**Capstone Project Concept Note and Implementation Plan**

**Project Title: Deepfake Video Detection**

**Team Member**

**Farmanuddin Farman**

**Concept Note**

**1. Project Overview**

* Deepfake videos, generated using artificial intelligence and deep learning techniques, pose significant threats to various sectors, including cybersecurity, misinformation prevention, and content integrity. The capstone project aims to create a robust deepfake detection model leveraging the capabilities of CNNs. By training the model on Kaggle DFDC dataset containing real and manipulated videos, the model will learn to identify anomalies and inconsistencies in visual data, distinguishing between real and manipulated content.

From an international security and stability perspective, ready access to increasingly sophisticated deepfake technologies could lower the barriers to weaponizing information and delivering tailored harm or disruption in society as well as in the political and military spheres. [1]

By focusing on deepfake detection using CNNs, the capstone project directly contributes to the broader goals of building a safer and more secure digital environment, aligning with the principles of peace, justice, innovation, education, and global partnerships as outlined in the United Nations' Sustainable Development Goals.

* The project addresses the escalating issue of deepfake videos. Deepfakes can have severe consequences, including the spread of misinformation, manipulation of public opinion, and potential threats to individuals' reputations and security.

The impact of the solution includes:

* Mitigating Disinformation: By accurately identifying and flagging deepfake videos, the solution contributes to the prevention of disinformation campaigns and fake information, fostering a more informed public.
* Preserving Trust and Integrity: Protecting individuals, organizations, and public figures from the harmful effects of deepfakes.

**2. Objectives**

* The specific objectives of the deepfake video detection project using Convolutional Neural Networks (CNNs) are outlined as follows:  
  Dataset Preparation, Preprocessing and Feature Extraction, Model Selection and Architecture Design (InceptionResNetV2 and fine-tuning), Training the Detection Model, Validation and Evaluation and Implementation of Data Augmentation.
* To achieve a high level of accuracy in detecting deepfake videos using the developed Convolutional Neural Network (CNN) model and also to ensure the developed model is robust to various deepfake generation techniques, including different AI-based approaches and evolving manipulation methods.

**3. Background**

* Powered by advances in artificial intelligence and deep learning, deepfake videos represent a significant technological challenge and growing concern for various sectors. These synthetic media use algorithms that it is incredibly sophisticated, often based on existing Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs), Manipulate or change the content of videos, create realistic yet fake scenes.
* **Why a Machine Learning Approach is Beneficial or Necessary:**

Deepfake generation techniques are evolving rapidly. A machine learning approach, particularly using Convolutional Neural Networks (CNNs), allows the model to adapt and learn from new patterns and features in deepfake videos.

CNNs excel at feature extraction and pattern recognition in image and video data.

Machine learning models, once trained, can process large volumes of data efficiently.

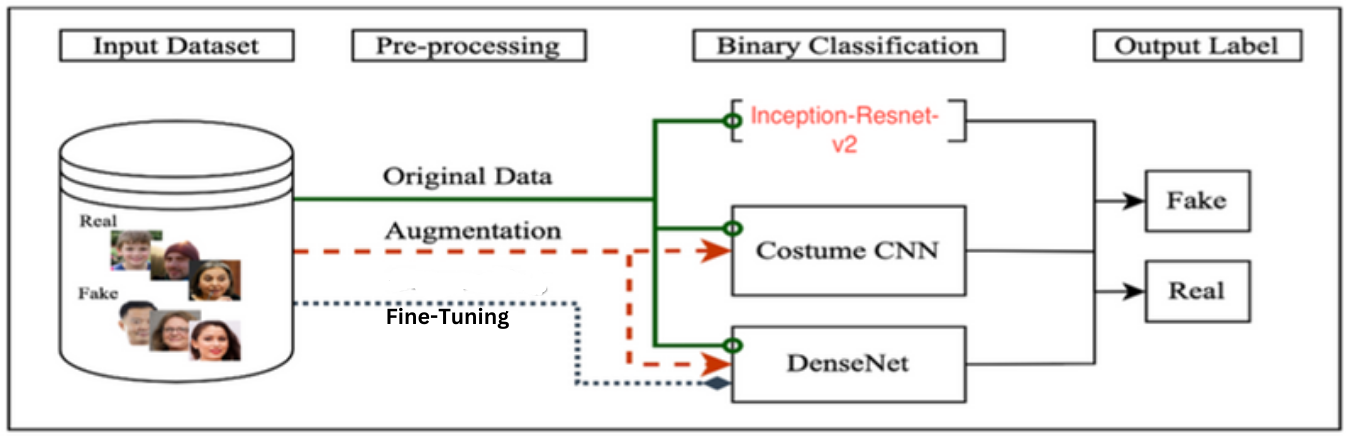
Many machines learning-based solutions, especially those leveraging CNNs, can perform real-time detection

**4. Methodology**

* Deep learning approach has been used in this research for detection of deepfakes from videos, more specifically Convolutional neural networks (CNNs).
* **Transfer Learning:** Leveraging pre-trained CNN models such as InceptionResNetV2 on large video dataset.
* **Fine-tuning:** Modifying and fine-tuning the pre-trained CNN model to adapt it to the specifics of deepfake detection.
* **Data Augmentation:** Applying image data augmentation techniques such as rotation, flipping, and scaling to artificially increase the diversity of the training dataset.
* **Global Average Pooling:** Using global average pooling layers to reduce the spatial dimensions of the data.
* **Activation Functions and Dense Layers:** Implementing activation functions like ReLU and adding dense layers for global pattern recognition.
* **Evaluation Metrics:** Employing evaluation metrics such as accuracy, precision, recall, and F1 score to assess the model's performance.

