ways – direct manipulation of data or presentation of unchanged content in a misleading context. Change in digital image, which might be copy-move and slicing that constitute to be examples of ‘deliberate manipulation’, on the other hand image repurposing is an example of ‘misleading context’. These days, spread of fib in the form of images and videos that are generated synthetically but show up to be realistic is a significant problem. This problem made us develop and use robust manipulation and detection methods. Though effort is being made in detecting manipulation in still images, no attention has been paid about the tampered faces in videos. Recurrent Convolution models [6] belong to a class of deep learning which has proven to be effective at using the temporal data from image streams across domains. In this paper, they are making an attempt to choose best strategy to combine variations in the models along with domain specific face preprocessing techniques. In specific, they try to detect (Deep-Fake, Face-to-Face and Face-Reciprocation) tampered faces in video streams. Evaluation is done on recently introduced FaceForensics++ dataset, aiming to improve previous state-of-the-art by 4.55% in accuracy. The approach that is proposed is a two-step process – 1) Cropping and alignment of faces (Cropping is done with the help of masks generated by computer graphics. For alignment two techniques are used, Firstly, explicit alignment using facial landmarks and secondly, implicit alignment using STN) 2) Manipulation detection over preprocessed facial region (Using encoding network or RNN training strategies). The base papers work includes – Video Processing with Deep Models, Face Manipulation Benchmarks and Face Manipulation Detection. Finally they have proved, that a combination of recurrent convolution model and face alignment approach will provide effective results when compared to the existing methods. They have explored several branches of alignment to apply the best suitable one alongside

combining CNN feature through recurrence and finally concluded that the proposed system provides best performance in detecting the manipulated faces in videos.

The land cover (biophysical cover of earth's surface) has been declared as one of fifty-four Essential Climate Variables. So, knowledge of land cover is a key information for researchers. These studies are helpful in monitoring the effects of climate change which are obliging in managing resources and assist in disaster prevention. The studied details have to be accurate – perfect to measurement and up-to-date – till latest developments. Remote sensing sensors acquire spatial and spectral SITS. These are series of images that play an important role as a component of classification system for obtaining up-to-date and accurate results. To the point, the combination of time-based, spectral and spatial resolutions of SITS makes it possible to observe vegetation dynamics. Although some algorithms such as Random Forests can be applied on SITS but they do not use the temporal domain to its fullest. There are converse cases where the approaches will use temporal domain, to specify one, RNN. This paper proposes an intense study of another deep learning approach called Temporal Convolutional Neural Networks (TCNNs) [7] where convolutions are applied in temporal dimensions. Since TCNNs is an approach and is applied upon SITS, the main goal is to evaluate the contribution of TempCNNs for SITS classification. In the paper they have proposed a set of experiments done on one million time series taken from 46 Formosat - 2 images. The results demonstrate that Temporal Convolutional Neural Networks are more accurate than other approaches (including Random Forests (RF) and Recurrent Neural Networks (RNNs) i.e., the TempCNN architecture outperforms other approaches by 1 to 3%. Lastly, it is also the current state of the art for SITS classification. The paper also highlights the different results obtained in computer vision, moreover provides some guidelines on network architecture, hyper parameter values such as batch size and common regularization. The paper also includes a visual check which tells that, the visual analysis shows good quality of TempCNNs to accurately map without any over representation of major classes. The paper also demonstrates the importance of time-based and spectral dimensions when calculating the convolutions.

It is estimated that millions of images are uploaded to social and professional networking sites out of which 40 to 50 percentage appear to be manipulated for sometimes good-humoured or mostly harmful reasons. Image manipulation (Especially, face manipulation creates problem to AFRS) is a serious issue since it is widely used as a lead in biometric for identification and authentication services. Also, due to the advancement in deep learning algorithm generating and manipulating realistic face samples have become easier. Therefore, the public won't hold their trust over digital communication and security applications. This paper gives an overview of recent technologies that support face manipulation generation, recognition, detection and database [8]. There are several challenges that remain unaddressed which include – generalized manipulation detectors (methods go well for specific kind of manipulation but the problem arises when they have high error for alterations absent in training), adversary aware FRSs (less work is been carried out to address the problems of digitally induced changes against FRSs), wearable manipulation detection (most of the detection applications or models are unfit for mobile applications because of high computational cost, so, compact and efficient models for mobiles should be worked upon), large scale databases (availability of high quality data is always a problem) which will definitely require future research.

The increase in refinement of mobile camera and effortless reach to social media and media sharing portals have made generating and spreading of videos more convenient than never before. Till recently the fake videos and their degree of pragmatism have been limited because of no proper editing tools. But thanks to the availability and accessibility to large collection of training data and high outturn computing power. Out of all, the improvement in machine learning and computer vision techniques eliminated the need of manual editing. A new AI based fake video generation method is called Deepfakes. It takes a video with 'target' as input and the output is another video with replaced target's faces. Backbone of Deepfakes is neural networks trained on images to automatically map expressions from source to target. With all these, a fake video can obtain high level of realism. According to their observation the present Deepfake algorithm can generate images of bounded resolutions that need to be further distorted in match with the original faces to the input video. This warping leaves some artifacts [9] due to – resolution

inconsistency between surrounding context and warped face area. Eventually, these artifacts can be used to detect digital misinformation. The proposed system uses a simple CNN to distinguish the difference between real and fake images/videos. For this a dataset is generated with warping faults which comprises of 3 steps – 1) Detecting the face and extracting the region using software package dlib. 2) Face alignment into multiple scales and then picking one scale which is then smoothened by Gaussian blur with kernel size 5x5. 3) The smoothened face undergoes warp back to same size to stimulate artifacts. The previous method uses a huge amount of real and Deepfake generated images in order to train the CNN classifier but the proposed method doesn't require fake images as negative training examples because they target the artifacts to distinctively distinguish real and fake images. As observed, their method holds two major advantages – (1) Since using Deepfake generated images as negative example is time-consuming and resource-remanding, stimulating directly using image processing operations on an image and using it as negative training examples saves plenty of time and resources. (2) Artifacts are created alongside Deepfake videos from several different sources, the robustness of the proposed method is more when compared to others. In here, they have evaluated the method on sets of available videos that shows up effectiveness in practice. With an evolution in the Deepfake videos they shall definitely try to improve the detection system. As mentioned in the paper, at first they would like to improve the robustness of the system and secondly, they would definitely like to explore dedicated network structure for more effective detection.

Due to increase in popularity of electronic products most importantly mobile phones and digital cameras, large amount of data (images and videos) have been created. It is nearly 2 billion images that are added to internet every day. Due to this, various digital image editing platforms have come into light. These platforms/tools help to effortlessly shape the digital data as required. Not only electronics but evolution in AI technology made forgery of digital images and videos easier. In recent times it has become more difficult to identify such forgeries. If these forgeries disseminate with baleful intent then it will impact social and political stability, alongside many ethical and legal challenges will arise. Due to these forgeries, public's trust on digital platform – in the stream of digital images and videos is fading day by day. In this paper, they made an attempt to propose a solution for facial fraud detection. In this system the detection efficiency is

reached by using CNN and image segmentation [10]. In detail, they proposed a method which required less training parameters and also improved the accuracy and robustness – the method includes Convolution neural network alongside Image segmentation. The proposed system can be briefed as two steps – 1) Extracting the face area for each frame in the video followed by alignment and cropping process which acts as input to classifier. 2) Dividing the blocks of pre-processed face area and training using CNN. Finally, hard voting will determine label of image. Also the paper mentions the results and the differences that are obtained by using different image segmentation techniques. Finally, the proposed system shows up improved detection capabilities.

Deep learning techniques are rapidly mounting to be sophisticated regarding creation and processing of media content. With the advancement of technology comes new problems, here, it is called Deepfakes – simply permits to create realistic videos where people's faces or sometimes even lips and eye movements are modified which constitutes a serious threat to public subjects. In this paper, a new forensic technique is introduced in order to differentiate between fake and real images within videos. Unlike state-of-the-art methods they proposed the adoption of optical flow fields to utilize possible inner frame dissimilarities (feature to be learned by CNN classifier) [11]. Initially, motion vectors have been considered as 3-channel images and then goes as input to CNN which has given promising results. The attempt to take possible peculiarities in the time-based division of sequence is quite innovative. Here, the base dataset is FaceForensics++ eventually providing promising performances. It paves way for many possible future works – firstly, checking the reliability of optical flow fields by comparing with larger collection of dataset and by applying other neural networks. Secondly, it will be interesting to study inconsistencies on the temporal axis. In the end it finally depends on how effectively the performances can be improved.

Deepfake is an AI method of manipulation. To create one Deepfake video, an auto-encoder is trained with an input of collection of photos and then condensing those photos into specific data points. A second auto-encoder will perform same actions on still faces of images that are to be replaced. Now the data points of the photos are superimposed on the data points of the video to replace heads. This way Deepfake implementation is becoming easier and reachable day by day. For example, apps like FaceApp and

FakeApp allow users to create their own Deepfake videos using their smart phone. So, now it's very essential to detect the fake videos that intend spread false information. Recent study tells that heart rate of false videos can be used to differentiate original and fake videos. In this paper, they have obtained the heart rate of original videos and trained state-of-the-art Neural-ODE [12] model after which they have created fake videos using software. The proposed system can be briefed as four steps namely – 1) Creating Deepfake dataset (using a commercial website). 2) Extracting heart rate from facial videos. 3) Neural ODE training using heart rate from original videos. 4) Using trained Neural ODE predicting heart rates of Deepfake videos. The analysis tells : (1) Average loss for first ten videos is 0.010926 (2) Average loss for ten donor videos are 0.010040. The trained ODE was able to predict the heart rate of Deepfake videos. Also adding, this is the very first attempt where Neural-ODE is trained on original data which is a revolution in deep neural networks and is quite well providing results. In future, is an idea to optimize the network such that it can be implemented on low cost single board computer.

