**DEEP FAKE ANALYSER**

### A Project Progress Report- 2

### *Submitted in the partial fulfilment for the award of the degree of*

### BACHELOR OF ENGINEERING

### IN

### ARTIFICIAL INTELLIGENCE AND

### MACHINE LEARNING

### Submitted by:

### Arnav R Patnaik

### 20BCS6658

### Himanshu Gidwani

### 20BCS6637

**Aryan Gupta**

**20BCS6656**

**Sahildeep Kaur**

**20BCS6587**

### Under the Supervision of:

**MRS.YASHIKA SHARMA**

### SUPERVISORS NAME

**MR. AMIT GARG**

# DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING APEX INSTITUE OF TECHNOLOGY

### CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413,

**PUNJAB**

#### MONTH & YEAR

**15 APRIL, 2022**

**DECLARATION**

I, **Arnav R Patnaik**, **Himanshu Gidwani**, **Aryan Gupta** and **Sahildeep Kaur**, students of **‘Bachelor of Engineering in Artificial Intelligence and Machine Learning**, **session: 2020-2024**, Department of Computer Science and Engineering, Apex Institute of Technology, Chandigarh University, Punjab, hereby declare that the work presented in this Project Work entitled ‘**Deep Fake Analyser’** is the outcome of our bona fide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. It contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgement has been made in the text.

### [20ONAIML2-A]

### (Arnav R Patnaik)

### Candidate UID: 20BCS6658

### (Himanshu Gidwani)

### Candidate UID: 20BCS6637

(**Aryan Gupta**)

Candidate UID: **20BCS6656**

(**Sahildeep Kaur**)

Candidate UID: **20BCS6587**

#### Date:15 April 2022

**Place: Chandigarh University, Mohali, Punjab**

**ii**

**ABSTRACT**

**With the speedy growth in the technology and automation sectors, different techniques have been developed which can easily manipulate multimedia content such as videos and images with the ultimate level of realism. It becomes very hard for a person to find out which of the media content is real and which is fake. With the help of deep learning and GANs (Generative Adversarial Networks), deep fakes can be produced. A Deepfake is a fake video or image where a person in an existing video or image is replaced by someone else by using an algorithm. If we look at the positives, we can see how deep fake technology can be used in advertisements, filmmaking, and video games. But on the contrary, it also results in the spread of fake news and the generation of pornographic content. This seriously impacts society and presents enormous threats. This research paper aims to present a project based on a deepfake analyzer that will be able to identify and recognise fake media with the help of deep neural networks.**

Keywords— deepfake, media content, fake news, manipulate, deep learning, GANs

**ACKNOWLEDGEMENT**

I would like to thank my supervisor, Mr Amit Garg sir, for his guidance and advice through each stage of making this project. I would also like to thank Mrs Yashika Sharma ma’am for giving me and my teammates Mr Himanshu Gidwani, Mr Aryan Gupta and Ms Sahildeep Kaur this opportunity to work on a minor project in which we can show our true potential, creativity, and hard work. I would also like to thank my family and friends, who have been a constant support and have always motivated me to work hard and bring out the best in me. This project’s success and end necessitated a great deal of direction and assistance from many people, and we are extremely fortunate to have received it all as part of the project's completion. We owe everything we've accomplished to their oversight and help, and we'd like to express our gratitude.

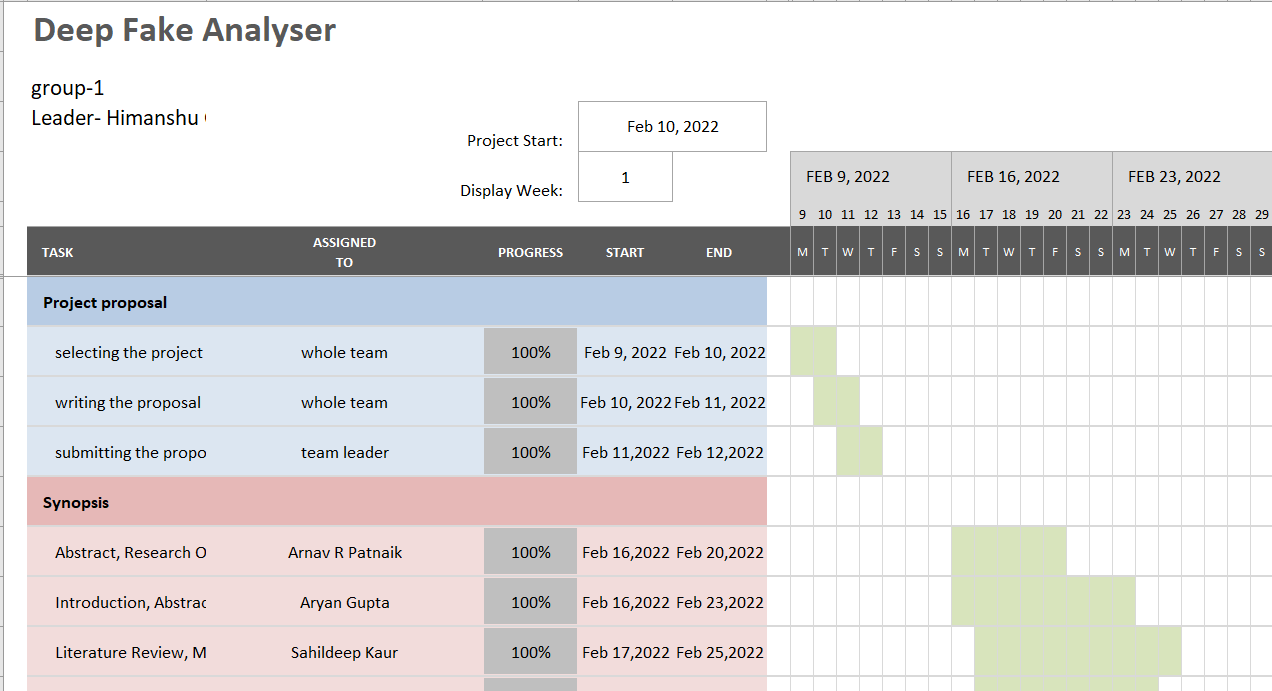
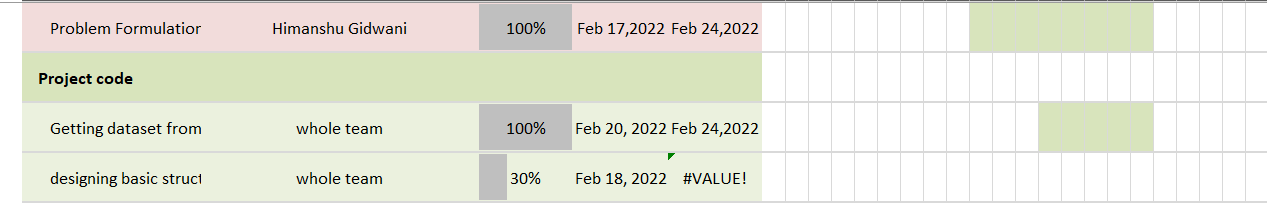
**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| ***Figure*** | ***title*** | ***page*** |
| 1 | Comparison between an original and deep faked image | 7 |
| 2 | Deepfake of politician’s speech going viral | 7 |
| 3 | Flowchart of Deep Learning process | 8 |
| 4 | Deep neural network chain | 9 |
| 5 | System Architecture of the project | 12 |
| 6 | System flow diagram of DeepFake Analyser | 15 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| ***Figure*** | ***title*** | ***page*** |
| 2.1 | Existing System Summary | 13 |

