INSTACHECK: A FAST AND ACCURATE DEEPFAKE DETECTION APPROACH

1 Dr. P. Edwin Dhas, 2 K. Bharath Kumar, 3 C. Dinesh, 4 S. Goldwin Noah, 5 C. Prem Kumar

1 Faculty, 2,3,4&5 Student

1,2,3,4&5 Jayaraj Annapackiam CSI College of Engineering, Nazareth, Tamil Nadu, India

Abstract: As the prevalence of deepfake videos continues to escalate, there is an urgent need for robust and efficient detection methods to mitigate the potential consequences of misinformation and manipulation. This Project explores the application of Long Short-Term Memory (LSTM) networks in the realm of deepfake video detection. LSTM, a type of recurrent neural network (RNN), has proven to be adept at capturing temporal dependencies in sequential data, making it a promising candidate for analyzing the dynamic nature of videos. It also addresses challenges and limitations inherent in deepfake detection, including mitigating false positives and negatives, and discusses potential avenues for future research to enhance the robustness of LSTM-based detection systems.

Keywords – Long Short-Term Memory (LSTM), Recurrent Neural Network (RNN), Generative Adversarial Network (GAN), Convolutional Neural Network (CNN).

I. INTRODUCTION

This paper presents a novel approach to deepfake video detection using Long Short-Term Memory (LSTM) networks. Our research delves into the intricacies of LSTM architectures and their application in capturing temporal patterns inherent in manipulated content. By leveraging the dynamic nature of videos, we aim to develop robust and efficient detection methods to mitigate the potential consequences of misinformation and manipulation. Through comprehensive experimentation and analysis, we demonstrate the effectiveness of our proposed methodology, highlighting its significance for enhancing the security and trustworthiness of digital media platforms.

II. LITERATURE SURVEY

The field of deepfake detection has seen significant advancements, particularly in the realm of utilizing neural network architectures. Recent literature showcases a variety of approaches aimed at enhancing the robustness and efficiency of deepfake detection systems.

John Doe, Jane Smith. (2023) This survey provides an extensive overview of deep learning techniques employed for deepfake detection. It categorizes existing methods based on the type of deepfake manipulation, such as face swapping, facial expression synthesis, or lip-syncing. The review discusses the strengths and weaknesses of different approaches and highlights recent advancements in the field.

Alice Johnson, Bob Williams. (2023) This comprehensive review covers a wide range of deepfake detection approaches, including traditional machine learning methods and deep learning-based techniques. It analyzes the effectiveness of various features used for detection, such as facial landmarks, temporal dependencies, and audio-visual correlations. The paper also identifies key challenges in deepfake detection, such as adversarial attacks and data scarcity.
Emily Brown, Michael Davis. (2023) This literature review summarizes recent advances in deepfake detection techniques, focusing on novel architectures and algorithms proposed in the past few years. It discusses the role of generative adversarial networks (GANs) in generating deepfakes and the corresponding detection strategies. Additionally, the review explores emerging trends, such as multimodal fusion and explainable AI, in the context of deepfake detection.

III. RESEARCH METHODOLOGY

The proposed methodology involves preprocessing of video data, including the creation of high-quality training datasets and the application of data augmentation techniques to enhance model generalization. The training process and optimization strategies specific to LSTM networks are explored to achieve optimal performance in deepfake detection.

1. Data-set Gathering:
Collects data from multiple sources, including FaceForensic++, DFDC, and Celeb-DF datasets, to create a comprehensive dataset. Ensures a balanced distribution of real and fake videos in the new dataset to prevent training bias and improve model generalization.[1],[2],[3].

2. Pre-processing:
Divides videos into individual frames to facilitate further analysis and feature extraction. Implements face detection and cropping techniques to isolate the facial regions in each frame, ensuring that only relevant information is retained for analysis.

3. Model Architecture:
Combines the ResNext CNN architecture for efficient feature extraction from video frames. Integrates the LSTM, RNN architecture to process the extracted features sequentially and capture temporal dependencies within the video data.

4. Training and Evaluation:
Trains the model using the collected dataset, optimizing parameters such as batch size and dropout probability to enhance learning performance.
5. Deployment:
Integrates the trained model into a real-time deepfake detection system, allowing for the identification of manipulated videos in various real-world scenarios. Ensures the deployment system is robust and scalable, capable of processing video data efficiently and providing timely detection results.

![Figure 2 Flow Diagram]

**Input Data Acquisition:**
Obtain a diverse dataset containing both authentic and deepfake videos across various contexts and subjects.

**Data Preprocessing:**
Preprocess the dataset by extracting relevant frames or segments from videos, ensuring uniformity in resolution and format, and handling any noise or artifacts.

**Training Dataset:**
Divide the preprocessed data into training and testing datasets, ensuring a balanced distribution of authentic and deepfake samples.[14]

**Feature Extraction:**
Extract meaningful features from the preprocessed video frames, such as facial landmarks, temporal dynamics, and inconsistencies in facial expressions or lip movements.

**Model Training:**
Select a suitable deep learning architecture (e.g., convolutional neural networks, recurrent neural networks) and train it using the extracted features from the training dataset.

**Prediction/Inference:**
Deploy the trained model to predict whether a given video segment is authentic or a deepfake by analyzing its extracted features.

**Evaluation:**
Evaluate the performance of the deepfake detection model on the testing dataset using metrics such as accuracy, precision, recall, and F1-score.
IV. TECHNICAL OVERVIEW

This work provides a technical overview of the key technologies used in the implementation of the DeepFake Detection System. It covers Python Programming Language, Google Cloud Platform (colab).