Recent forge ahead in computer graphics, machine learning and computer vision have made it easier to incorporate compelling fake audio, image and video. In audio domain, highly level-headed audio can be synthesized in which, a neural network along with enough sample recordings can synthesize speech in your voice. In image domain, realistic images can now be synthesized using GAN's. In video domain, realistic videos can be synthesized of anybody saying and doing anything that creator wants. These synthesized audios and videos are called Deepfakes. The Deepfakes are undoubtedly very entertaining but can also be easily weaponized. Unchallenging access to technology that can create misinformation for images and videos still remains the point of concern because it mostly results in small or large-scale frauds, power to derange democratic elections and create non-consenting pornography. In this paper, they are using a biometric-based forensic technique [14] for detecting face-swap Deepfakes. The proposed technique puts together a static biometric based on facial recognition that includes a time-related, behavioral biometric based on movement of head and facial expression. Here, Behavior-Net was designed to capture 'Spatiotemporal behavior', while VGG captures facial identity and after ensuring that both the results are not tangled further process is done to detect fakes. This technique can be used upon video datasets.

Due to the advancement in Deepfakes, i.e., manipulating images and videos using advanced deep learning tools like auto-encoders or generative adversarial networks is the reason that made fake multimedia a central problem in the last few years. There is a very thin boundary left between original and synthesis digital videos and images. The so called Deepfakes are mostly used in fraud, manipulating public opinion during elections and to blackmail public figures or sometimes even common people. This paper aims to show analysis of ways for visual media integrity verification [15]. The analysis will show the limitations of the existing forensics tools which will suggest future directions in the field of research. The paper is concluded saying that for present time, tools are being developed in large scale to break the norm of Deepfakes and to protect people from reaching fault information.

Methodology

3.1. Proposed system

Rapid mounting in Artificial intelligence, Deep learning techniques alongside refined mobile camera followed by easy reach to applications that support modification and last but not the least effortless access to internet, social media, sharing portals etc., made creating and spreading of misinformation (be it images or videos) expedient like never before. Fake videos/images are undoubtedly very interesting and with no reason can be converted into weapons to target public figures.

There are many advantages of advancement in technology, when there are advantages arises the disadvantages, in this case it is Deepfakes. In specific, Face-Swaps – digitally swapping one object by another with no hints of being fake. When these are spreading at a count of several thousands every hour, a way has to be found to distinguish the difference between real and fake digital objects.

The system which we proposed comprises of two parts – Firstly, a Convolutional Neural Network [13] (Also referred to as CNN or ConvNet is a deep neural networks class mostly applied to analyze visual imagery. Based on CNNs characteristics like shared-weights architecture and translation invariance they are also named as shift invariant or space invariant artificial neural networks (SIANN). Multilayer perceptrons, the terms mean a fully connected network. CNNs are regularized versions of these perceptrons. Though image processing algorithms also work the same way, the only difference between image processing algorithms and CNNs is CNNs use pre-processing where the other algorithms hardly do. CNNs are used in many fields like video and image recognition, image classification, natural language processing and many more) -- to scan and identify the features and characteristics of the image frames in a sourcevideo(provide input video). Secondly, a Recurrent Neural Network (Also referred

to as RNN is an artificial neural network class that supports connection between a nodes from directed graph alongside a timely sequence. So, RNN exhibits temporal dynamic behavior. RNNs can use their memory to process long sequence of inputs. RNNs are applicable to tasks such as handwriting and speech recognition (which are unsegmented and connected).) -- to analyze the temporal and sequential features from a selected video (out of a long length/collection of source videos/inputs).

We extract ‘n’ frames which are distributed across the entire video in sequential order. Then these frames are given as input to a CNN, which generates the feature sets of each frame respectively. These features are then sent to the Recurrent Neural Network in sequential fashion and predict the result using a Dense network (It is a grid in which the links of each node is somewhere equal to maximum number of nodes. In brief, every node is connected or linked to ‘almost every other node’ in the network. In case, if all the nodes are exactly connected to ‘all the other nodes’ then it is called as ‘completely connected network’.).

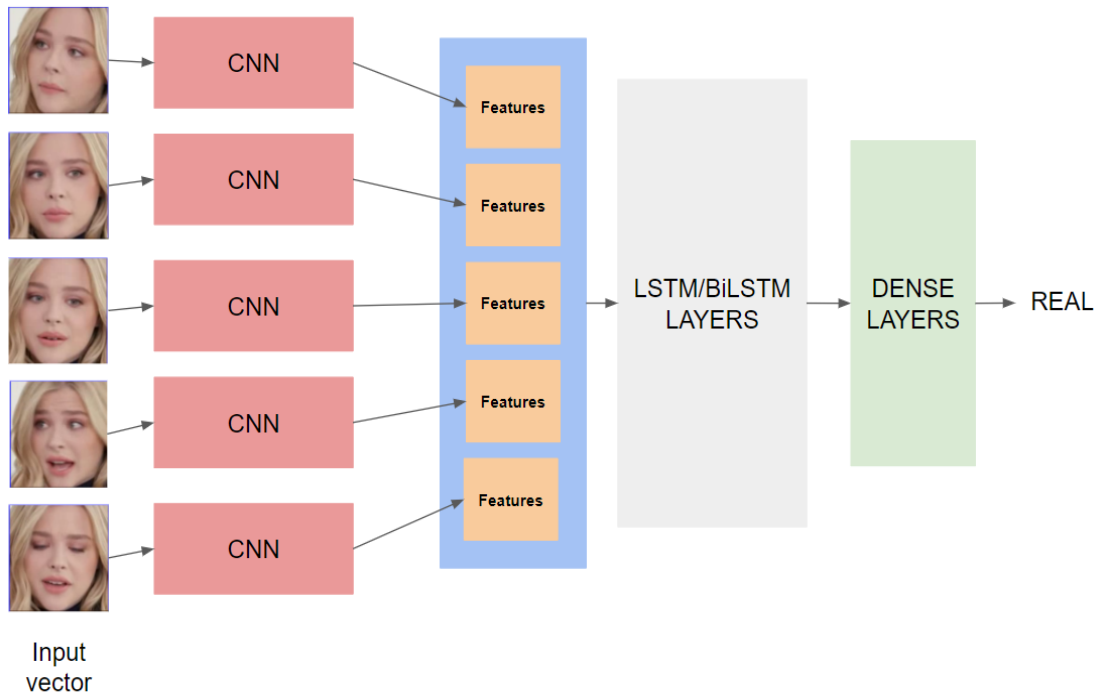


Fig. 3.1: Proposed System for Deepfake Detection

3.1.1. Network architecture

VGG Neural Networks, also referred to as VGGNets is an object-recognition model which supports up to 19 layers. It is also defined as a pre-trained Convolution Neural Network (CNN). VGG does not have any standard scientific abbreviation as it was named after the group that has invented it and stands for Visual Geometry Group. Its main feature is to distinguish the objects and classify unseen objects. Though it has been very precise with object recognition, it does not work well with scene recognition. Due to the achievements like discriminative nature of the decision function because of presence of more nonlinear rectification layers and improvement in error reduction with respect to depth of the network and also because of the controlled set of parameters for its depth, we preferred to use VGGNets as over other convolution networks.

We built two models, one with VGG16 [2] and other with VGG19 [2], respectively to identify any differences in the performances. Both the models are pre-initialized with ImageNet weights and in both of them; we tweak the weights with an intention to train the model to identify only those characteristics which are important to identify the differences between a real and a deepfake video. We extract only the features from the last max pooling layer (Pooling layer is one of the building block of CNN. It progressively reduces the spatial size of representation to reduce the parameters and computation in the network. It acts independently on each feature.) of the architectures, and provide this as input to the RNN. The model comparison of VGG16 and VGG19 is demonstrated in Table 3.1.

Table 3.1: Architectures of VGG16 and VGG19

VGG16	VGG19
ConvNet Configuration	
16 weight layers	19 weight layers
Input (224 x 224 RGB image)	
Conv3-64 Conv3-64	Conv3-64 Conv3-64
Maxpool	
Conv3-128 Conv3-128	Conv3-128 Conv3-128
Maxpool	
Conv3-256 Conv3-256 Conv3-256	Conv3-256 Conv3-256 Conv3-256 Conv3-256
Maxpool	
Conv3-512 Conv3-512 Conv3-512	Conv3-512 Conv3-512 Conv3-512 Conv3-512
Maxpool	
Conv3-512 Conv3-512 Conv3-512	Conv3-512 Conv3-512 Conv3-512 Conv3-512
Maxpool	
FC - 4096	
FC - 4096	
FC - 1000	
Soft-max	

To address the problems in regards to vanishing gradient and loss of information in feedback RNN's, LSTM's (LSTM is the acronym for Long Short-Term Memory. It is used in the field of deep learning and is an artificial recurrent neural network (RNN) architecture. It holds feedback connections. It processes not only single data point but also sequence of data (which basically includes images or videos). It is applicable to

tasks like unsegmented, connected handwriting and speech recognition. It usually is composed of a cell which remembers values over arbitrary time intervals and three gates (input, output, forget) which regulates the flow of information in and out of cell. To deal with vanishing gradient problem which is encountered when training traditional RNNs, LSTM networks were developed. These networks are suitable to classify, process and make predictions based on time series data.) use 'memory cell', with which it can keep information for longer durations. Hence, we chose to work with LSTM layers. Our architecture consists of two LSTM layers connected consecutively.

