A computer and world map

Description automatically generated

Cairo University

Faculty of Computers and Artificial intelligence

Department of Computer Sciences

Deepfake Detection

Supervised by

*Dr. Lamia Abo Zaid*

Implemented by

|  |  |
| --- | --- |
| *20200362* | *Omar Mahmoud Ali* |
| *20201014* | *Esraa Osama Mohamed* |
| *20201134* | *Fatma Mahmoud Ramadan* |
| *20201045* | *Basmala Magdy Mohamed* |
| *20201065* | *Rahma Mohamed Sayed* |

Graduation Project

Academic Year 2023-2024

Midyear Short Documentation

Project Short Documentation

**Abstract**

**Deepfake** technology is an emerging field that combines **artificial intelligence (AI)** and **machine learning** techniques to create highly realistic manipulated media content. The rapid advancement of **deepfake** technology poses a significant threat to the integrity of multimedia content on various platforms. In response to this challenge, this project aims to develop a web application to detect **deepfake**, this documentation introduces a **deep learning-based** method designed to effectively discern AI-generated fake videos from authentic ones. Our approach will integrate a **Res-Next Convolutional Neural Network (CNN)** for extracting frame-level features, complemented by a **Long Short-Term Memory (LSTM)** based **Recurrent Neural Network (RNN)** for capturing temporal dependencies. Ensuring the model's proficiency in identifying subtle alterations introduced during **deepfake** creation. The model will be trained and evaluated exclusively on the **Celeb-DF-v2** dataset. To enhance adaptability and learning, our web application incorporates user feedback, contributing to a more robust defense against the evolving challenges posed by AI-generated deep fakes. By doing so, we aim to create a safer digital environment for users, mitigating the potential impact of manipulated media content.

Table of Contents

**1 Introduction 5**

**2 Background5**

**3 Problem Definition 6**

3.1 Technologies and tools used**7**

3.1.1 Hardware Resources Required**7**

3.1.2 Software Resources **7**

**4 Related Work 7**

4.1 The existing similar implementations to the idea of our project **7**

4.2 The main differences between the current related work and our project **9**

4.2.1 Continual Lifelong Learning Approach**9**

**5 Project Specifications 10**

5.1 Structure Diagram **10**

5.2 Stakeholders **11**

5.3 Functional Requirements **11**

5.3.1 Functional requirement of the model **11**

5.3.2 Functional requirement of the Application **12**

5.4 Non-Functional Requirements **13**

**6 Work Plan 13**

**7 References 15**

List of Figures

[Figure 1 EWC…………………………………………………………………………………………………………………………………………10 Figure 2 accuracy and training efficiency comparison……………………………………………………………………………10](#_Toc158307500)

[Figure 3 training flow…………………………………………………………………………………………………………………………….10](#_Toc158307501)

[Figure 4 Prediction flow………………………………………………………………………………………………………………………...11](#_Toc158307502)

[Figure 5 stakeholders…………………………………………………………………………………………………………………………….11](#_Toc158307503)

[Figure 6 Gantt chart……………………………………………………………………………………………………………………………….14](#_Toc158307504)

List of Tables

[Table 1 Minimum server side hardware requirement 7](#_Toc158314664)

[Table 2 work plan 13](#_Toc158314665)

**1. Introduction**

**Deepfake** technology has rapidly emerged as a formidable challenge to the authenticity and integrity of multimedia content across various platforms. Leveraging sophisticated **artificial intelligence (AI)** and **machine learning** techniques, deepfake technology enables the creation of highly convincing manipulated media content, blurring the lines between reality and fabrication.

In response to this pressing issue, this documentation introduces a **deep learning-based** **approach** aimed at effectively discerning AI-generated fake videos from authentic ones. The proposed method integrates **advanced neural network architectures**, specifically a **Res-Next Convolutional Neural Network (CNN)** for extracting frame-level features and a **Long Short-Term Memory (LSTM)** based **Recurrent Neural Network (RNN)** for capturing temporal dependencies. By leveraging these cutting-edge techniques, our model demonstrates proficiency in identifying subtle alterations introduced during the creation of deepfake content (1).