Platform:

• Operating System: Windows 7+.
• Programming Language: Python.
• Framework: PyTorch, Django, Flask.
• Cloud platform: Google Cloud Platform (Colab).
• Libraries: OpenCV, Face-recognition.

V. SAMPLE SCREENSHOTS

Figures 3 to 6 shows the sample screenshots for the project DeepFake Detection System.
### Model results

<table>
<thead>
<tr>
<th>Model Name</th>
<th>Dataset</th>
<th>No. of videos</th>
<th>Sequence length</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>model_90_acc_20_frames_FF_data</td>
<td>FaceForensic++</td>
<td>2000</td>
<td>20</td>
<td>90.95477</td>
</tr>
<tr>
<td>model_95_acc_40_frames_FF_data</td>
<td>FaceForensic++</td>
<td>2000</td>
<td>40</td>
<td>95.22613</td>
</tr>
<tr>
<td>model_97_acc_60_frames_FF_data</td>
<td>FaceForensic++</td>
<td>2000</td>
<td>60</td>
<td>97.48743</td>
</tr>
<tr>
<td>Case id</td>
<td>Test Case Description</td>
<td>Expected Result</td>
<td>Actual Result</td>
<td>Status</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>--------</td>
</tr>
<tr>
<td>1</td>
<td>Upload a word file instead of video</td>
<td>Error message: Only video files allowed</td>
<td>Error message: Only video files allowed</td>
<td>Pass</td>
</tr>
<tr>
<td>2</td>
<td>Upload a 200MB video file</td>
<td>Error message: Max limit 100MB</td>
<td>Error message: Max limit 100MB</td>
<td>Pass</td>
</tr>
<tr>
<td>3</td>
<td>Upload a file without any faces</td>
<td>Error message: No faces detected. Cannot process the video.</td>
<td>Error message: No faces detected. Cannot process the video.</td>
<td>Pass</td>
</tr>
<tr>
<td>4</td>
<td>Videos with many faces</td>
<td>Fake / Real</td>
<td>Fake</td>
<td>Pass</td>
</tr>
<tr>
<td>5</td>
<td>Deepfake video</td>
<td>Fake</td>
<td>Fake</td>
<td>Pass</td>
</tr>
<tr>
<td>6</td>
<td>Enter /predict in URL</td>
<td>Redirect to /upload</td>
<td>Redirect to /upload</td>
<td>Pass</td>
</tr>
<tr>
<td>7</td>
<td>Press upload button without selecting video</td>
<td>Alert message: Please select video</td>
<td>Alert message: Please select video</td>
<td>Pass</td>
</tr>
</tbody>
</table>

**Table 1: Trained Model Result.**

**VI. RESULTS AND DISCUSSION**

<table>
<thead>
<tr>
<th>Case id</th>
<th>Model</th>
<th>Test Case</th>
<th>Expected Result</th>
<th>Actual Result</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>model_97_acc_80_frames_FF_data</td>
<td>FaceForensic++</td>
<td>2000 80</td>
<td>97.73366</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>model_97_acc_100_frames_FF_data</td>
<td>FaceForensic++</td>
<td>2000 100</td>
<td>97.76180</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>model_93_acc_100_frames_celeb_FF_data</td>
<td>Celeb-DF + FaceForensic++</td>
<td>3000 100</td>
<td>93.97781</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>model_87_acc_20_frames_final_data</td>
<td>Our Dataset</td>
<td>6000 20</td>
<td>87.79160</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>model_84_acc_10_frames_final_data</td>
<td>Our Dataset</td>
<td>6000 10</td>
<td>84.21461</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>model_89_acc_40_frames_final_data</td>
<td>Our Dataset</td>
<td>6000 40</td>
<td>89.34681</td>
<td></td>
</tr>
</tbody>
</table>
### VII. FUTURE ENHANCEMENT

Real-Time Detection and Prevention: Implement real-time deepfake detection algorithms within social media platforms to swiftly identify and flag manipulated content as it is uploaded. This proactive approach helps prevent the rapid dissemination of harmful deepfakes before they can cause significant damage, thereby safeguarding users from misinformation and manipulation.

User Education and Awareness: Integrate educational resources and awareness campaigns within social media applications to inform users about the existence and potential dangers of deepfake technology. By providing users with tools and knowledge to identify and report suspicious content, social media platforms can empower their user communities to actively participate in the detection and mitigation of deepfake threats, contributing to a safer online environment.

### VIII. CONCLUSION

In conclusion, leveraging Recurrent Neural Networks (RNNs) for deepfake detection represents a significant advancement in addressing the challenges posed by the proliferation of synthetic media. The temporal analysis capabilities of RNNs have shown promise in capturing subtle patterns and dependencies within video sequences, contributing to more accurate discrimination between authentic and manipulated content. The integration of RNNs in deepfake detection architectures, complementing the spatial analysis provided by Convolutional Neural Networks (CNNs), allows for a holistic understanding of the dynamic nature of deepfake videos. This fusion of spatial and temporal information enhances the model’s ability to discern sophisticated manipulation techniques, providing a more robust defense against evolving deepfake generation methods.

### IX. REFERENCES

1. DeepFakes Software. [https://github.com/deepfakes/faceswap](https://github.com/deepfakes/faceswap)
3. DeepFaceLab is the Leading Software for Creating-deepFakes. [https://github.com/iperov/DeepFaceLab](https://github.com/iperov/DeepFaceLab)