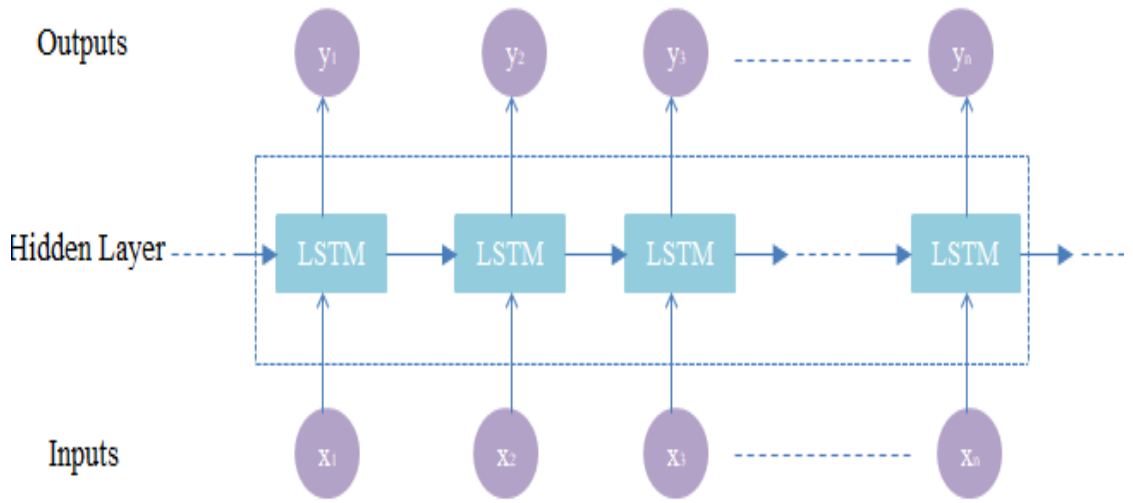


Fig. 3.2: Standard LSTM network

We have also used two different model architectures, alongside VGG16 and VGG19, instead of using a uni-directional LSTM, we now use a Bi-directional LSTM (also known as BiLSTM). BiLSTM are nothing but just putting together two independent RNNs. The structure of BiLSTMs allow it to obtain both forward and backward temporal information over the traditional uni-directional LSTM. We run the inputs in both forward and backward directions so that the network is able to store features from both past and future. They perform the task of understanding the context of the input sequence, much better than uni-directional LSTMs.

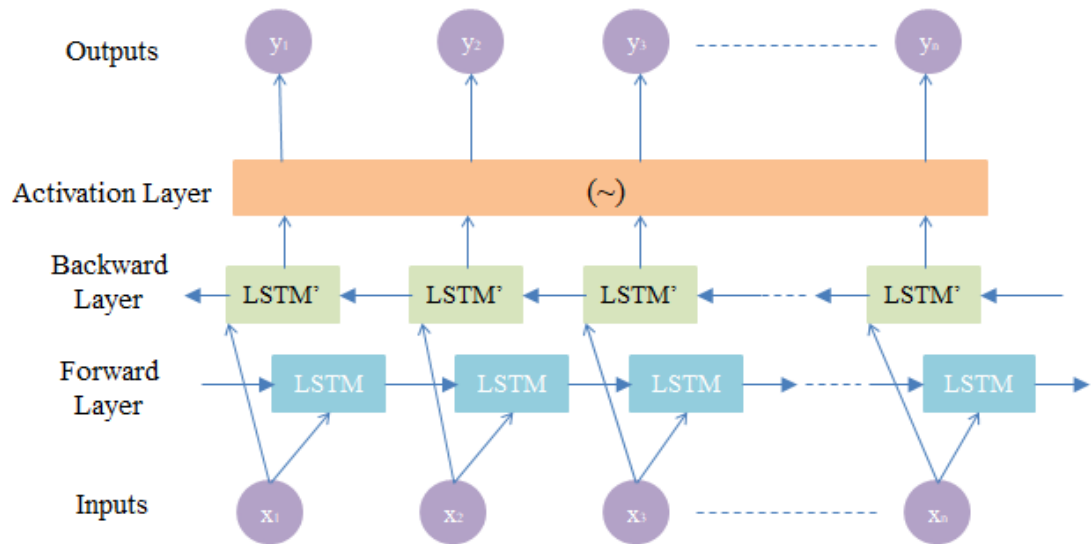


Fig. 3.3: Bi-directional LSTM

All these four models are now connected to a Dense network, followed by an output from a soft-max layer to obtain the probabilities for an input to be real or synthesized. Each layer has a dropout of 0.3, which makes the model competent.

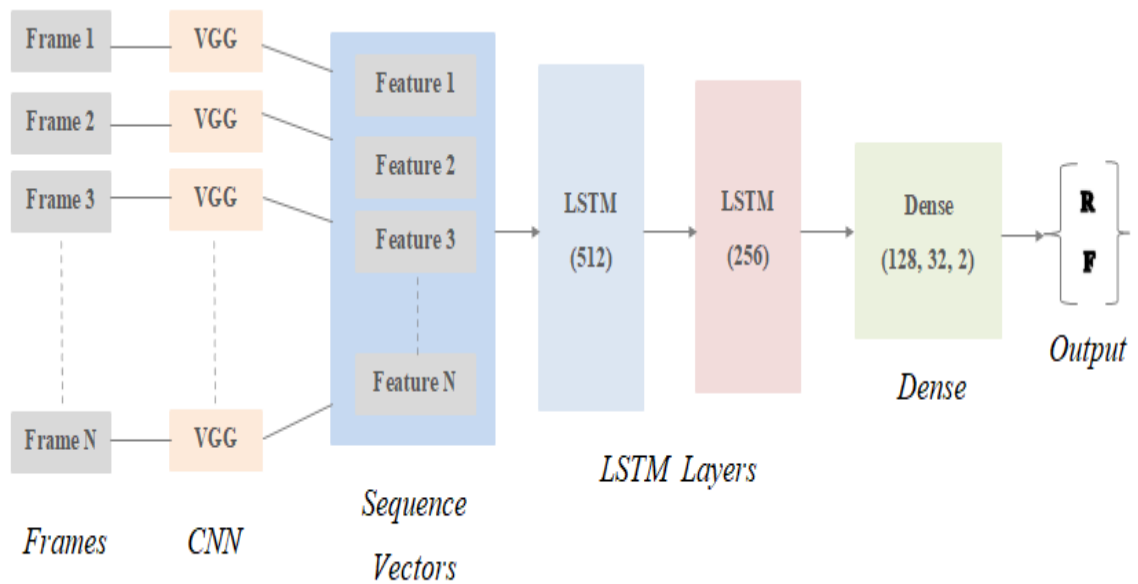


Fig. 3.4: Proposed model using LSTM

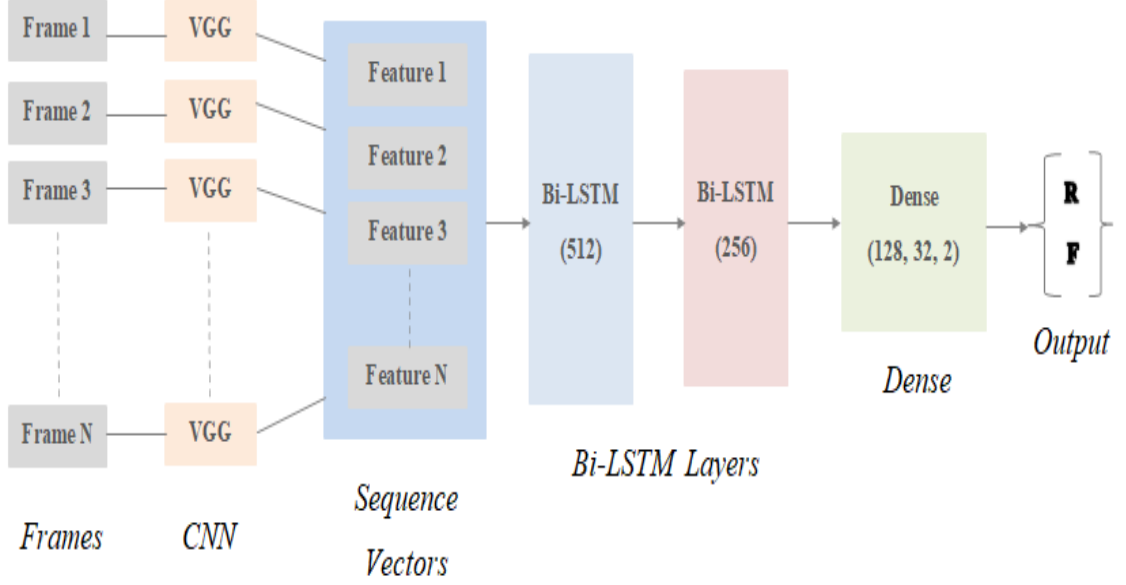


Fig. 3.5: Proposed model using Bi-LSTM

3.1.2. Dataset

The dataset Celeb-DF-v2 [16], used for training the model, is a vast collection of 6528 original and synthesized videos made by DeepFake Forensics, using a refined synthesis algorithm that reduces the visual artifacts, which were a problem in the previously generated datasets. Currently, this dataset has the highest benchmark scores in comparison to all the available datasets.