**TIMELINE/GANTT CHART**

# Table of Contents

|  |  |  |
| --- | --- | --- |
| **SR.NO** |  | **PAGE NUMBER** |
| 0 | Title page  Declaration of the Student  Abstract  Acknowledgement  List of Figures  List of Tables  Timeline / Gantt Chart | 1  2  3  4  5  5  5 |
| 1 | **INTRODUCTION\***   * 1. Problem Definition   2. How do Deep Fakes work?   3. How datasets are constructed for deepfake detection?   4. Problem Overview   5. Hardware Specification   1.5.1 PC  1.6 Software Specification | 7  7-9  9  9-10  10  10-11 |
| 2 | **LITERATURE SURVEY**   * 1. Existing System Summary | 11-13 |
| 3 | **PROBLEM FORMULATION** | 14 |
| 4 | **RESEARCH OBJECTIVES** | 14 |
| 5 | **METHODOLOGY** | 15 |
| 6 | **TASKS ACCOMPLISHED** | 15-17 |
| 7 | **ONGOING AND UPCOMING TASK** | 17 |
| 8 | **CONCLUSIONS AND DISCUSSION** | 17 |
| 9 | **REFERENCES** | 18 |

# INTRODUCTION

#### PROBLEM DEFINITION

#### A deep fake is a fake, manipulated video or image produced by using deep learning and machine learning techniques that typically features a person’s likeness and voice in a scenario that did not occur to produce false results. This technology can create revenge porn that causes distress to the victim, which includes anger, guilt, paranoia, depression, or even suicide. People use this technology to spread misinformation and false narratives about people.

#### Deepfake technology makes it easy to swap the faces of people in videos or images, so people use it to create videos of trusted people to make other people fall for scams or steal their data or money.

#### 



1. Comparison between an original and deep faked image

#### As shown in the above figure, Facebook's CEO, Mr Mark Zuckerberg, was the subject of a deepfake video in which he appeared to credit the success of the social platform Facebook to a secretive organization. [3]



1. Deepfake of politician’s speech going viral

The image above is a deep fake video of Mr Donald Trump, the former US president, that became popular on the internet. In a roughly 30-second video, Mr Trump claims in a robotic-sounding voice that he will return to social media by joining RuTube, the Russian edition of the video hosting site Youtube. **[4]**

#### HOW DO DEEP FAKES WORK?

Over the years, a large amount of data has been generated which is publicly available on different internet sites, social media, etc. With faster development in machine learning and deep learning techniques like Generative Adversarial Networks, producing realistic fake media has become pretty easy.**[1]**

 Different deepfake techniques can be implemented for manipulating the faces of people. The most commonly used manipulation techniques we see are identity swap, expression swap, and complete face synthesis. We provide the manipulation techniques with larger sets of data that are available publicly.

 Machine learning has a crucial role in deepfakes. Machine learning methods to make deepfakes are built on deep learning techniques and neural networks. The first step of deep faking itself involves training a neural network, which is known as a subset of machine learning, from which we give the model a clear understanding of what a person in an image or video looks like in different conditions like different angles, lighting etc. With the help of machine learning, the model used in deepfake is divided into training and testing. On one hand, model training on a dataset is performed to generate fake videos, while on the other hand, detection is performed to detect the videos.

A very important concept that makes this possible is deep learning. Deep learning is a specialised machine learning technique that instructs and teaches the computer to do what comes to humans naturally. In deep learning, a computer model can classify different objects and tasks just by directly accessing the images, videos (which are broken down into different frames), texts, and sounds. Models learn straight from the data without going through the traditional feature extraction method. Large sets of labelled data and neural network architecture are being utilised to train the models. In contrast, deep learning also requires a good amount of computational power. Here, high-performance GPUs come in handy as they provide a parallel architecture that works efficiently for deep learning. The advantage that deep learning offers over others is that as the amount of data increases, the deep learning models improve as well. This makes Deep Learning achieve greater heights, sometimes outdoing human-level performance.

