FakeTweet Busters: A combination of BERT and Deepfake detection to resolve the spreading of fake AI generated news.

Department of Computer Science and Engineering
Netaji Subhas University of Technology
New Delhi

Name of the Student	**Roll Number**
Shubhangee Pal	2020UCO1574
Bhagyashree Das	2020UCO1615
Amrisha Das	2020UCO1687

Under supervision by:
Dr. Preeti Kaur

CERTIFICATE

Department of Computer Science and Engineering

This is to certify that the work embodied in project thesis titled, "**FakeTweet Busters:A combination of BERT and Deepfake detection to resolve the spreading of fake AI generated news**" by Shubhangee Pal (2020UCO1574), Bhagyashree Das (2020UCO1615) and Amrisha Das (2020UCO1687) is the bona fide work of the group submitted to Netaji Subhas University of Technology for consideration in 8th Semester B.Tech. Project Evaluation.

The original Research work was carried out by the team under my/our guidance and supervision in the academic year 2023-2024. This work has not been submitted for any other diploma or degree of any university. On the basis of a declaration made by the group, we recommend the project report for evaluation.

Dr. Preeti Kaur

(Associate Professor)

Department of Computer Science & Engineering

Netaji Subhas University of Technology

CANDIDATE(S) DECLARATION

Department of Computer Science and Engineering

I/We, Shubhangee Pal (2020UCO1574) , Bhagyashree Das (2020UCO1615) and Amrisha Das (2020UCO1687) of B.Tech. Department of Computer Science and Engineering hereby declare that the Project Thesis titled "**FakeTweet Busters: A combination of BERT and Deepfake detection to resolve the spreading of fake AI generated news**" which is submitted by us to the Department of Computer Science and Engineering, Netaji Subhas University of Technology, Delhi in partial fulfillment of the requirement for the award of the degree of the Bachelor of Technology, is original and not copied from source without proper citation. The manuscript has been subjected to plagiarism checks by "TurnItIn" software. This work has not previously formed the basis for the award of any degree.

Place:
Date:

Shubhangee Pal	Bhagyashree Das	Amrisha Das
(2020UCO1574)	(2020UCO1615)	(2020UCO1687)

CERTIFICATE OF DECLARATION

Department of Computer Science and Engineering

This is to certify that the project Thesis titled "**FakeTweet Busters:A combination of BERT and Deepfake detection to resolve the spreading of fake AI generated news**" which has been submitted by Shubhangee Pal (2020UCO1574), Bhagyashree Das (2020UCO1615) and Amrisha Das (2020UCO1687) of B.Tech. Department of Computer Science and Engineering hereby declare that the Project Thesis which is submitted by us to the Department of Computer Science and Engineering, Netaji Subhas University of Technology, Delhi in partial fulfillment of the requirement for the award of the degree of the Bachelor of Technology, is a record of the thesis work carried out by the students under my supervision and guidance. The content of this thesis, in full or in parts, has not been submitted for any other degree or diploma.

Place: New Delhi
Date: 06-05-2024

Dr. Preeti Kaur
(Associate Professor)
Department of Computer Science & Engineering
Netaji Subhas University of Technology

ACKNOWLEDGEMENT

We would like to express my gratitude and appreciation to all those who make it possible to complete this project. Special thanks to our project supervisor(s) Dr. Preeti Kaur whose help, stimulating suggestions and encouragement helped us in writing this report. We also sincerely thank our colleagues for the time spent proofreading and correcting our mistakes. We would also like to acknowledge with much appreciation the crucial role of the staff in Computer Science & Engineering, who gave us permission to use the lab and the systems and gave permission to use all necessary things related to the project.

Shubhangee Pal (2020UCO1574)

Bhagyashree Das (2020UCO1615)

Amrisha Das (2020UCO1687)

ABTRACT

The spread of misinformation through social media, particularly fake tweets, poses a significant challenge. This work proposes a novel multimodal approach for detecting fake tweets, combining textual and visual analysis. We leverage the strengths of BERT, a pre-trained transformer model, for comprehensive textual analysis, and VGG19, a convolutional neural network, for image feature extraction. Additionally, we incorporate deepfake detection to address manipulated visuals. For tweets containing both text and images, a multi-head self-attention mechanism effectively fuses the textual and visual features extracted by BERT and VGG19, respectively. This combined approach aims to achieve superior accuracy in identifying fake tweets compared to methods solely focused on text or image analysis.

CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CNN - Convolutional Neural Network
BERT - Bidirectional Encoder Representations from Transformers
GPT- General purpose technology
VGG - Visual Geometry Group
BOW - Bag of Words
CharCNN - Character Level CNN
LSTM - Long short-term memory
ML - Machine Learning
API - Application programming interface
GAN - Generative Adversarial Network
SVM – Support Vector Machine
MTCNN – Multi- Task Cascaded Convolution Neural Networks
CT-BERT – Covid Twitter Bidirectional Encoder Representations from Transformers
ASVpoof – Automatic Speaker Verification Spoofing (Dataset)
URL – Uniform Resource Locator

CHAPTER 1: INTRODUCTION AND LITERATURE REVIEW

1.1 Introduction

Social media platforms like Twitter have become integral parts of modern communication, offering users a platform to share thoughts, opinions, and information rapidly. However, with the rise of fake news and misinformation, distinguishing between genuine and deceptive content has become increasingly challenging. In this context, the detection of fake tweets, which may contain manipulated images or false information, has emerged as a crucial research area.

This project aims to develop a robust system for detecting fake tweets, leveraging a multimodal approach that combines textual and visual features. By analyzing both the content of the tweet text and the characteristics of accompanying images, our system seeks to provide a comprehensive solution for identifying deceptive content on Twitter. For the text analysis part, we utilize state-of-the-art language models such as BERT (Bidirectional Encoder Representations from Transformers), which are adept at capturing the nuances and context of natural language. Concurrently, for the analysis of images, Convolutional Neural Networks (CNNs) are employed to extract visual features and patterns from the images associated with tweets.

Moreover, deepfake detection techniques have been integrated into the analysis of images and audio data to enhance the system's effectiveness in identifying manipulated content. Deepfakes, which involve the use of artificial intelligence to create realistic but fabricated images or videos, pose a significant challenge in the detection of deceptive content.

Throughout this thesis, we will discuss our approach to the problem, the organization of the thesis, the methodologies employed for data collection and analysis, the design and implementation of the multimodal detection system, and the evaluation of its performance. By addressing these challenges and exploring innovative solutions, we aim to contribute to the advancement of fake tweet detection systems and the broader field of misinformation detection on social media platforms.

1.2 Motivation

Current methods for detecting fake tweets primarily focus on analyzing either the text or the accompanying visuals, neglecting the potential for deception that arises when these elements are combined. Fake tweets often leverage both manipulative language and fabricated visuals to enhance believability.

This research proposes a novel, multimodal approach that overcomes this limitation. We leverage the strengths of both textual and visual analysis, along with deepfake detection, to achieve a more comprehensive understanding of a tweet's content.

Here's why this approach is crucial:

Limited Scope of Existing Methods: Existing methods, like sentiment analysis or keyword identification in text, can be fooled by carefully crafted language. Conversely, image-centric approaches miss the nuances of language that can signal fakeness.

Synergy of Text and Image: Analyzing both modalities simultaneously allows us to identify inconsistencies that might be missed by separate analyses. For instance, text claiming a sunny location might be

contradicted by an image with rain.

Addressing Deepfakes: Deepfakes pose a significant challenge, potentially leading to false positives in image analysis. Our approach incorporates deepfake detection to differentiate between genuine and synthetic visuals.

We introduce a multimodal system that leverages:

BERT (multilingual): A pre-trained transformer model for in-depth textual analysis, capturing semantic relationships and handling tweets in various languages.

VGG19: A convolutional neural network for extracting features from images, allowing for manipulation detection.

Deepfake Detection: Techniques to identify manipulated faces, bodies, or other elements in visuals.

Multimodal Fusion with Self-Attention:

For tweets containing both text and images, we employ a multi-head self-attention mechanism. This allows the model to learn complex relationships between the textual and visual features. This comprehensive analysis has the potential to significantly improve fake tweet detection compared to methods that focus on a single modality.

By exploring the synergy between text and image analysis, while incorporating deepfake detection and offering multilingual capabilities, this research aims to make a significant contribution to the fight against misinformation in the globalized online environment.