3.1.3. Data preprocessing

The dataset is preprocessed to extract only the faces from all the videos. We have achieved this by using two classifiers. The first one is a OpenCV classifier (OpenCV (Open Source Computer Vision Library) is an open source, highly optimized library that focuses on real-time application and was built to provide common base or foundation for applications and to accelerate the machine perception in commercial products. It has more than 2500 optimized algorithms including both classic and state of the art machine learning and computer vision algorithms.) and Haar Cascade classifier (Haar Cascade is

a machine learning object detection algorithm where a cascade function is trained using lots of positive and negative images (superimposing positive images over the negative images) which is mainly used for object detection. The training is usually done on a server and has several steps. By using high quality images and increasing the amount of steps or stages better results can be obtained.) to detect the faces on all the videos and cropping the video, thereby reducing the unnecessary details from every video. We also resize the resulting cropped videos to the input size of our CNN (224-by-224). The resulting videos are used as the dataset for training the model.



Fig. 3.6: Depiction of data preprocessing. The frames are taken as input and using OpenCV's Haar Cascades, we detect faces from the frames and crop them accordingly.

3.1.4. Parameter Settings

We have partitioned the dataset in an uneven split of 50%, 10% and 40% to create three disjoint sets, for train, validation and test respectively. We use these partition percentages to extract train, validation and test splits from both real and synthesized videos. Corresponding splits are merged to form the entire train, validation and test data. We also set the number of frames extracted from each video to 5.

Stochastic Gradient Descent [1], Gradient in terms means slant or slope of a surface. In such case, Gradient Descent directly indicates descending a slope on a surface to reach the lowest point. SGD can also be stated as stochastic approximation of gradient descent which is an iterative method for revamping an objective function with smoothness properties. Especially in high dimensional optimization, it reduces the computational burden. It provides faster iteration in trade for a lower convergence rate. SGD is common algorithm and very popularly used in machine learning algorithms. It is also considered as basis of neural networks. SGD works by randomly picking one data point from a complete dataset at each iteration to reduce data processing enormously. It is common to sample small number of data points rather than just one point at each step, that is called ‘mini-batch’ gradient descent which tries to strike a balance between the goodness of gradient descent and of SGD. The advantages include – easy fit into memory because only single sample is processed by the network, fast computation, updates to the parameters is more frequent, easy in obtaining local minimums of the loss function. This optimizer with a learning rate of 1e-3 and a decay of 1e-4 is used in this model.

Categorical cross entropy is a loss function that is used for single label categorization. This is when only one category is applicable for each data point. In other words, an example can belong to one class only. The loss function used in the model is Categorical Cross Entropy function.

$$L(y, \hat{y}) = - \sum_{j=0}^M \sum_{i=0}^N (y_{ij} * \log(\hat{y}_{ij}))$$

3.2. Requirements

This section includes all the software and hardware that is been used in the making of the proposed model and also briefs the working, features and functionalities of each software for a better understanding.

3.2.1. Hardware requirements

The proposed system will work well on

- I. A *CPU RAM* of 13GB or above
- II. A *GPU VRAM* of 16GB or above

3.2.2. Software requirements

3.2.2.1. Keras

API that is designed for human being not for machines is called Keras [3] (also referred to as a deep learning framework). It is used in reducing cognitive load -- provides simple and consistent API's, number of user actions required for use cases are minimized and provides a actionable error message. It is one among the top used frameworks in Kaggle the reason is, it empowers an individual to try more ideas than one's competition.

Built on TensorFlow 2.0, Keras can scale to large clump of GPU's or the entire TPU pod becoming the industry strength framework. It is the central part of tightly-connected TensorFlow ecosystem and almost covers every step of machine learning workflow. It has low level flexibility but high level convenience features to speed up experimental cycle. Keras supports all neural network models – be it embedded, recurrent, convolution, fully connected, pooling etc. Fig. 3.7 shows the Keras functionality to use in deep learning task.

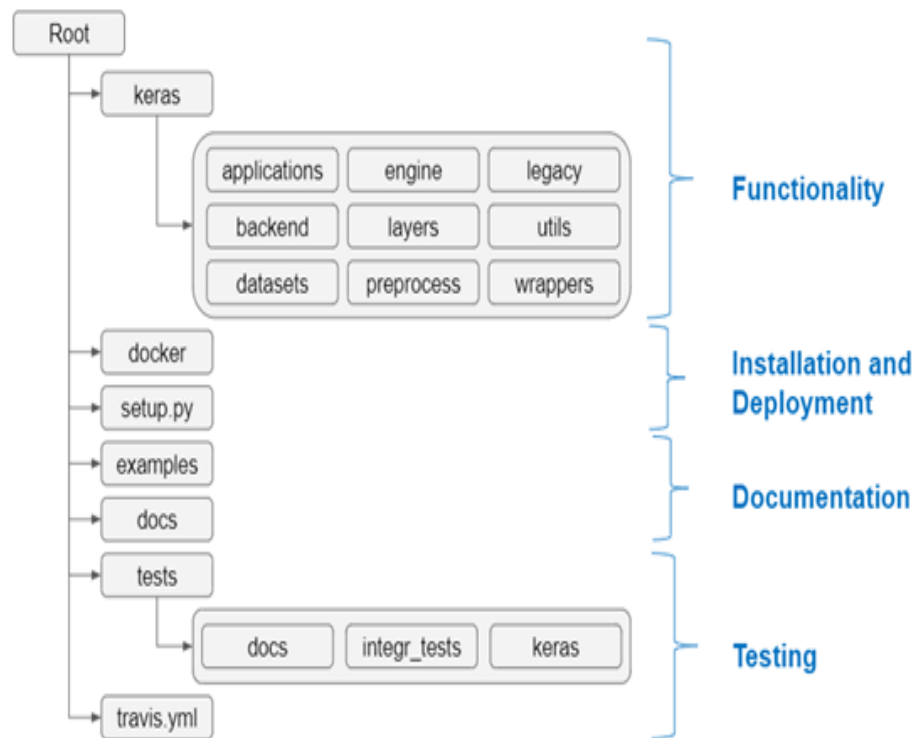


Fig. 3.7: Functionalities of Keras [18]

The amazing features of Keras include – prelabeled datasets, many implemented layers and parameters, multiple data preprocessing methods, model evaluation, modularity etc. Its focus on user experience and easy to use nature made it an accessible superpower and recommended as one of the best way of learning deep learning.

3.2.2.2. TensorFlow

TensorFlow [17] (also referred to as a symbolic math library) is an open source and free software library for dataflow and differentiable programming over a range of tasks. It has libraries, flexible ecosystem of tools and resources that let researchers push the state of the art in Machine Learning (ML). It helps to easily create and deploy ML powered applications using instinctive high level API's (like Keras) which leads to immediate model iteration and easy debugging.

It facilitates training and deployment of models in cloud no matter what language you use. It has a flexible architecture that takes ideas from notation to code to state-of-the-art and then to publish.

Features of TensorFlow include – Working with mathematical expression in multi-dimensional arrays has become more efficient, great support of ML and deep learning concepts, same code can be executed on CPU/GPU computing architectures, computation with high scalability across machines and large data sets. The training and deployment model of TensorFlow is shown in Fig. 3.8.

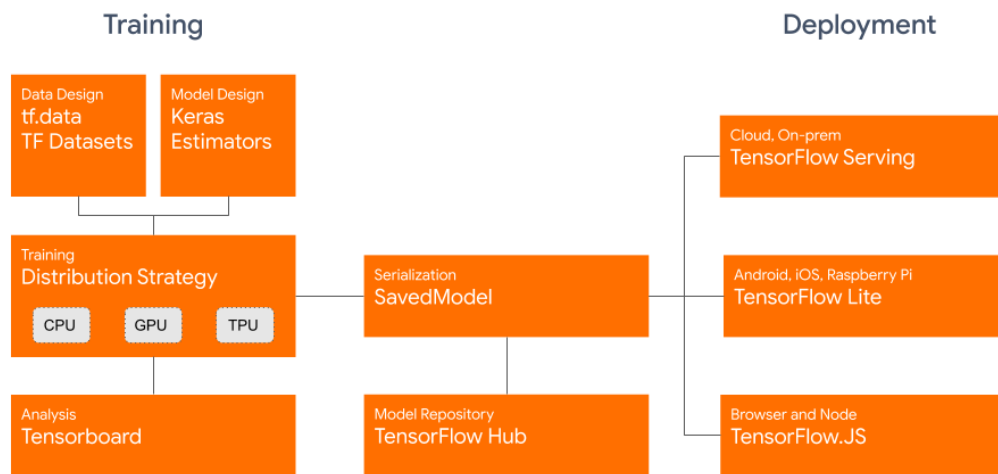


Fig. 3.8: Dataset / model is trained and deployed using TensorFlow [19]

Using TensorFlow leads to several advantages – flexibility, portability, production and research, performance and auto differentiation.