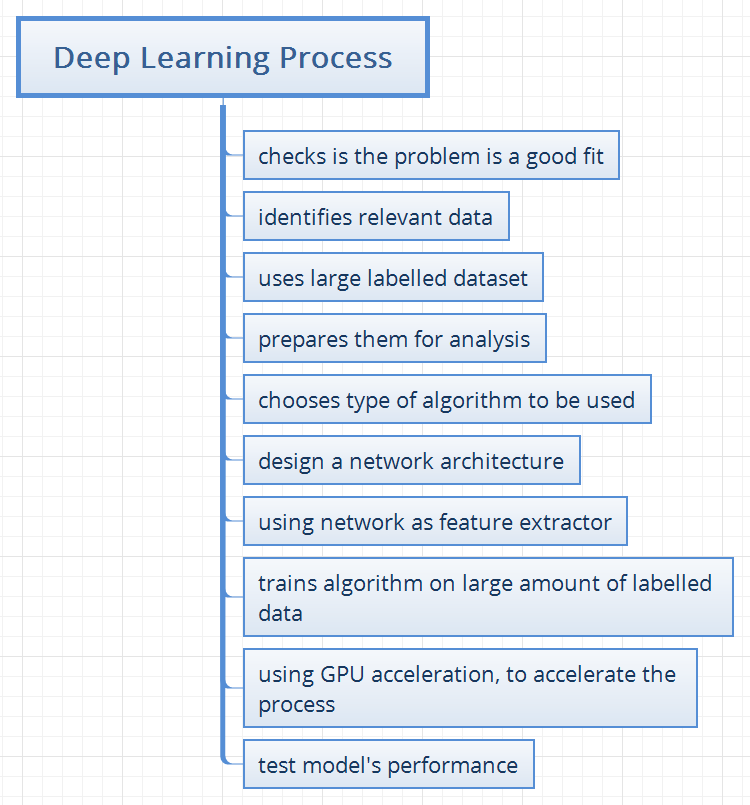


Fig 3: Flowchart of Deep Learning process

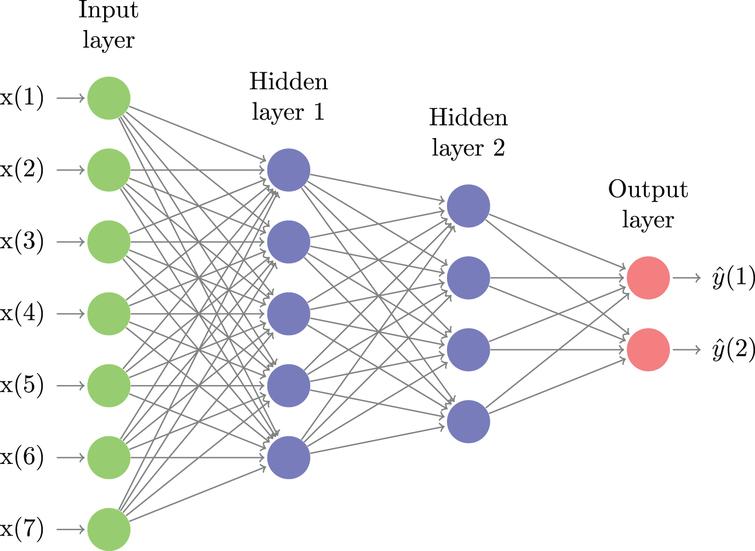


Fig 4:Deep neural network chain **[2]**

A deep neural network is a system that uses many layers of nodes to perform high-level functions from the input that is being provided. A convolutional neural network is used for image recognition purposes and plays a significant part in deep fake detection as well, where we have to detect the image and video frame by frame. Convolutional neural networks are the workhorse behind a lot of what we call artificial intelligence today. There are computational pieces that make them work, and they are called convolution operations. To perform a convolution, we need an image and a kernel image. An image is a two-dimensional array of numbers. The first step that we do is turn the image in the kernel by 180 degrees. This kernel also maps the numbers to colours. The convolution takes the reverse kernel and moves it row by row and column by column, matching it to that patch in the image, multiplying all the elements one at a time by each other and adding up the result. This kernel is the detector of the feature. That is the process of convolution. A convolutional neural network will learn these kernels based on what is more useful and will try to get a better result. A convolutional neural network (CNN) is a type of deep learning algorithm that takes an image as input, assigns learnable weights and biases to objects in an image, and can separate them from one another. These "deep fakes" are hyper-realistic videos and images that apply artificial intelligence to depict someone saying and doing things that have never occurred. Deep fakes are the product of Generative Adversarial Networks (GANs) namely two neural networks working together to generate realistic-looking media. This synthetic content is difficult to detect as it uses real footage and uses authentic-sounding audio.

#### 1.3 HOW DATASETS ARE CONSTRUCTED FOR DEEPFAKE DETECTION?

To detect deepfakes, various datasets consisting of real and fake videos from several sources are made. Deepfake detection dataset necessitates extensive data collection and manipulation. To build a dataset for deepfake detection, high-resolution images of various people with lots of poses and expressions are collected. The dataset includes various alterations to a single video. The dataset for deepfake detection available from Kaggle consists of a total of 100,000 clips from 3,426 paid actors using various deepfake methods. Some datasets are gathered from real videos on YouTube, and then deepfakes are generated from them with different kinds of methods. The Face forensics dataset and the Celeb-DF dataset are all constructed from real videos on YouTube. Some other datasets, like deepfake in the wild (2019), are constructed from deepfakes available online. Some datasets, like DF Timit (2018), which used face swap-GAN and Google deepfake detection (2019), are made by hiring actors and actresses. A real video dataset is made by this and then deepfakes are generated from them.