**5.** **Architecture Design Diagram**

* This high-level architecture illustrates the flow of the deepfake detection project, from data preparation to model evaluation.



* **Data Preparation:** Collect and preprocess a diverse dataset containing real and deepfake videos. Extract frames and perform face detection to focus on facial features. OpenCV for video processing and face detection.
* **Feature Extraction using CNN:** Pre-trained CNN model (InceptionResNetV2) for feature extraction. TensorFlow and Keras for building and fine-tuning the CNN model.
* **Data Augmentation:** Data augmentation techniques to increase dataset diversity and improve model generalization. Data augmentation functions provided by Keras.
* **Activation Functions and Dense Layers:** Implementing activation functions like ReLU and adding dense layers for global pattern recognition.
* **Evaluation Metrics:** Employing evaluation metrics such as accuracy, precision, recall, and F1 score to assess the model's performance.

**6. Data Sources**

* The dataset used in this report is a famous Kaggle competition dataset, known as deepfake detection challenge (DFDC). The dataset was used for the purpose to detect deepfakes in videos in 2020, and numerous benchmarks were attained as a result of application of various architectures. Although this competition is closed as of today, some data is still available for researchers to apply deepfake detection algorithms.

The dataset contains:

* A folder named “train sample videos”, comprising 401 training videos
* A folder named “test videos”, comprising of 400 testing videos
* The videos belong to 2 classes, i.e., real & fake
* A separate JSON file is in both folders also provided explicitly, which can be used to distinguish real & fake videos while extraction of frames [2].

For distinguishing real video frames from fake frames, appropriate approaches are applied after extracting the relevant information from the JSON file.

**7. Literature Review**

* In recent research, the model was applied to the DFDC dataset containing 60 different real and fake clips. This study process goes through three stages. In the first stage, each video is converted into frames and the face in each frame is detected by pre-processing and then sliced ​​with the Haar-Cascade function; ResNet-50 is used in the feature extraction phase. As a feature extraction model, in the last step, the CNN classifier is used to determine whether the image is fake or real. In this method detection accuracy was achieved at 98% [3].
* my project builds upon this work by incorporating data augmentation technique. The introduction of data augmentation enhances the model's ability to generalize by exposing it to a broader range of variations in facial expressions, lighting conditions, and other factors. This extension aims to further improve the model's robustness and adaptability to diverse deepfake scenarios, contributing to the overall effectiveness of deepfake video detection systems.

**Implementation Plan**

**1. Technology Stack**

* For applying the proposed architecture on DFDC dataset, a specific flow is followed, typically involved in a deep learning lifecycle for various problems such as image classification, segmentation, etc. This is done in multiple steps, i.e., from loading the dataset into memory, till the prediction on extracted frames from dataset and making predictions.
* Loading Dataset from Kaggle into Google Colab
* Extracting Sequenced Frames from Videos
* Placing Frames in a Particular Folder in Google Drive
* Applying Image Data Augmentation to Frames
* Initializing Deep Learning Model (CNN)

Making Layers of CNN (InceptionResNetV2)

Fine Tuning CNN

Making the Final Layer to Complete CNN Architecture

* Fitting the Model
* Calculating Accuracy & Graphs

Loss Graph

Accuracy Graph

* Evaluation and Testing

**2. Timeline**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Task | **Day 1** | **Day 2** | **Day 3** | **Day 4** | **Day 5** |
| Data Collection and Preprocessing |  |  |  |  |  |
| Video-to-frame conversion/Data augmentation |  |  |  |  |  |
| Model Development/ Initializing CNN model |  |  |  |  |  |
| Fine-tuning CNN |  |  |  |  |  |
| Training and Evaluation |  |  |  |  |  |
| Deployment |  |  |  |  |  |

**3. Milestones**

* **Key Milestones in Project Development:  
   1. Dataset Preparation:** Download the DFDC dataset from Kaggle using API. And Complete video-to-frame conversion. Also apply data augmentation to enhance dataset diversity.

**2.** **Fine-Tuning and Layer Construction:** Fine-tuning the CNN model on the augmented dataset. Construct additional layers, including global average pooling and dense layers, to complete the model architecture.

**3. Accuracy:** Above 80%.

**4. Challenges and Mitigations**

* Inconsistent or noisy data in the DFDC dataset may impact model performance. Variations in lighting, facial expressions, and video quality can pose challenges.  
  -Implementing thorough data exploration and preprocessing to identify and address data inconsistencies.
* Achieving optimal model performance is a common challenge in deep learning projects. Overfitting or underfitting may occur.  
  -Utilizing techniques like transfer learning with a pre-trained CNN to leverage knowledge from large datasets.
* Limited computational resources, especially when using Google Colab, can impact the speed and scale of model training.  
  -Optimizing code for efficiency and resource utilization.

**5. Ethical Considerations**

* The source videos in DeeperForensics-1.0 are recorded in a controlled, studio setting, while the DFDC Dataset contains videos captured in diverse real-world conditions. By addressing these ethical considerations, the project aims to ensure responsible development, deployment, and use of the deepfake detection model.

**6. References**

1. A. Anand and B. Bianco, The 2021 Innovations Dialogue Conference Report: Deepfakes, Trust and International Security, Geneva, Switzerland: UNIDIR, 2021.
2. <https://www.kaggle.com/competitions/deepfake-detection-challenge/data>
3. S. R. Adnan and H. A. Abdulbaqi, "Deepfake Video Detection Based on Convolutional Neural Networks," 2022 International Conference on Data Science and Intelligent Computing (ICDSIC), Karbala, Iraq, 2022, pp. 65-69, doi: 10.1109/ICDSIC56987.2022.10075830.