3.2.2.3. Anaconda

Anaconda [20] is an open source, free distribution of python and R languages for computing large collection of data aiming to simplify package management and deployment. It is a technology for real ML and data science applications. It is versatile and allows you to solve problems of any type. Catches vulnerabilities and provides

controlled access to data, models and packages. It supports innovation without sacrificing governance and security.

Anaconda's features include – compiled with latest python release, added new and improved packages, better reliability, enhanced CPU performance, easy to launch and configure a cloud based cluster, distributed computing services etc. Moreover, different features provided by Anaconda is depicted in Fig. 3.9.

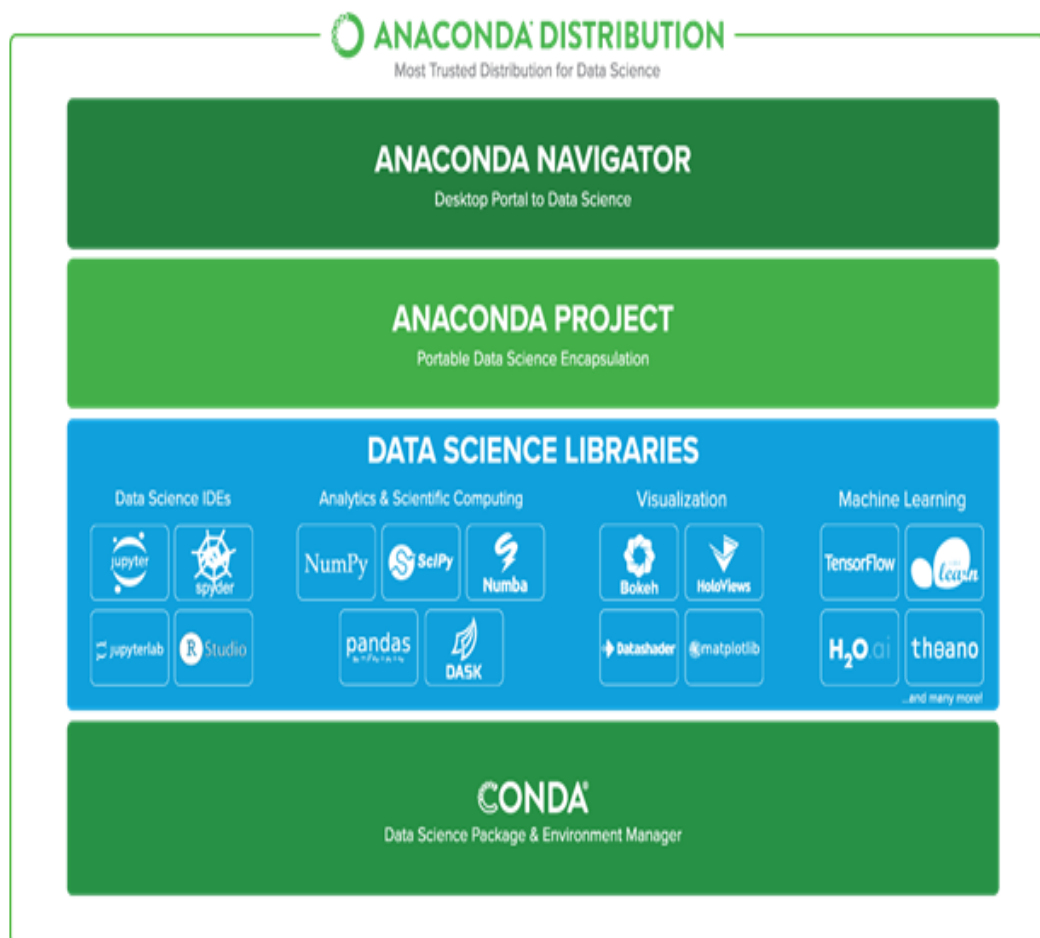


Fig. 3.9: Collection of features by Anaconda [21]

Using anaconda benefits in many ways – open source, can use/download more than 1500 python/R packages, simplified management of libraries, dependencies and environments, easy built and train mechanism, performing virtualization etc.

3.2.2.4. Kaggle Kernel (Online)

Kaggle kernel [22] (formally referred to as scripts) is a cloud based work-easel for data science and machine learning (ML). It allows data scientists to allowance code and analysis in R and python programming language. Over 150k code snippets (kernels) are shared on Kaggle covering anything and everything from segment analysis to object detection. The Kaggle kernel is demonstrated in Fig. 3.10.

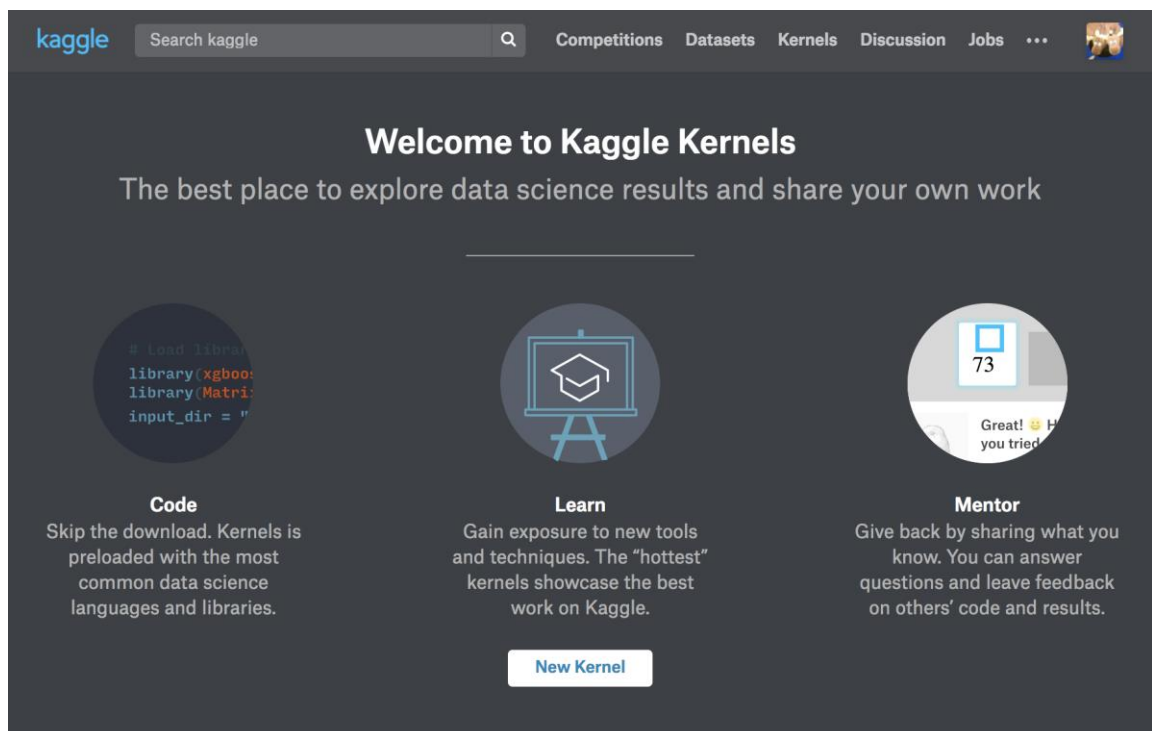


Fig. 3.10: Kaggle Kernels [23]

The kernels are no longer short scripts, they have been enhanced and improvised with a combination of code, input and output all stored together. In Kaggle, kernel is a requisite tool and core to the work that one does. Kaggle Kernels contain code which makes the entire model reproducible and allows collaborators when required. They run on docker containers.

The advantages of using Kaggle Kernel include – immense computation power, access to state-of-the-art data analysis and ML packages, no compatibility issues.

Results and Discussion

4.1. Results

While detecting Deepfakes, it's generally not the case that we find a video which is manipulated only in certain parts. So, we extract the frames from different partitions of the video and pass them to our models to detect if any tampering is done or not. This way of extraction is done for all the data splits sequentially. We presented the performances of our models in the Table 4.1 for number of frames equals to 5.

As we can see from the outcomes, our models are trying to predict whether the video has traces of manipulations or not. When observed keenly, the table shows the variation in train, validation and test accuracies when different model architectures are used. In this way models can be compared and the one that provides masterly accuracies can be used for further enhancement.