Key to the effectiveness of our approach is its exclusive training and evaluation on the **Celeb-DF-v2** dataset (2), a benchmark dataset widely recognized in the field of deepfake detection. Additionally, our solution goes beyond mere detection by incorporating user feedback within a web application framework. This dynamic integration of Continual Lifelong Learning enables the model to adapt and improve over time, enhancing its robustness against evolving deepfake creation techniques.

Through the development of this solution, we aim to contribute to the creation of a safer digital environment, safeguarding users against the pernicious effects of manipulated media content. By bolstering the capabilities of deepfake detection technology, we seek to mitigate the potential impact of misinformation, fraud, and digital deception.

**2. Background**

**Deepfake** technology, which uses **neural network** tools such as **GANs (Generative Adversarial Network)** and **Autoencoders**, has become a prominent method for human image synthesis. This technology superimposes target images onto source videos using deep learning, producing deceptively realistic fake videos. The challenge is to detect fake videos generated by artificial intelligence, which are often indistinguishable to the human eye. This project will present a deep learning-based method designed to effectively distinguish fake videos from genuine ones.

The motivation for deepfake detection arises from the pressing need to counter the proliferation of manipulated media content fueled by **AI** advancements. These videos, while visually persuasive, pose significant risks such as spreading misinformation and eroding trust. By leveraging the limitations of deepfake creation tools, detecting subtle traces becomes imperative for safeguarding individuals and organizations from potential harm.

**Deepfake detection technology** has several beneficiaries across different sectors, making a positive impact on various fronts. For individuals, **deepfake detection technology** serves as a crucial safeguard, protecting against deceptive videos or images that could lead to misinformation or fraud. News and media organizations can rely on this technology to maintain their credibility by ensuring the authenticity of content before publishing it and preventing the unintended spread of false information. In Digital Forensics deepfake detection can help Investigators find and analyze fake videos faster and more accurately, saving time and making their investigations more effective.

**Main Techniques used to detect deepfake:**

**Deepfake detection** relies on a variety of techniques drawn from **computer vision**, **machine learning**, and **forensic analysis**. These techniques may include:

1. **Facial and Body Movement Analysis:** Analyzing facial expressions, eye movements, lip synchronization, and other physiological cues to detect inconsistencies indicative of deepfake manipulation (3).
2. **Deep Learning Models:** Training deep neural networks to recognize patterns and artifacts specific to deepfake manipulation, often using large datasets of both authentic and manipulated media.
3. **Metadata and Source Analysis**: (4) Examining metadata, such as timestamps and geolocation data, and analyzing the digital footprint of media to assess its authenticity.

The main application of deepfake detection technology is to identify and mitigate the spread of manipulated media content across various platforms and communication channels. This includes social media platforms, news websites, video-sharing platforms, and messaging apps. Additionally, deepfake detection tools may be integrated into content moderation systems, media forensics tools to provide a defense against deepfake-based threats.

**3.** **Problem definition**

The primary concern is the potential for deepfakes to be exploited for malicious purposes, including but not limited to disinformation campaigns, reputation damage, identity theft, and the erosion of trust in digital media. As deepfakes become more sophisticated and accessible, it is imperative to identify and address the underlying issues to mitigate their negative impact, It becomes very important to detect these deepfake, We will take a step forward in detecting the deep fakes using **LSTM** based **artificial Neural network**. The project aim is to develop a **web application** that allows users to enter URL of video or uploading it and classify it as either fake or real, develop browser plugin (extension) for automatic deepfake detection. This expansion would enable seamless integration into popular applications like Facebook, providing users with an easy way to pre-detect deepfakes before sending them to others.