#### 1.4 PROBLEM OVERVIEW

* Obtain a deep fake dataset from Kaggle and refine it with real-world samples. This aids in the development of a better feature selection system.
* Develop a deep learning and neural network-based algorithm to achieve optimum accuracy.
* We'll add another function to the algorithm that will present the analysis as a bar chart, line chart, or pie chart, giving the user a better understanding of how much the number of deep fakes has grown and how they differ based on their confidence scores.
* Anvil, a free Python-based drag-and-drop web app builder, will be used to create the GUI.

#### 1.5 HARDWARE SPECIFICATION

#### 1.5.1 PC

A PC is a personal computer that can be utilised for multiple purposes depending on its size, capabilities, and price. They are to be operated directly by the ultimate consumer. Personal computers are single-user systems and are portable. Our web application program will be installed on the PC for our clients to use it. This makes it feasible for individual use.

#### SOFTWARE SPECIFICATION

### The software tools that will be utilised in the development of this project are as follows:

|  |  |  |
| --- | --- | --- |
| Software tool used | Description | Logo |
| Jupyter Notebook | Jupyter Notebook is a web-based open-source application that is used for editing, creating, running, and sharing documents that contain live codes, visualisations, text, and equations. Its core supported programming languages are Julia, R, and Python. The Jupyter notebook comes with an IPython kernel that allows the programmer to write programs in Python. There are over 100 kernels other than IPython available for use. |  |
| Atom text editor | Atom is a text and source code editor which works across all operating systems. It speeds up find-and-replace operations by an order of magnitude and improves loading performance for large, single-line files. It’s a desktop application built with HTML, JavaScript, CSS, and Node.js integration. |  |
| Visual Studio Code | Visual studio code is an open-source code editor built for Windows, Mac OS, and Linux which can be used for various programming languages like Java, JavaScript, Python, C, C++, Node.js. It contains support for multiple features like debugging, syntax highlighting, code snippets, code refactoring and git command built-in. |  |
| Anvil | Anvil is a simple way to develop web applications. It consists of a drag and drop design, built-in storage, simple programming in python and easy integration with external services. |  |

# LITERATURE REVIEW

Deep Fake Detection using Inconsistent Head Pose:

**[6]** This project is proposed by Xin Yang, Yuezen Li, and Siwei Lyu. Here synthetic content like deep fake videos and images is detected using inconsistent head poses. The 3D poses correspond to rotation and the translation of worlds coordinates to the corresponding camera coordinates. Here R is a rotation matrix that is of 3x3 order and t is a 3x1 translation vector. In 3D head poses estimation we need to solve the reverse of R and t by using 2D image coordinates. As per the model SVM that is Support Vector Machine is selected to detect synthetic contents. SVM is trained to classify based on differences between head pose estimated using a full set of facial landmarks and those in central face regions to differentiate. Here training and testing data involve a dataset of deep fake and real images and videos. Yang and Li in the year 2018 released an article regarding this topic. The topic released by Yang focuses on analysing synthetic content based on head pose estimation. A technique like this involves a classifier like SVM (Support Vector Machine). The parameters that were used to evaluate this model or design include complexity and accuracy.

Deep Fake Detection:

**[7]** This project is implemented by Piyush Chandra during the deep fake detection Challenge that is also known as DFDC. This project can be found on Kaggle This model is designed to detect deep fake videos. To achieve this technology face recognition is used that will capture the video in particular frames thus giving time to the model to detect whether it is real or fake. For a baseline, the submission facenet is used. Facenet is a face recognition pipeline that learns a mapping from faces to positions in multidimensional space where the distance between the points directly corresponds to a measure of face similarity. The final output that is whether the video is real or fake has been provided by plotting a histogram. The plotted histogram provides us with the confidence level that whether the model is confident about the prediction that has been made. The parameters that were used to evaluate this model or design include complexity and accuracy. But this model lacks a graphical user interface, which just makes it a simple text-based program. Besides this, the final output was not clear as it was provided in the form of a graph.

Deep Learning Deep Fake Detection (ResNext and LSTM):

**[5]** This project was designed by a college student team. The technique that they adopted is called LSTM, a neural network. Aside from neural networks, web applications were also created using the Django application. The parameters that were used to evaluate this model or design include complexity and accuracy. This project is designed to detect videos that fall under the category of "deep fakes." Here, deep learning techniques like LSTM and ResNext CNN are used to obtain a feature vector. A further LSTM layer is trained using these features. The system architecture of this model includes the upload action of videos. Then comes the pre-processing part, where the splitting of data takes place. This is followed by a deep fake detection model that includes ResNext Feature extraction and LSTM video classification. After deepfake detection comes the model evaluation, where a confusion matrix is used that gives us our final and required output in the form of a label that says whether the video is authentic (which is original), or synthetic (which comes under the tag fake).

A website has been formed by making use of the Django application to create a real product. Django is a high-level Python web framework that encourages rapid development and clean code. Django is a collection of Python libraries allowing you to efficiently create a quality web application and is suitable for both the backend and the frontend.