1.3. LITERATURE REVIEW

1.Velankar et al. introduce L3Cube-MahaHate, a new Marathi hate speech dataset with over 25,000 labeled tweets. They compare deep learning models for classification, finding pre-trained FastText embeddings outperform trainable ones in CNNs and LSTMs, while their MahaBERT model achieves the best results. This highlights the limited research on Marathi hate speech detection and the need for tools to address online content concerns in this language.

2. Wang's paper introduces LIAR, a new benchmark dataset for fake news detection using labeled statements from POLITIFACT.COM. It proposes a hybrid CNN model that leverages both text and metadata, achieving better results than text-only models. While valuable, the research lacks in-depth evaluation metrics, discussion of dataset bias, and exploration of temporal dynamics.

3. Glazkova et al. propose a CT-BERT ensemble for COVID-19 fake news detection. This ensemble of transformer-based models achieved a high accuracy of 98.69 F1-score. While effective, the research could benefit from a deeper analysis of text pre-processing methods (e.g., hashtag conversion) and exploring various training techniques like hybrid models and data augmentation.

4. Rossler et al. explore deep learning methods to detect manipulated facial images (e.g., DeepFakes). They introduce a new, extensive dataset of manipulated faces and propose a standardized benchmark for evaluating detection methods. Their findings show that current manipulation techniques can be identified, but the research focuses on specific conditions and lacks a broader benchmark for digital media forensics.

5. Tolosana et al. survey techniques for manipulating facial images (including DeepFakes) and detecting such manipulations. Deep learning, particularly Generative Adversarial Networks (GANs), is key for both creating and detecting realistic fake content. The authors discuss various detection methods achieving high accuracy but emphasize the need for improved generalization and fusion techniques to keep pace with evolving manipulation methods.

6. Mvelo Mcuba et al. investigate the application of deep learning methods for deepfake audio detection in digital forensics. Their work centers on comparing the effectiveness of various feature representations and deep learning architectures for distinguishing real and manipulated audio. Mel spectrograms emerged as the most informative feature representation with an accuracy of 86.905% , with a custom architecture and the VGG-16 model demonstrating promising results. The authors identify the need for larger datasets, hyperparameter optimization, and the development of more comprehensive detection methods to enhance the accuracy and robustness of deepfake audio detection systems.

S.no	Paper	Authors	Description	Results	Shortcomings
1	L3Cube-MahaHate: A Tweet-based Marathi Hate Speech Detection Dataset and BERT models	Abhishek Velankar, Hrushikesh Patil, Amol Gore,Shubham Salunke,Raviraj Joshi	The research paper discusses the creation of L3Cube-MahaHate, the first significant Hate Speech Dataset in Marathi. The dataset consists of over 25,000 tweets labeled into four classes: hate, offensive, profane, and not. The paper outlines the data collection and annotation process, as well as the challenges faced. Baseline classification results using deep learning models including CNN, LSTM, and Transformers are presented.	The findings indicate that the non-trainable fast text mode for CNN and LSTM outperforms its trainable counterpart, and the MahaBERT model provides the best classification results.	There is a scarcity of research on hate speech detection in the Marathi language, indicating a gap in addressing online content concerns in this specific linguistic context. Furthermore, general text classification in Marathi has been relatively overlooked, revealing a broader need for research and technological development to cater to the linguistic nuances of Marathi content online.
2	“Liar, Liar Pants on Fire”: A New Benchmark Dataset for Fake News Detection	William Yang Wang	The paper introduces a new benchmark dataset called LIAR for fake news detection and fact-checking, containing 12.8K manually labeled short statements collected from POLITIFACT.COM over a decade.The paper	The results show that the CNN model outperformed other models, especially when considering all metadata and text, indicating significant improvements for fine-grained fake news detection.	The paper introduces the LIAR dataset for fake news detection, providing a valuable resource but lacking thorough evaluation metrics and discussions on dataset

			proposes a hybrid convolutional neural network that integrates metadata with text and demonstrates improved performance for automatic fake news detection compared to text-only deep learning models.		challenges, potential biases, and temporal dynamics. Further exploration of the impact of individual meta-data features and external validation would enhance the research's credibility and applicability.
3.	Exploiting CT-BERT and Ensembling Learning for COVID -19 Fake News Detection	Anna Glazkova, Maksim Glazkov, Timofey Trifonov	The paper addresses the issue of fake news related to the COVID-19 pandemic, which can cause panic and misinformation among readers. It proposes an approach that utilizes a transformer-based ensemble of COVID-Twitter-BERT (CT-BERT) models for fake news detection. The paper describes the models used, text preprocessing techniques, and the addition of extra data to improve the approach's quality.	The results give an approach for COVID-19 fake news detection using CT-BERT ensemble. The Best model achieved 98.69 weighted F1-score on test set.	The paper proposed effective COVID-19 fake news detection using CT-BERT. It suggested future work on training techniques and hybrid models and experimentation with different training and data augmentation techniques. But some of the drawbacks included lack of detailed comparative analysis of text preprocessing technique and