In our project, the main techniques revolve around leveraging advanced technologies. Computer vision takes the spotlight, playing a crucial role in processing videos and frames through **OpenCV**. To dive into the technicalities, our method capitalizes on the distinctive artifacts left by current deepfake creation tools. Using a **Res-Next Convolution Neural Network**, we extract frame-level features, effectively capturing these artifacts. This process is complemented by the implementation of a **Long Short-Term Memory (LSTM)** based **Recurrent Neural Network (RNN)**, ensuring our model's proficiency in identifying subtle alterations introduced during deepfake creation. To round it off, a **PyTorch** trained model acts as a classifier, determining whether the source video is a deepfake or pristine, tying together the intricate components of our deep learning approach (1).

**3.1 Technologies and tools used:**

After analysis we decided to use the **PyTorch** framework along with **python3** language

or programming. (1) **PyTorch** is chosen as it has good support for **CUDA** i.e. **Graphic**

**Processing Unit (GPU)** and it is customize-able. We will evaluate our model with a large

dataset which includes YouTube videos, Celebrity real videos and Celeb-synthesis.

The Confusion Matrix approach will be used to evaluate the accuracy of the trained model.

**3.1.1 Hardware Resources Required:**

In this project, a computer with sufficient processing power is needed. This project

requires too much processing power, due to the image and video batch processing.

**3.1.1.1 Client-side Requirements:**

  Browser: Any Compatible browser device.

**3.1.1.2 Server-side Requirements:**

|  |  |  |
| --- | --- | --- |
| No. | **Parameter** | **Minimum Requirement** |
| 1 | Intel Xeon E5 2637 | 3.5 GHz |
| 2 | RAM | 16GB |
| 3 | Hard Disk | 100 GB |
| 4 | Graphic card | NVIDIA GeForce GTX Titan (12 GB RAM) |

Table 1 Minimum server side hardware requirement

**3.1.2** **Software Resources:**

* Operating System: Windows 10
* Programming Language: Python 3.10.4
* Framework: PyTorch 2.0
* Libraries: OpenCV, Face-recognition

**4. Related work:**

In the field of deepfake detection, several existing implementations have been developed to address the challenge of distinguishing between authentic and manipulated media content. These implementations aim to leverage advanced artificial intelligence techniques and algorithms to detect the presence of deepfakes and mitigate the potential harm caused by the spread of misinformation and fraudulent media. In this section, we provide an overview of the deepfake detection related work, highlighting the key methodologies, approaches, and advancements that have been proposed by previous researchers and practitioners and the main difference between the current related work and our project.

**4.1** **The existing solutions for detecting deepfake videos:**

* **Edward J. Delp,** [**David Guera**](https://ieeexplore.ieee.org/author/37086017582)**.** (5)This paper focuses on the growing threat of deep fake videos, particularly in the realm of facial manipulations, amidst the surge in digitalization and the adoption of AI technologies. The study utilizes the **HOHA** dataset, consisting of 300 videos. The proposed solution introduces a groundbreaking deep learning model that synergizes **Convolutional Neural Networks (CNNs)** for feature extraction with **Long Short-Term Memory (LSTM) Recurrent Neural Networks** for classification. This innovative approach achieves an impressive accuracy of **92.49%,** demonstrating its efficacy in discerning manipulated videos from authentic ones and establishing itself as a robust defense against AI-based fake videos.