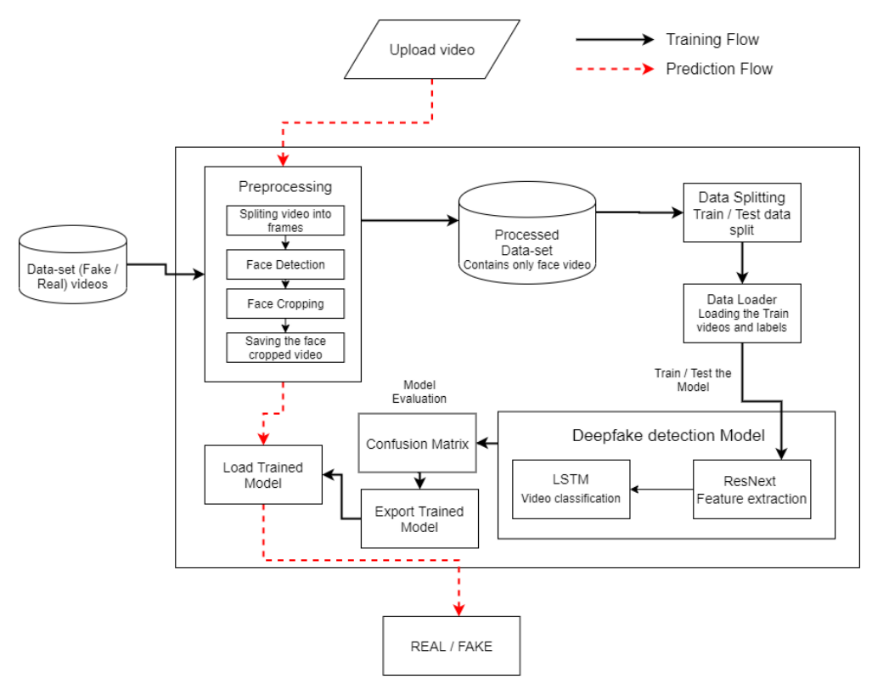


Fig 5: System Architecture of the project

**2.1 Literature Review Summary**

Table 2.1: Literature review summary

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Year and citation** | **Article Title** | **Purpose of study** | **Tools / Software Used** | **Comparison of technique** | **Source (Journal / Conference)** | **Findings** | **Data set (if used)** | **Evaluation parameters** |
| Yang, X., Li, Y. and Lyu, S. (2018). Exposing Deep Fakes Using Inconsistent Head Poses. *arXiv:1811.00661 [cs]*. [online] Available at: https://arxiv.org/abs/1811.00661 | Exposing deep fakes using inconsistent head pose | It focuses on analyzing deep fake contents using inconsistent Head poses estimation. In this kind of technique classifier like SVM is used. | Inconsistent head poses estimation | * Deep Learning * Head pose estimation | https://arxiv.org/abs/1811.00661 | Fake image and videos detection | paperswithcode.com/dataset/faceforensics-1 | * Complexity * accuracy |
| kaggle.com. (n.d.). *Deepfake Detection*. [online] Available at: https://www.kaggle.com/piyush357/deepfake-detection. | Deepfake Detection | Used to detect synthetic video by making use of face recognition. | * Face recognition * Jupyter Notebook | Deep Learning | https://www.kaggle.com/piyush357/deepfake-detection | Detection of fake images and videos | piyush357/deepfake-detection/data | * Complexity * accuracy |
| Jadhav, A.H. (2022). *Deepfake detection using Deep Learning (ResNext and LSTM)*. [online] GitHub. Available at: https://github.com/abhijitjadhav1998/Deepfake\_detection\_using\_deep\_learning [Accessed 26 Feb. 2022] | Deepfake detection using Deep learning (ResNext and LSTM) | Detects deep fake video by making use of a neural network like LSTM. Also involves Django application to create a web application. | * Django Application * Jupyter Notebook | * Deep learning * LSTM | https://github.com/abhijitjadhav1998/Deepfake\_detection\_using\_deep\_learning | Fake images and videos are detected. | piyush357/deepfake-detection/data | * Complexity * accuracy |

# PROBLEM FORMULATION

### The major problem arising in the project work is that we do not have enough datasets for real and deepfake prediction. We should have more datasets to improve the model's prediction accuracy. It's critical to overcome this problem since it won't be able to anticipate real-world deepfake pictures because the vast majority of the deepfake dataset comes from pornographic sites or a specific portion thereof. When a person submits an image to see if it is real or a deep fake, the problem becomes even worse. They could get erroneous findings.

### There is a need to make a bigger and better dataset with varied types of images and people of different ethnicities. We can come up with a few methods to tackle this problem. We can either create or arrange a platform for users to upload their images, or we can ask a few large image platforms, such as Meta, which deals in a variety of different types of user images, to provide us with datasets for research purposes, or we can merge the publicly or freely available datasets of deepfake and real photos into one.

### Second, there is another issue: deep fake analysis is not the ultimate solution to reducing the amount of synthetic information available on the internet. The final answer, we discovered, is verification of the material accessible on various websites on the internet. In the real world, this may be accomplished by large IT businesses banding together and including some type of identification into their platform while publishing material.