Table 4.1: Proclaiming the accuracies on our dataset splits for LSTM and BiLSTM models

Model	Training Accuracy (%)	Validation Accuracy (%)	Test Accuracy (%)
VGG16 LSTM	96.17	96.14	96.12
VGG19 LSTM	95.89	95.88	95.86
VGG16 BiLSTM	96.27	96.25	96.24
VGG19 BiLSTM	96.6	96.4	96.4

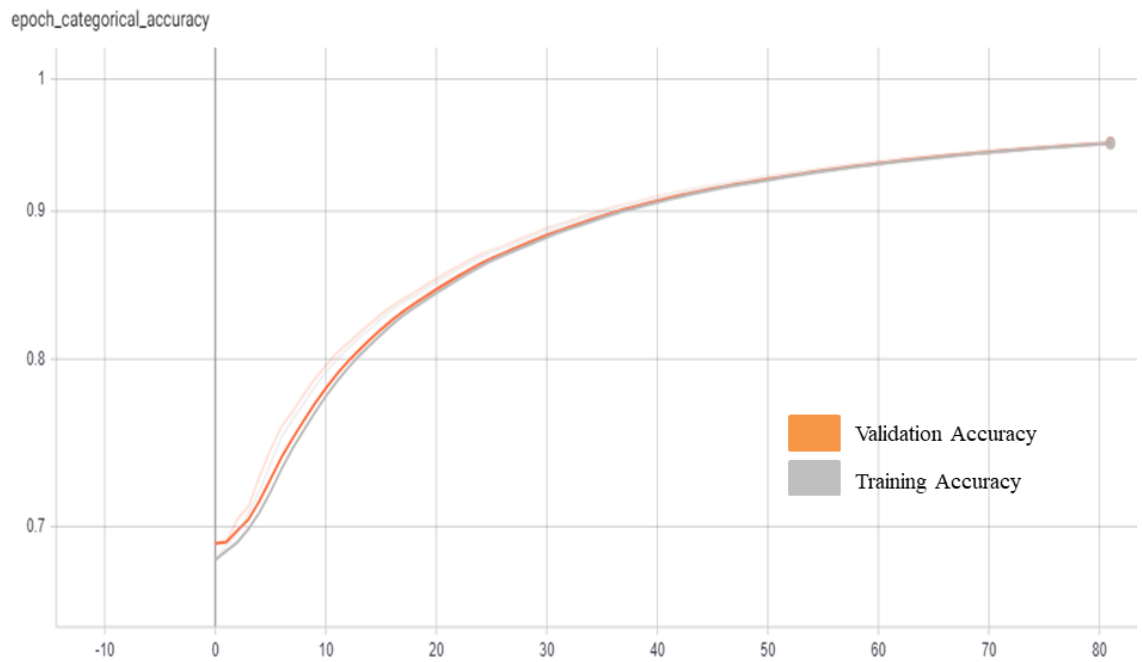


Fig. 4.1: Epoch categorical accuracy of the VGG-16 + LSTM model where it attains training accuracy of ~96% after 80 epochs

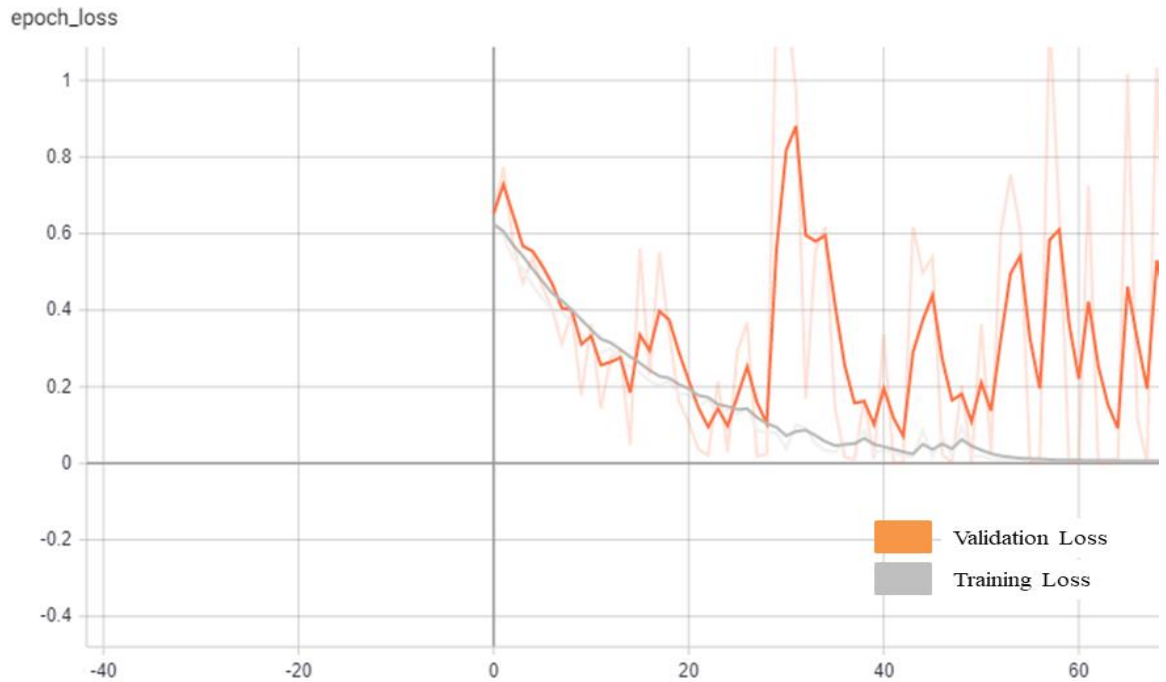


Fig. 4.2: Epoch loss of the VGG-16 + LSTM model where it achieves a loss value less than 0.1 after 45 epochs

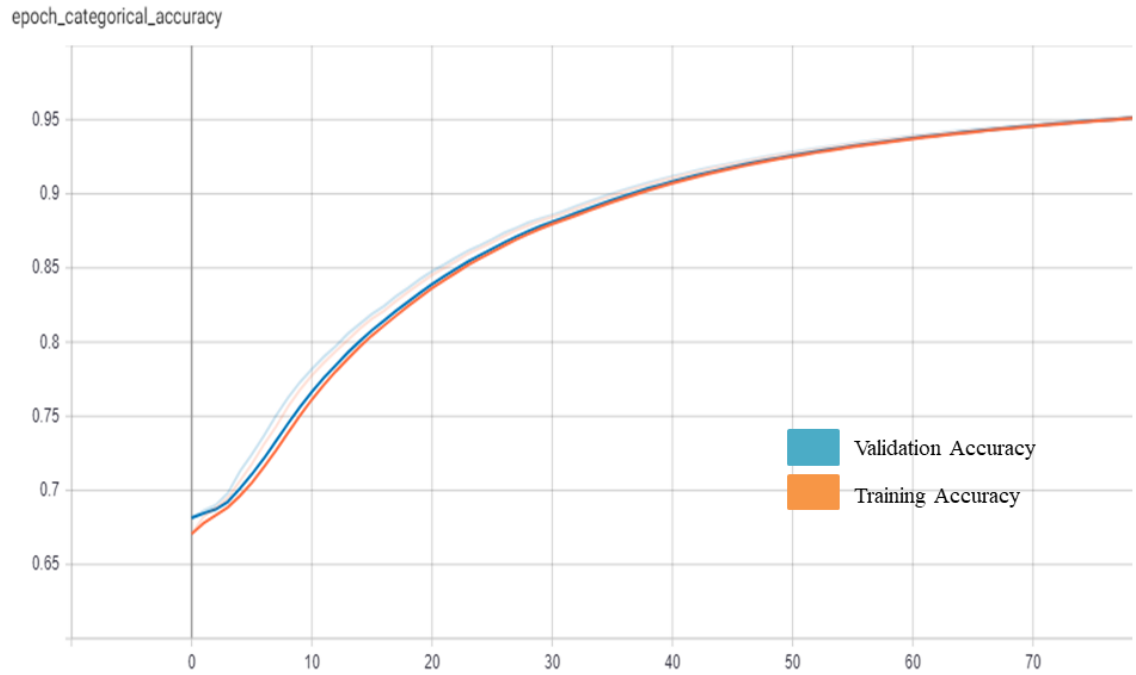


Fig. 4.3: Epoch categorical accuracy of the VGG-19 + LSTM model where it attains training accuracy of ~96% after 70 epochs

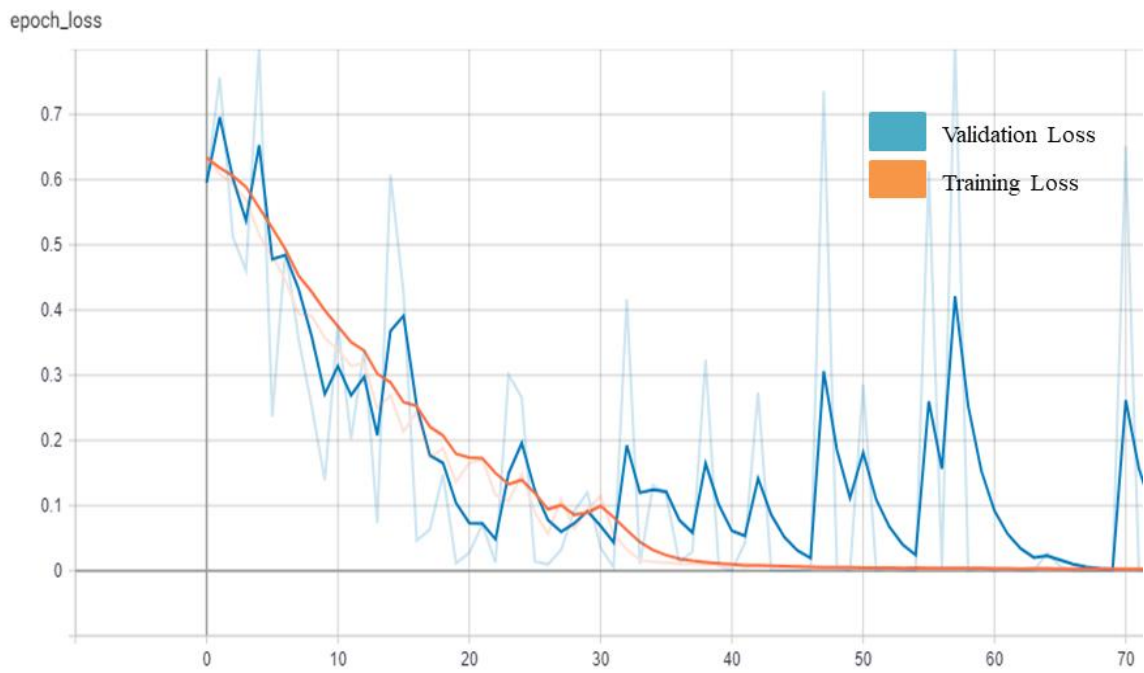


Fig. 4.4: Epoch loss of the VGG-19 + LSTM model where it achieves a loss value less than 0.1 after 35 epochs