* **Hasam Khalid, Simon S. Woo**. (6) this paper focuses on Deepfake classification using a **One-Class Variational Autoencoder (OC-VAE)** and introduces two variants: **OC-FakeDect-1** and **OC-FakeDect-2**. Trained exclusively on real images from **FaceForensics++**, **OC-FakeDect** surpasses the baseline **OC-AE** in precision, recall, and F1 scores across diverse deepfake datasets. **OC-FakeDect-2** achieves the highest performance, showcasing its superiority. The **VAE-based approach** excels in learning probability distribution parameters, enhancing its capability to detect various types of fake images. The paper leverages **MTCNN** for face detection, **GradCAM** for feature visualization, and anomaly scoring for classification. Robust model training includes data augmentation and statistical thresholding. The results underscore **OC-FakeDect's** effectiveness in deepfake detection.
* **Darius Afchar** et al. (7) This paper focuses on deep learning techniques to detect manipulated content created through **Deepfake** and **Face2Face** techniques. The study uses two network architectures**, Meso-4** and **MesoInception-4**, to perform the analysis at the microscopic level. The **Meso-4** network starts with four layers of **convolution** and **sequential pooling**, complemented by a **dense network** featuring a single hidden layer that uses **ReLU** activation for efficient generalization. The **MesoInception-4** network includes a different version of the initial unit for its first two convolutional layers. **MesoInception-4** achieves a high accuracy of **98%** on **Deepfake** and **95%** on **Face2Face**. The datasets used consist of the self-collected Deepfake dataset and the FaceForensicsdataset for **Face2Face**. The results underscore the importance of focusing on the eyes and mouth regions in determining the content of the manipulated face.
* **Ruben Tolosana** et al. (1) the study extensively evaluated DeepFake detection accuracy across different datasets, including both 1st and 2nd generations of DeepFakes. In 1st generation datasets, **Xception**, **Capsule Network**, and **DSP-FWA** achieved near-perfect accuracy, particularly in **UADFV**, though **FaceForensics++** presented challenges with a higher Equal Error Rate. Transitioning to 2nd generation datasets, **Celeb-DF v2** demonstrated strong performance (**97.90% AUC**), and the Eyes region consistently yielded superior results. However, the **DFDC** Preview dataset posed challenges, resulting in lower AUC values (**88.85%**). Fusion techniques, both at the system and facial-region levels, further improved accuracy, achieving state-of-the-art results. The preference for a specific detection approach or dataset depends on the application's requirements and the desired trade-off between accuracy and robustness. The algorithms used include **Xception**, **Capsule Network**, and **DSP-FWA**, each tailored for individual facial regions and datasets, showcasing their effectiveness in detecting evolving Deepfake techniques.

**4.2** **The main differences between the current related work and our proposed project:**

**4.2.1** **Continual Lifelong Learning Approach**

In our **deep fake detection projec**t, we implement a Continual Lifelong Learning approach to enhance the model's accuracy over time by incorporating user feedback. Continual Lifelong Learning ensures that the model remains up-to-date and adaptable to evolving scenarios (8).

**4.2.1.1 Challenges of Continual Lifelong Learning**

While Continual Lifelong Learning offers the advantage of ongoing model improvement, it introduces the risk of **catastrophic forgetting**. This occurs when training a model with new samples or classes leads it to gradually forget previously learned information, as the model's weights and biases adjust primarily to accommodate the new data (8) (9).

**4.2.1.2 Solutions to Catastrophic Forgetting**

**4.2.1.2.1 Regularization Techniques:**

* **Regularizing Weights:** This approach involves imposing constraints on the weights during training to prevent significant changes. However, due to the neural network's complexity, altering weights excessively may lead to erroneous outputs (8).
* **Regularizing Outputs:** To mitigate catastrophic forgetting, this method requires the model to retain knowledge of old examples while being trained on new data. However, it demands substantial computational resources and time (8).

**4.2.1.2.2 Progressive Neural Network:**

* In a Progressive Neural Network (10), each time a new dataset is trained, new neurons, output layers, and connections are added without modifying existing weights. While this approach accommodates expanding datasets, it incurs high computational costs and time overhead.

**4.2.1.2.3 Elastic Weight Consolidation (EWC):**

**EWC** stands out as the most effective solution to mitigate **catastrophic forgetting.** It involves identifying crucial weights from previous datasets and penalizing updates to these weights during training on new data. This mechanism allows the model to retain important knowledge from past tasks while adapting to new information.