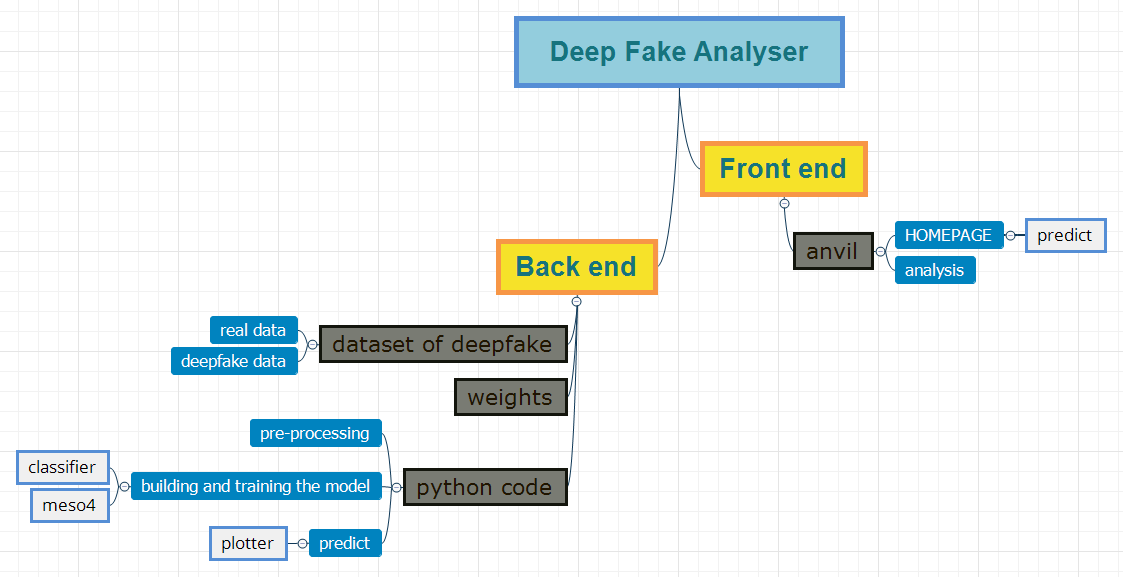
# RESEARCH OBJECTIVES

The proposed study is to conduct research that will lead to the creation of a method for identifying deepfakes. The Deepfake Analyser, which is the proposed project, will be accomplished by separating it into the following goals:

1. Obtain a deep fake dataset from Kaggle and refine it with real-world samples. This aids in the development of a better feature selection system.
2. Develop a deep learning and neural network-based algorithm to achieve optimum accuracy.
3. We'll add another function to the algorithm that will present the analysis as a bar chart, line chart, or pie chart, giving the user a better understanding of how much the number of deep fakes has grown and how they differ based on their confidence scores.
4. Anvil, a free Python-based drag-and-drop web app builder, will be used to create the GUI.

# METHODOLOGY

The following methodology will be followed to achieve the objectives defined for the proposed research work:



1. System flow diagram of DeepFake Analyser

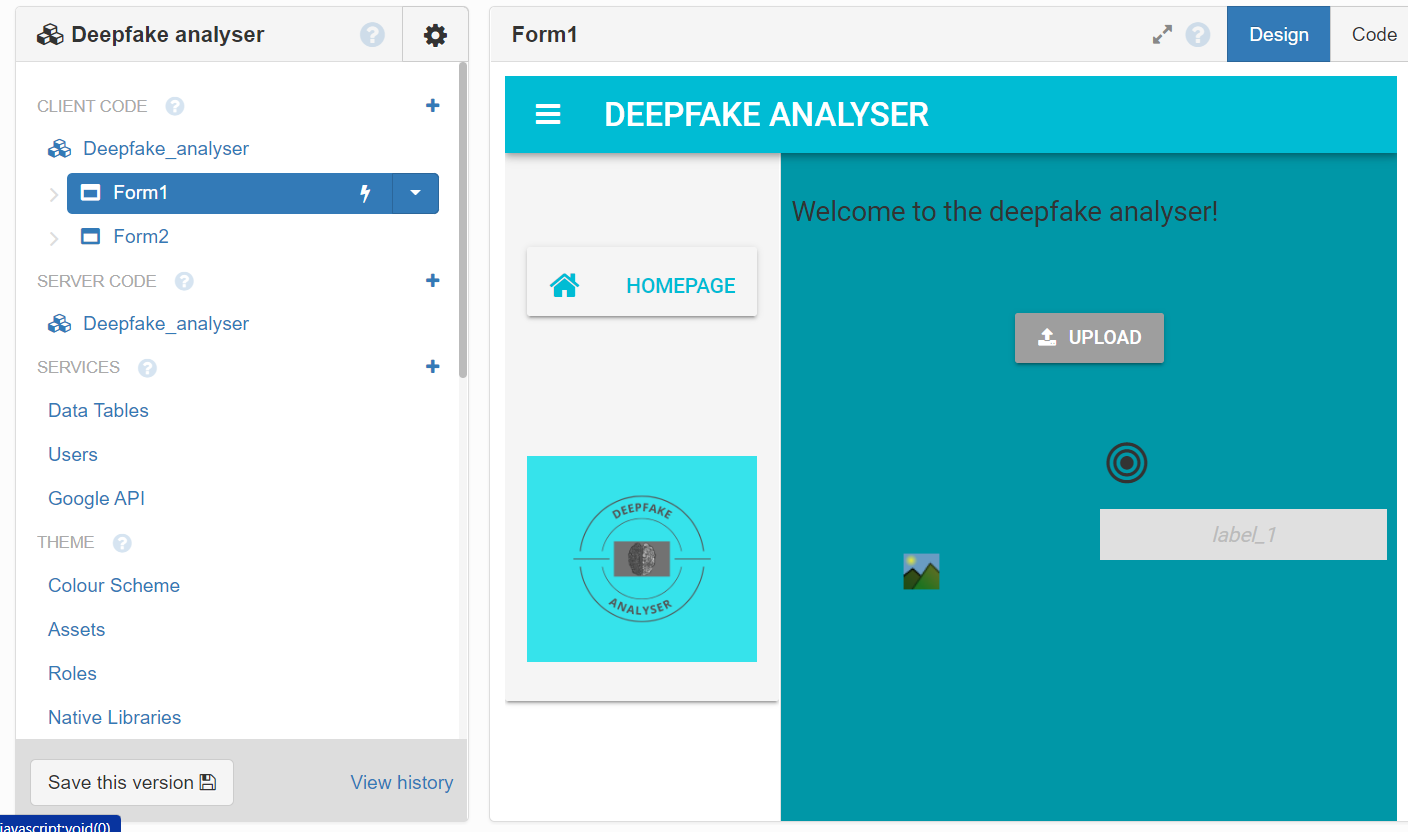
The system flowchart of our project, which is based on the Deep Fake Analyser, is illustrated in the diagram. It is split into two parts: the backend and the frontend. The datasets, which include both genuine and deep-fake photographs of individuals, make up the backend element. It also has weights since a CNN layer learns by modifying those weights to extract the features it needs to classify the item it's learning to classify. The Python code is then separated into preprocessing, model building, training, and prediction sections.

