**Objective**

The objective of the research paper is to introduce a novel deep-learning technique to identify deepfake videos by employing a hybrid method that utilizes the ResNeXt-Swish-BiLSTM architecture. The proposed method aims to enhance the accuracy and efficiency of deepfake detection by identifying artifacts in manipulated images and videos.

**Implication**

The implications of this research are significant for digital forensics and the broader societal context. By providing a robust and efficient deepfake detection model, the study helps in mitigating the spread of misleading and potentially harmful manipulated media. This is particularly relevant for social media platforms, news agencies, and law enforcement agencies that are grappling with the challenges posed by deepfake technologies. Furthermore, the research could pave the way for the development of more advanced algorithms and the integration of tamper-proof technologies such as blockchain for video authenticity verification.

**METHODOLOGY**

**4.1.1 Data Collection:** Images and videos for forgery detection are collected from

various sources. These sources include online repositories, datasets specifically

curated for deepfake detection, and manually generated forgeries.

**4.1.2 Sources:** DFDC (DeepFake Detection Challenge), FaceForensics++, Celeb-

DF, and other publicly available datasets.

**4.1.3 Preprocessing:** Collected data is preprocessed using several techniques to

prepare it for model training.

**4.1.3.1 Resizing:** Standardize image/video dimensions to ensure uniformity.

**4.1.3.2 Normalization:** Normalize pixel values to a consistent scale to aid in

model convergence.

**4.1.3.3 Augmentation:** Apply transformations such as rotation, flipping, and

cropping to increase the diversity of the training data and improve model

robustness.

**4.1.2 Feature Extraction:** Utilize advanced methods to extract meaningful features

from the preprocessed data.

**4.1.3 ResNext:** Employ a pre-trained ResNext model to extract deep spatial features

from each frame of the video or image.

**4.1.4 Swish Activation:** Use the Swish activation function to enhance the model’s

ability to learn complex patterns by providing smoother gradient flows.

**4.1.5 Temporal Modeling:** Incorporate temporal dependencies using Bidirectional

Long Short-Term Memory (BiLSTM) networks.

**4.1.5.1 BiLSTM:** Integrate a BiLSTM layer to capture sequential

dependencies in video frames, enabling the model to understand

temporal dynamics and improve detection accuracy.

**4.1.6 Model Training:** Train the combined ResNext-Swish-BiLSTM model on the

preprocessed and feature-extracted data.14

**4.1.7 Training Process:** Utilize supervised learning techniques, optimizing the

model using appropriate loss functions and optimizers to minimize errors.

**4.1.8 Hyperparameter Tuning:** Adjust hyperparameters such as learning rate,

batch size, and the number of BiLSTM layers to achieve optimal performance.

**4.1.9 Evaluation:** Evaluate the trained model on a separate test dataset to assess its

accuracy in detecting forgeries.

**4.1.9.1 Test Dataset:** Use a distinct set of images/videos not involved in the

training process to evaluate model performance.

**4.1.9.2 Metrics:** Employ standard evaluation metrics such as accuracy,

precision, recall, F1-score, and ROC-AUC to measure the model’s

effectiveness.

**4.1.10 Testing:** Test the model on real-world data to evaluate its generalizability and

robustness.

**4.1.10.1 Real-World Data:** Collect images and videos from various sources,

including social media, news articles, and user-generated content.

**4.1.10.2 Generalization:** Assess the model’s performance in accurately

detecting forgeries in diverse and previously unseen data, ensuring it can

handle various real-world scenarios effectively.

This outline provides a clear and structured approach to our proposed methodology,

detailing each step from data collection to evaluation and testing, ensuring

comprehensive coverage of the entire process.

**4.2 Implementation Strategy**

We are implementing our strategy by using a Web Application. The outline of the

following strategy is mentioned as follows:

**4.2.1 Video Submission by User on Web Portal:**

- Users can visit our web portal and submit videos in MP4 format. The video can be

either authentic or a deepfake.15

**4.2.2 Pre-processing of Video:**

Face Detection and Cropping: The video submitted by the user will undergo face

detection. If no face is detected, the process ends, and the user is notified. If a face

is detected, the frames containing the face are cropped and saved for further

processing.

Normalization and Augmentation: The detected face frames are normalized and

augmented to ensure uniformity and increase the diversity of the training data. This

step will be conducted in the backend of our application.

**4.2.3 Feeding Video Frames to the Model:**

**4.2.3.1 Data Loader:** The pre-processed face frames are loaded into the

model using a data loader, which ensures the correct labels and data structure

are maintained.