A diagram of a task

Description automatically generatedA graph of training time

Description automatically generated

Figure 1 EWC Figure 2 accuracy and training efficiency comparison

**Conclusion:**

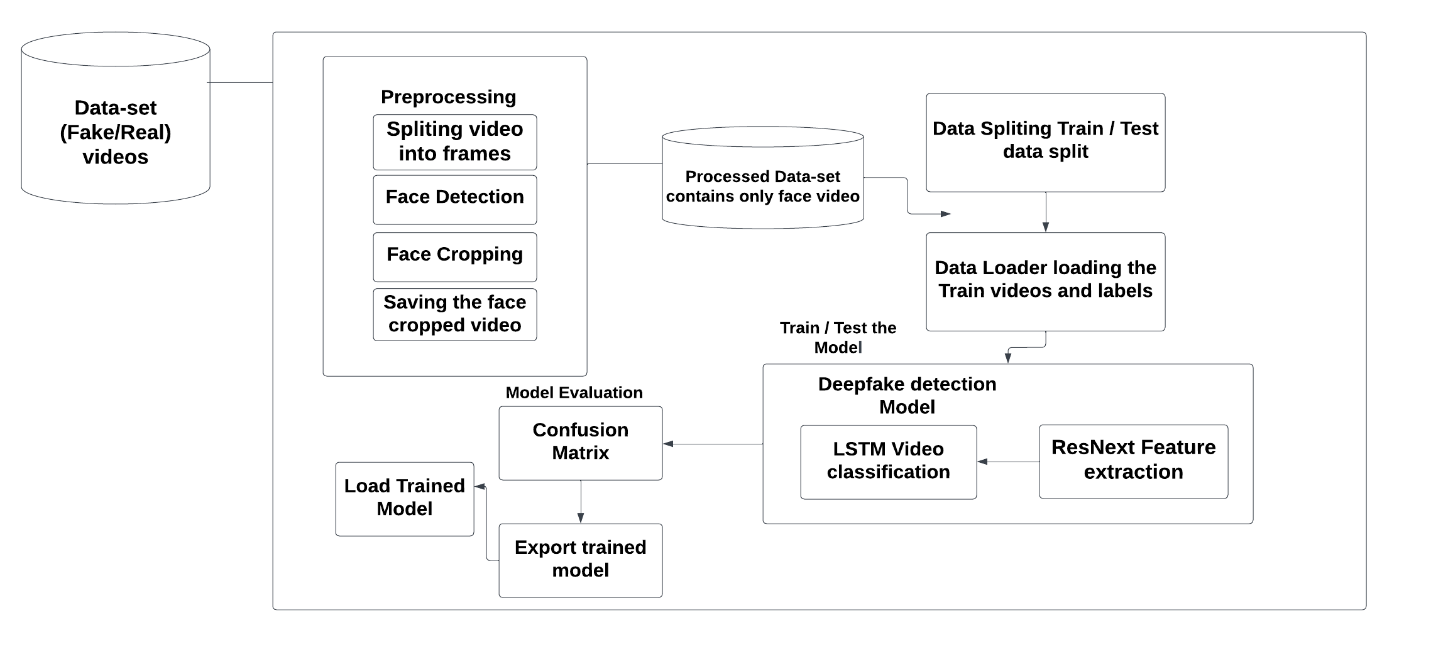
Among the various methods for addressing **catastrophic forgetting** in Continual Lifelong Learning, **Elastic Weight Consolidation (EWC)** emerges as the optimal solution. Its ability to preserve important information from previous tasks while accommodating new data makes it the most suitable approach for ensuring the ongoing learning and adaptation of our deep fake detection model.