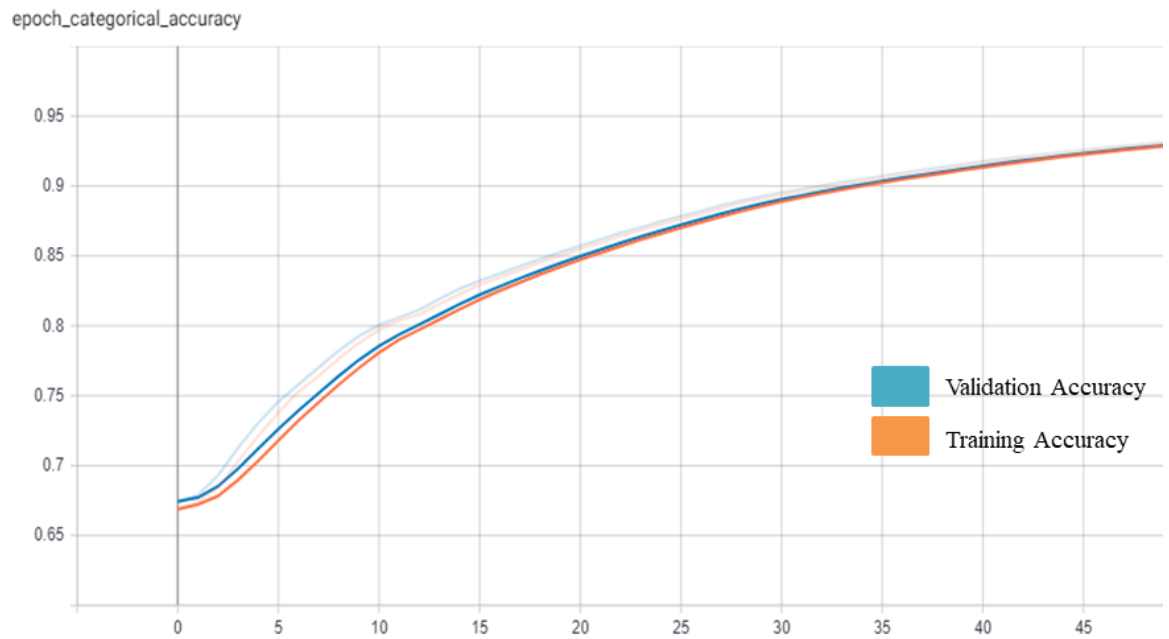


Fig. 4.5: Epoch categorical accuracy of the VGG-16 + BiLSTM model where it attains training accuracy of ~96% after 45 epochs

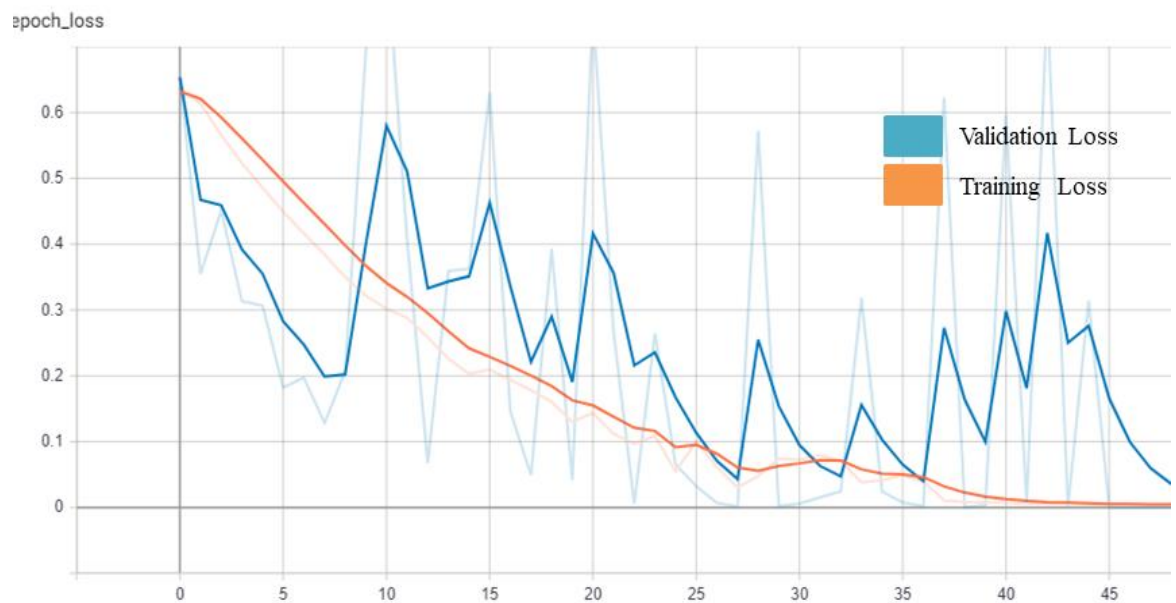


Fig. 4.6: Epoch loss of the VGG-16 + BiLSTM model where it achieves a loss value less than 0.1 after 25 epochs

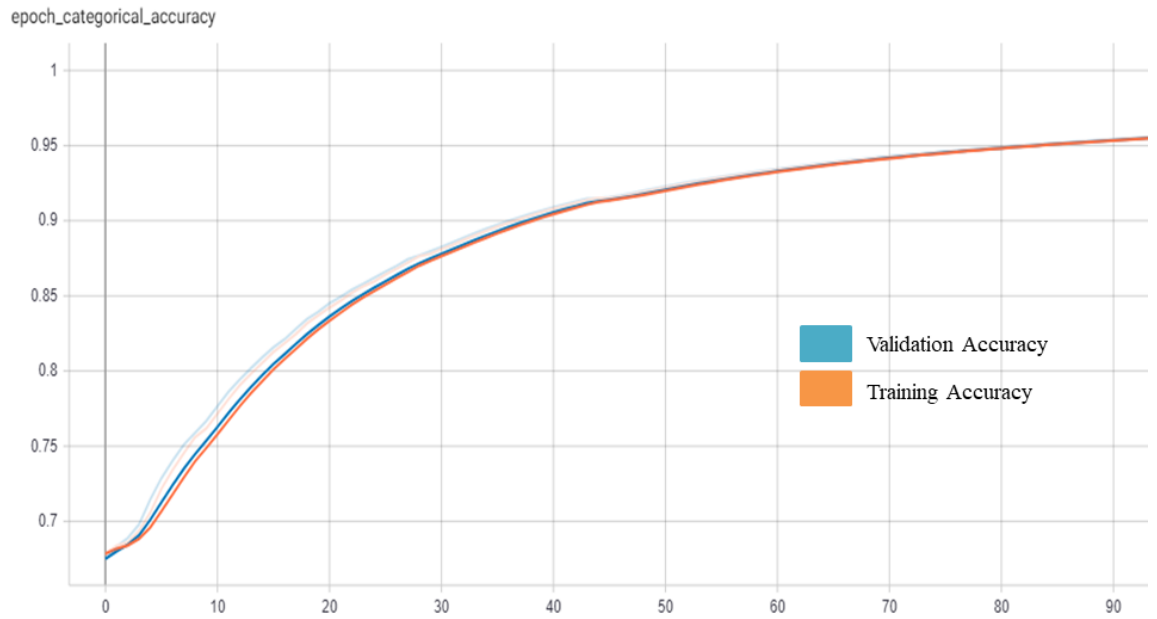


Fig. 4.7: Epoch categorical accuracy of the VGG-19 + BiLSTM model where it attains training accuracy of ~97% after 85 epochs

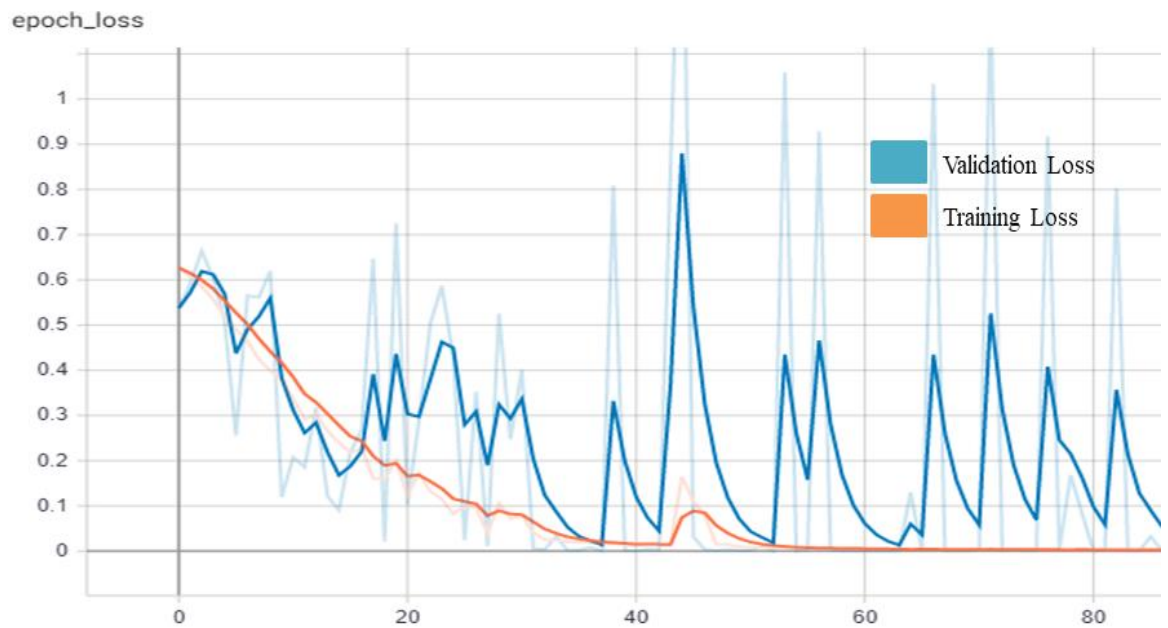


Fig. 4.8: Epoch loss of the VGG-19 + BiLSTM model where it achieves a loss value less than 0.1 after 30 epochs