**4.2.3.2 Model Processing:** The frames are passed through the ResNeXt-

Swish CNN for feature extraction. The extracted features are then processed

through a BiLSTM network to capture temporal dependencies and enhance

classification accuracy.

**4.2.4 Classification of Video:**

**4.2.4.1 Output Label:** The model classifies the video as either authentic or

fake. Additionally, it provides model metrics such as accuracy, precision,

recall, and F1-score.

**4.2.4.2 Masked Frames:** The model also returns the masked frames

highlighting areas of potential tampering or changes, helping users to

visualize where the model detected possible forgeries.

Uploading Results to Cloud:

**Result Storage:** The web server uploads the classification results, including model

metrics and masked frames, to the user’s account on the cloud.

**User Access:** Users can access the results from their cloud account, providing a

convenient and secure way to review the analysis.16

**Fig 4.1** Social Rakshak Application workflow

The Fig 4.1 tell about the workflow Architecture of Social Rakshak our model and

the steps used are as follows:

**4.2.3 Social Rakshak Workflow (Web Application Flow):**

**Step 1:** User uploads an MP4 video to the web portal.

**Step 2:** The video is sent to a web server via the internet.

**Step 3:** The web server processes the video, checking for face detection. If no face

is detected, the user is notified.

**Step 4:** If a face is detected, the server pre-processes the video frames (face

cropping).

**Step 5:** The pre-processed frames are fed to the trained model for classification.

**Step 6:** The model returns the classification result (authentic or fake) and the masked

frames highlighting potential tampering.

**Step 7:** The result is sent back to the user's web portal via the internet.17

**Step 8 :** The user receives the result on their device.

**Fig 4.2** Training and Evaluation Workflow

The Fig 4.2 shows the Training and Evaluation Workflow of our model and the

steps used are as follows:

**4.2.4 Training and Evaluation workflow :**

**Step 1:** Data Collection from FF++, DFDC, and Celeb-DF datasets, totaling 6,000

videos.

**Step 2:** Pre-processing, including face detection and cropping, resulting in a dataset

of face-cropped frames.

**Step 3:** Dataset is split into training (4,200 videos) and testing (1,800 videos).

**Step 4:** Data Loader loads videos and labels into the model.

**Step 5:** ResNeXt-Swish CNN extracts features from the frames.18

**Step 6:** BiLSTM captures temporal dependencies and classifies the video.

**Step 7:** Confusion Matrix is used to evaluate the model’s performance.

**Step 8:** The trained model is exported and deployed for use in the web application.

**Final Step :** The accuracy of the model is assessed and reported.

This detailed strategy outlines the entire workflow from user video submission to

final result delivery, ensuring a clear understanding of the process and the

implementation of the proposed deepfake detection model

**Conclusion**

We investigated various models and activation functions to enhance the

accuracy and efficiency of deepfake video detection. By evaluating multiple deep

learning models against well-known datasets such as FF++, DFDC, and Celeb-DF,

we aimed to identify the optimal model for this task. Our results demonstrate the

superior performance of the ResNeXt-Swish-BiLSTM model, which achieved the

highest evaluation metrics with an accuracy (AC) of 0.992, precision (PR) of 0.99,

recall (RC) of 0.996, and F1-score of 0.993.

Additionally, we conducted a comprehensive analysis of different activation

functions, measuring their impact on accuracy, training time, and classification time.

Among the functions tested, Swish emerged as the most effective, yielding the

highest accuracy of 98.0% and maintaining competitive training and classification

times. This makes Swish particularly suitable for use in very deep networks, offering

a promising avenue for future implementations.

The results of our experiments underscore the importance of selecting both the right

model architecture and activation function to achieve optimal performance in

deepfake detection. Our final model, ResNeXt-Swish-BiLSTM, combined with the

Swish activation function, demonstrates a robust capability in identifying deepfakes,

marking a significant advancement in the field of video forensics with an accuracy

of 95.20% against combined dataset of DFDC, FF++ and Celeb - DF ,

95.66% against DFDC and FF++ dataset , 99.24% against only DFDC dataset and

97.82% against only FF++ Dataset.

This research provides a solid foundation for future work aimed at improving the

detection of synthetic media, thereby contributing to the broader efforts of

combating misinformation and ensuring digital content integrity.

Therefore, after thoroughly examining the ResNeXt-Swish-BiLSTM at the

statistical and digital media levels, we can conclude that our work in the field of

advanced digital investigation, such as criminal forensics.