**Incorporating EWC into Our Model:**

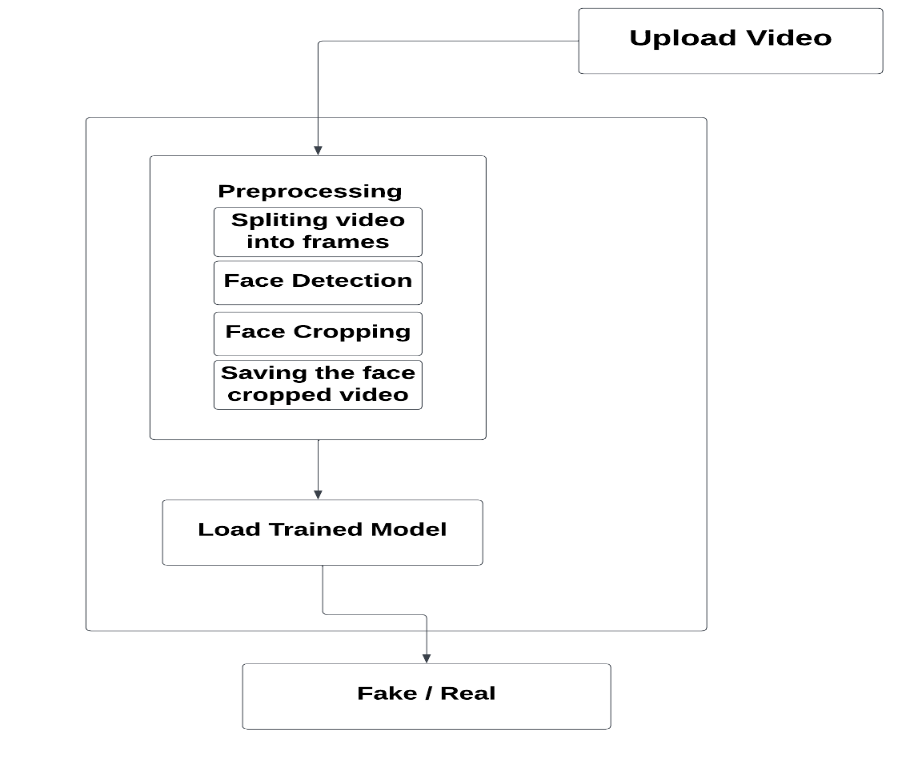
To achieve Continual Lifelong Learning without succumbing to **catastrophic forgetting,** we will integrate **Elastic Weight Consolidation (EWC)** into our deep fake detection model. By implementing this approach, we aim to continuously improve the model's performance while maintaining its ability to adapt to new challenges and evolving scenarios.

**5.** **Project Specifications:**

**5.1 Structure Diagram:**

**5.1.1 Training flow: ***Figure 3 training flow*

**5.1.2 Prediction flow:**

Figure 4 Prediction flow

**5.2 Stakeholders:**

Our project involves a multitude of stakeholders with various interests and roles in its success. The upcoming figure will illustrate the diverse range of stakeholders involved in our project.

A diagram of a system

Description automatically generated

Figure 5 stakeholders

**5.3 Functional Requirements:**

We divide functional requirement into two categories functional requirement of the model, functional requirement of the web application that will use the model.

**5.3.1 Functional requirement of the model:**

* User of the application will be able detect the whether the uploaded video is fake or real, along with the model confidence of the prediction, Users can determine the authenticity of uploaded videos and assess the model's prediction confidence. They can also provide feedback to facilitate ongoing learning and ensure the model remains up to date with evolving deepfake techniques and variations.

In order to achieve this requirement,

* we have identified set of parameters:

1. Blinking of eyes
2. Teeth enchantment
3. Bigger distance for eyes
4. Moustaches
5. Double edges, eyes, ears, nose
6. Iris segmentation
7. Wrinkles on face
8. Inconsistent head pose
9. Face angle
10. Skin tone
11. Facial Expressions
12. Lighting
13. Different Pose
14. Double chins
15. Hairstyle
16. Higher cheekbones

* We will train the model on **Celeb-Df-v2** dataset.