We evaluated the performance of our models with some of the existing systems and some noteworthy observations. The outcomes of the comparisons were that although our model doesn't have the highest accuracy, the main achievement is that we are able to predict the manipulation of the video much faster than the existing models i.e., The attempted models can analyze fault just under 1 second part of the entire video (frame rate of the videos is 30 fps) with an acceptable accuracy of precisely 97%.

Table 4.2: Comparison of performance of our best working model with some of the existing models

Dataset	Method	Classifier	Dealing with	Test Accuracy
Face2Face	Optical flow	CNN / VGG16	Videos	81.61
Custom	Conv LSTM	CNN	Videos	97.1
Celeb-DF-v2	Conv BiLSTM	VGG19	Videos	96.4

4.2. Summary

Last few years, due to advancement in Deepfakes, i.e., rapidly mounting deep learning techniques that subject to manipulation of images and videos, made it easy to effortlessly generate and spread fake videos and images like never before. Phones with refined camera and easy reach to applications made creating misinformation simpler. This type of furtherance in technology made it possible for the fake digital content to barely leave any hints of being fake. The created misinformation can undoubtedly be very funny and interesting at times, but peeking towards the negative intent they carry and how dangerous it can be on public is the point one focuses on. So, there is an urgent need to create tools that can distinguish the difference between real and fake digital data (be it images or videos) to avoid the spread of fault information.

In the system which we proposed the dataset is pre-processed using OpenCV and Haar Cascade classifiers. After which, 'n' frames are extracted that are distributed across the video in a sequential order which are given as input to CNN to study and recognize features and characteristics of image frames in a digital input(video). These features are then sent to RNN to inspect the time based and sequential features for the provided input features of selected video. Here, we have preferred VGG when compared to others convolutional networks. Experimenting with both VGG16 and VGG19 have helped us understand how they work and how well and accurately they can provide results. We have included two layers of LSTM to reduce vanishing gradient and information loss. We have also made an attempt to check the accuracies by applying BiLSTM architecture to the proposed model. Finally, the output is obtained using a dense network. The proposed system efficiently produces accurate results in distinguishing the difference between real and digital misinformation.

The more sophisticated the Deepfakes will turn into from as of now, we would definitely like to make the required effort to condense and develop our model as per the requirements.

Conclusion and Further Enhancement

At first going through many problems that Deepfakes have caused, thoughtfully sitting and understanding them. Looking for and understanding the penned and experimented views of others, made us interested in swinging our stick into this problem.

Accordingly, we have presented a system, which detects the temporal and visual parameters from a video and detects its sanctity. Our experiments that can be applied on large sets of finagle videos showed us that by using convolutional LSTM and convolutional BiLSTM architectures, we can precisely predict if digital data (be it videos or images) are subjected to manipulation or not within just 5 frames from the video.

We are sure that our project will hold a good spot in the category of defense against fake digital misinformation. We also show that, the proposed models offers competitive outcomes in this field of research, while using simple deep learning architectures. Our future work would be in similar lines, but using other forms of data, like histograms to identify manipulations in videos.

References

- [1] Bottou Leon, “ Online Algorithms and Stochastic Approximations,” Online Learning and Neural Networks, Cambridge University Press, 1998.
- [2] Karen Simonyan, Andrew Zisserman, “ Very Deep Convolutional Networks for Large – Scale Image Recognition,” International Conference on Learning Representation, 2015.
- [3] Francois Chollet, Keras.io, MIT, 27th March, 2015.
- [4] Bapi Saha, Ajitesh Pal, Pratihari HK, “ Examination of Genuine and Fake Images by Histogram and Edge Detection Method – A Case Report,” Avens Publishing Group, Journal of Forensic Investigation, Vol : 5, Issue : 2, October 2017.
- [5] David Guera, Edward J. Delp, “ Deepfake Video Detection using Recurrent Neural Networks,” 15th IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS), Auckland, New Zealand, 2018. (Added to IEEE Xplore on 14th February 2019.)
- [6] Ekraam Sabir, Jiabin Cheng, Ayush Jaiwal, Wael AbdAlmageed, Iacopo Masi, Prem Natarajan, “ Recurrent Convolutional Strategies for Face Manipulation Detection in Video,” CVPR Workshop paper in Open Access vision, published and provided by Computer Vision Foundation, 2019.

[7] Charlotte Pelletier, Geoffrey I. Webb, Francois Petitjean, “ Temporal Convolutional Neural Network for Classification of Satellite Image Time Series,” <https://doi.org/10.3390/rs11050523>, MDPI and ACS style, Remote Sens., 11(5), 523, 4th March 2019.

[8] Zahid Akhtar, Dipankar Dasgupta, Bonny Banerjee, “ Face Authenticity : An Overview of Face Manipulation Generation, Detection and Recognition,” Proceedings of International Conference of Communication and Information Processing (ICCIP), 2019.

Available at SSRN: <https://ssrn.com/abstract=3419272>

<http://dx.doi.org/10.2139/ssrn.3419272>

[9] Yuezun Li, Siwei Lyu, “ Exposing Deepfake Videos By Detecting Face Warping Artifacts,” CVPR Workshop paper in Open Access vision, published and provided by Computer Vision Foundation, arXiv.org, 2019.

[10] Chia-Mu Yu, Ching-Tang Chang, Yen-Wu Ti, “ Detecting Deepfake-Forged Contents with Separable Convolutional Neural Network and Image Segmentation,” arXiv.org, December 2019.

[11] Irene Amerini, Leonardo Galteri, Roberto Caldelli, Alberto Del Bimbo, “ Deepfake Video Detection through Optical flow based Convolutional Neural Network,” IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Korea, 2019. (Added to IEEE Xplore on 05th March 2020)

[12] Steven Fernandes, Sunny Raj, Eddy Ortiz, Iustina Vinitila, Margaret Salter, Gordana Urosevic, Sumit K. Jha, “ Predicting Heart Rate Variations of Deepfake Videos using Neural ODE,” IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Korea, 2019. (Added to IEEE Xplore on 05th March 2020)

- [13] Jianxin Wu, “ Convolutional neural networks,” LAMDA Group, May 14, 2020.
- [14] Shruthi Agarwal, Tarek El-Gaaly, Hany Farid, Ser-Nam Lim, “ Detecting Deep-Fake Videos from Appearance and Behavior,” arXiv.org, 2020.
- [15] Luisa Verdoliva, “ Media Forensics and Deepfakes : An overview,” arXiv.org, 2020.
- [16] Li Y, Yang X, Sun P, Qi H, Lyu S. Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2020 (pp. 3207-3216).
- [17] Tensorflow.org, <https://www.tensorflow.org/guide>
- [18] se.ewi.tudelft.nl, <https://se.ewi.tudelft.nl/desosa2019/chapters/keras/>
- [19] Towardsdatascience.com, <https://towardsdatascience.com/getting-your-hands-dirty-with-tensorflow-2-0-and-keras-api-cc8579eb0915>
- [20] Anaconda.com, <https://www.anaconda.com/library>
- [21] Medium.com, <https://medium.com/@kumarankita764/new-features-of-anaconda-5-3-5bfdfe9b4240>
- [22] Kaggle.com, <https://www.kaggle.com/>
- [23] Towardsdatascience.com, <https://towardsdatascience.com/introduction-to-kaggle-kernels-2ad754ebf77>

Acknowledgement

It is my pleasure to acknowledge the roles of several individuals who were instrumental for completion of my final year project. First of all, I would like to express my gratitude to Dr. Jitendra V. Tembhumne, who encouraged me to pursue this project and taught me the art of proper engineering. I truly enjoyed working in an environment that stimulates original thinking and initiative which he created. His skillful guidance, willingness to help solve most difficult issues, innovative ideas and stoic patience are greatly appreciated.

I would like to acknowledge the support from my classmates G. Shiva Kumar and Mayur Selukar for their valuable discussions and inputs that helped to shape this project and will be grateful for their never ending moral support.

The acknowledgements would not be complete without mentioning my friends – Kavya Koppoju, E.Hari Kishan, L. Sathvik and Sri Vikas for their helpful suggestions. It is a great pleasure to see their interest in my work and I truly appreciate their ideas. My deepest appreciation belongs to my family for their patience and understanding.