Anvil, a web application developer, is what we used for the front-end. The front-end section is divided into two sub-sections: Homepage and Analysis. It will feature an upload button on the site via which the dataset or a specific image can be submitted, and the confidence score will be predicted. In the analysis area, there is a pie chart, a bar chart, and a line chart that illustrate the representation of deep falsified media and which sectors they are aimed at, for example (finance, politics, revenge porn).

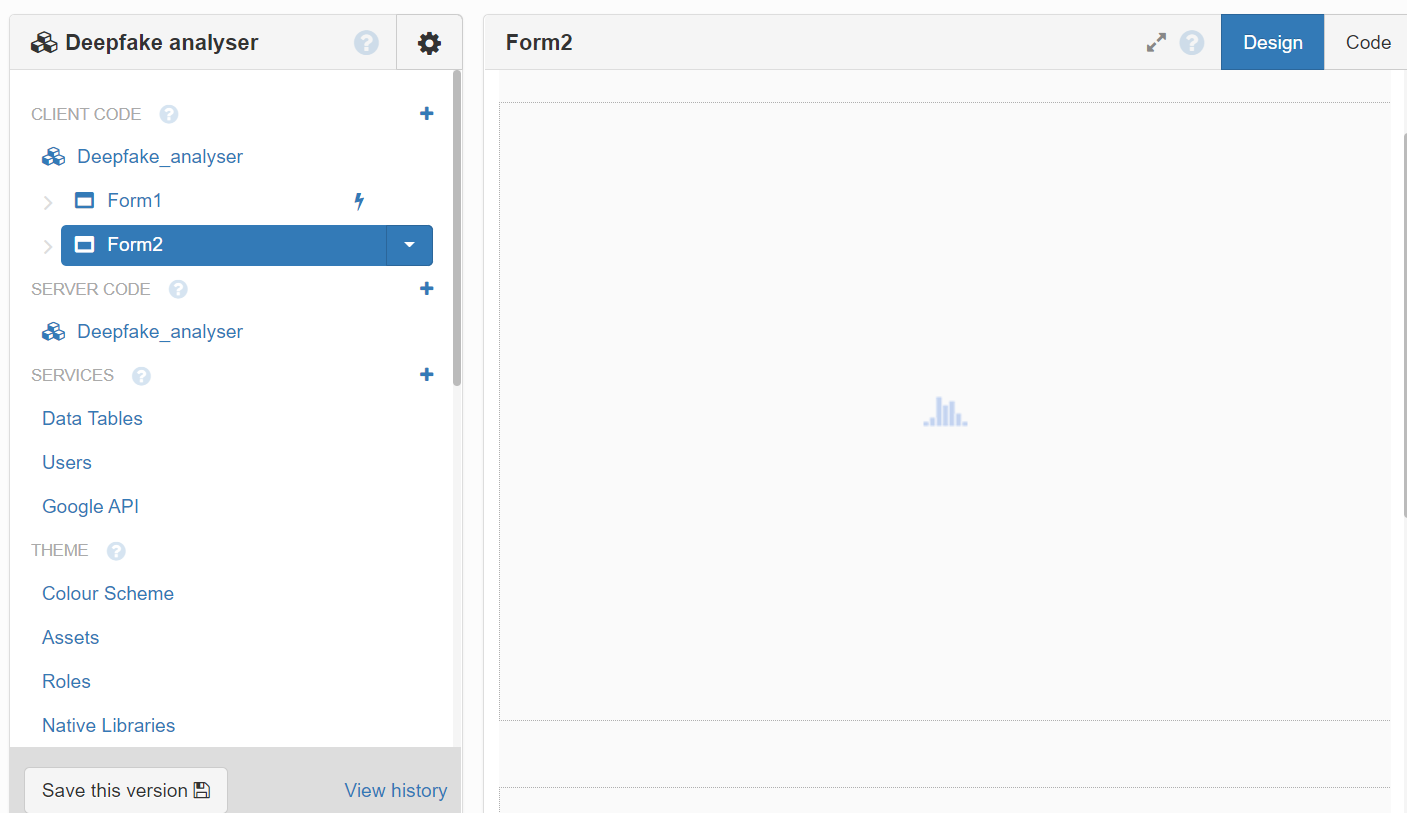
1. **TASK ACCOMPLISHED**

To add GUI to our main Model we took the help of an anvil which is used to develop web applications. We divided the GUI of the app into two forms. The first form shows the output of the model i.e. the confidence value, whether the image is real or deepfake and the second form shows the statistical output like bar graphs, pie charts etc. An example of this is given below:

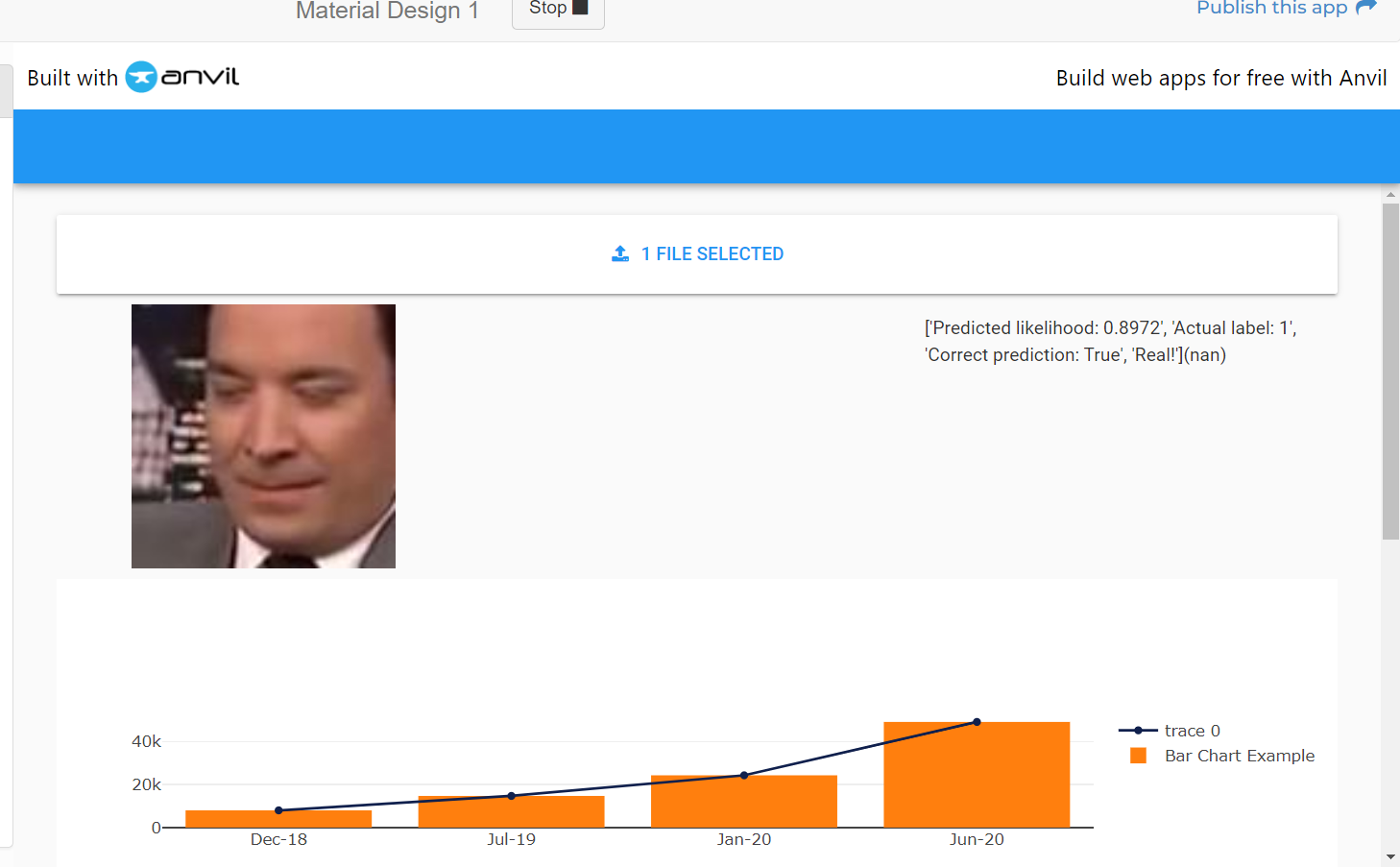
**FORM 1:**



**FORM2:**



To show the values of predicted output, bar graphs we added components like labels, images, plot in the anvil. In the code, we first imported all libraries required like an anvil.server, ploty.graph\_objects to achieve the desired outcome. To set properties of form we created init function and then int the loader function we first created a variable func and then called the server function created in the Jupyter notebook. Then we assigned component variable label to predicted output value fetched from the model. The same method we did for images and graphs. The output of work accomplished till date is given below:



1. **ONGOING AND UPCOMING TASKS**

**Ongoing Tasks:**

Currently we are working on the following tasks:

1. To add more Statistical analysis like pie charts, line charts, bar graphs.
2. To show the Predicted output in one form and the statistical analysis on the other form.

**Upcoming tasks:**

In the future we will try to do the following changes in our project:

1. We will try to shift to better platform like flutter for GUI.
2. Features like keeping the record of accuracy calculated by our model previously on the data provided by the user can be added.
3. **RESULTS AND DISCUSSION**

Deep learning is now making waves in the field of artificial intelligence, and image-based challenges have aided in this achievement. Because of recent developments in convolutional neural networks, we believe Ai will surpass humans in most traditional vision-based activities. There are true and significant ramifications for society, information, and the media that will have a big influence for years to come now that AI is capable of making convincing fake photographs and videos.

# REFERENCES

1. Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A. and Ortega-Garcia, J. (2020). Deepfakes and beyond: A Survey of face manipulation and fake detection. *Information Fusion*, [online] 64, pp.131–148. Available at: <https://arxiv.org/pdf/2001.00179.pdf>.
2. Dixon, M., Klabjan, D. and Bang, J.H. (2017). Classification-based financial markets prediction using deep neural networks. *Algorithmic Finance*, 6(3-4), pp.67–77.
3. Dailymail.co.uk. (2022). [online] Available at: https://i.dailymail.co.uk/1s/2019/06/12/00/14669252-0-image-a-2\_1560294887875.jpg [Accessed 5 Apr. 2022].
4. Palmer, E. (2022). *Donald Trump’ deepfake shows ex-president joining Russia’s version of YouTube*. [online] Newsweek. Available at: https://www.newsweek.com/trump-deepfake-russia-rutube-1687647 [Accessed 5 Apr. 2022].
5. Jadhav, A.H. (2022). *Deepfake detection using Deep Learning (ResNext and LSTM)*. [online] GitHub. Available at: https://github.com/abhijitjadhav1998/Deepfake\_detection\_using\_deep\_learning.
6. Yang, X., Li, Y. and Lyu, S. (n.d.). *EXPOSING DEEP FAKES USING INCONSISTENT HEAD POSES*. [online] Available at: https://arxiv.org/pdf/1811.00661v2.pdf [Accessed 7 Apr. 2022].
7. kaggle.com. (n.d.). *Deepfake Detection*. [online] Available at: https://www.kaggle.com/code/piyush357/deepfake-detection/notebook [Accessed 7 Apr. 2022].