The structure of the dataset is:

* + Celeb-real consists of 590 Celebrity real videos.
  + YouTube-real consists of 300 Additional real videos.
  + Celeb-synthesis consists of 5639 synthesized videos made from   celeb real videos.

**5.3.2** **Functional requirement of the Application:**

In case of use the web application by entering URL of video

1. Get the URL: Prompt the user to input the URL of the video.
2. Download the Video:

* Use **youtube\_dl** or similar library to download the video from the provided URL.
* Save the downloaded video frames to a specified directory.

Then in case of uploading video or entering video’s URL

Upload Video Frames to Model:

* Load your pre-trained **PyTorch** model designed to detect fake videos.
* Preprocess each video frame:
  + Resize the frame to match the input size expected by the model (e.g., 224x224 pixels).
  + Convert the frame to a **PyTorch tensor**.
  + Normalize the pixel values of the frame according to the normalization parameters used during training.
* Read each video frame and pass it through the model:
  + Convert the frame to a format compatible with **PyTorch** (e.g., PIL Image).
  + Apply the preprocessing transformations to the frame.
  + Pass the preprocessed frame through the model.
* Interpret the model's output:
  + If the prediction score is above a certain threshold (e.g., 0.5), classify the video as fake.
  + If the prediction score is below the threshold, classify the video as authentic.

Output the classification result to the user.

**5.4** **Non-Functional Requirements:**

* **Prediction**: User will have the ability to view the targeted video while simultaneously observing the output on the faces within the video. This output will indicate whether the faces are classified as real or fake by the model. Additionally, the level of confidence associated with each classification will be displayed, providing the user with an indication of how certain the model is in its determination. This visual feedback allows the user to assess the model's performance and make informed judgments about the authenticity of the video.
* **Easy and User-friendly User-Interface**: Users seem to prefer a more simplified process of Deep Fake video detection. Hence, a straightforward and user-friendly interface is implemented, The UI contains an input field to enter video URL or uploading it for processing. It reduces the complications and at the same time enriches the user experience.
* **Cross-platform compatibility**: with an ever-increasing target market, accessibility should be your main priority. By enabling a cross-platform compatibility feature, you can increase your reach to across different platforms. Being a server-side application, it will run on any device that has a web browser installed in it, This approach eliminates the need for users to download and install platform-specific software, reducing barriers to entry and making the application more user-friendly

**6. Work plan:**

|  |  |  |
| --- | --- | --- |
| **Task** | **Task Title** | **Description** |
| 1 | Literature Review | Gather enough information about the problem through scientific papers |
| 2 | Gather different datasets | Choose the dataset that achieved the best accuracy which is Celeb-DF-v2 dataset |
| 3 | Dataset Preparation | preprocess Celeb-DF-v2 dataset |
| 4 | Develop Deep fake Detection Model | Implement Res-Next CNN and LSTM-based RNN |
| 5 | Web Application Development | Create a user-friendly website for uploading videos |
| 6 | User Feedback Integration | Implement a system for users to provide feedback |
| 7 | Model Training and Evaluation | Train our model helping it learn to distinguish real from manipulated videos. |
| 8 | Continual Lifelong Learning Integration | Implement EWC to enable ongoing model improvement based on user feedback |
| 9 | Documentation | Prepare detailed documentation for the project |
| 10 | Final System Testing | Testing of the complete system, including the web application and deepfake detection model. |

Table 2 work plan

**Gantt Chart:**

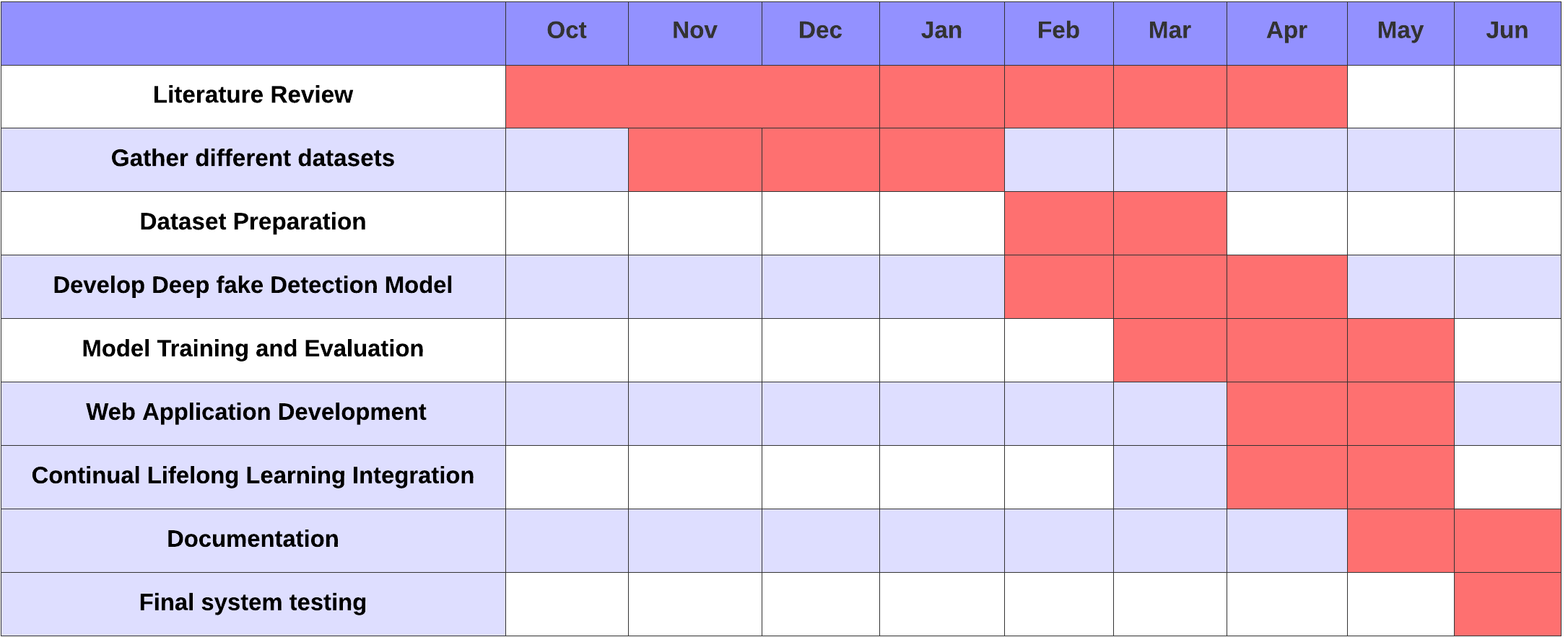


Figure 6 Gantt chart

# References

(1). Retrieved from https://www.bioinf.jku.at/publications/older/2604.pdf

(2). Retrieved from https://paperswithcode.com/paper/celeb-df-a-new-dataset-for-deepfake-forensics

(3). Retrieved from https://ieeexplore.ieee.org/document/9072088

(4). Retrieved from https://arxiv.org/pdf/2001.06564.pdf

(4). Retrieved from https://ieeexplore.ieee.org/document/8639163

(6). Retrieved from https://ieeexplore.ieee.org/document/9150876

(7). Retrieved from Retrieved from https://www.researchgate.net/publication/330791801\_MesoNet\_a\_Compact\_Facial\_Video\_Forgery\_Detection\_Network?\_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InNpZ251cCIsInBhZ2UiOiJwdWJsaWNhdGlvbiIsInByZXZpb3VzUGFnZSI6InNlYXJjaCJ9fQ

(8). Retrieved from https://www.sciencedirect.com/science/article/pii/S0893608019300231

(9). Retrieved from https://arxiv.org/abs/1612.00796

(10). Retrieved from https://arxiv.org/pdf/1606.04671