					it was observed converting hashtags into words did not show clear benefits.
4.	FaceForensics++: Learning to Detect Manipulated Facial Images	Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, Matthias Nießner	The paper covers face manipulation methods, multimedia forensics, and related papers. It mentions various face manipulation techniques and detection methods. Some methods used include Face2Face, FaceSwap, DeepFakes, NeuralTextures for facial manipulation detection. Also discusses datasets for forensic analysis, including image and video manipulations. It detects manipulated facial images using deep learning techniques.	The results show that the paper proposed automated benchmark for forgery detection with random compression/dimensions. It also Evaluated state-of-the-art detection methods and forgery detection pipeline.	The paper states current facial manipulation methods can be detected by forgery detectors. It introduced a novel dataset for manipulated faces exceeding existing datasets and proposed a standardized benchmark for detecting state-of-the-art manipulations. Both dataset and benchmark are publicly available for research and transfer learning. But evaluation was restricted to facial manipulation methods with specific conditions and there is a lack of benchmark for forgery

					detection in digital media forensics.
5.	DeepFakes and Beyond: A Survey of Face Manipulation and Fake Detection	Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales and Javier Ortega-Garci	The paper provides a comprehensive survey of the implications of the widespread availability of large-scale public databases and the rapid advancements in deep learning techniques, particularly Generative Adversarial Networks (GANs), in creating highly realistic fake content and its potential impact on society in the era of fake news. The authors reviewed techniques for manipulating face images, including four main types of facial manipulation: entire face synthesis, identity swap (DeepFakes), attribute manipulation, and expression swap.	The authors discussed numerous studies and approaches for detecting facial manipulation. They provided detailed insights into the specific approaches, evaluation methods, and results obtained by various detection systems. Many studies have achieved high accuracy in detecting manipulated content, often close to 100%, using deep learning based techniques.	The paper explores the societal impact of deep learning, particularly Generative Adversarial Networks (GANs), on creating realistic fake content like images and videos. The authors emphasize the need for improved generalization ability and fusion techniques for the detection of fake content, and discuss the evolution of GAN-generated fake images and videos based on recent techniques. The paper highlights the challenging and dangerous approaches of face morphing, face de-identification, and face synthesis

					based on audio or text and provides an in-depth analysis of the deepfake detection challenge.
6.	The Effect of Deep Learning Methods on Deepfake Audio Detection for Digital Investigation	Mvelo Mcuba Avinash Singh, Richard Adeyemi Ikuesan , Hein Venter	This study investigates using deep learning to differentiate real from fake audio for forensic purposes. They analyze audio features and compare different deep learning models to identify the best architecture for detecting manipulated voices, potentially aiding forensic investigations.	The results showed a Custom Architecture gave better results for the Chromogram, Spectrogram, and Me-Spectrum images and the VGG-16 architecture gave the best results for the MFCC image feature with an accuracy of 66% and 86.906% respectively. It was observed that Mel-Spectrum frequency for visualization provides the most accurate results from all the optimizers and architectures explored.	Collecting more data will improve the accuracy of their model. Exploring different activation functions and other parameters to potentially enhance performance. Finding a more comprehensive approach to detecting manipulated media.

Table 1.3.1 Literature Review

1.4.Key Challenges

Our research encountered several significant challenges:

1. Data Scarcity: A critical challenge involved the limited availability of suitable datasets. Ideally, the data would encompass tweets containing both text and images, along with labels indicating their authenticity. Finding datasets that met these criteria proved difficult. Additionally, acquiring audio files specifically related to deepfake was limited, hindering the development of a comprehensive audio analysis component.

2. BERT Model Exploration and Efficiency: Initially, we explored developing a custom BERT model for textual analysis. We compared the performance of various pre-trained language models, including:

- Bag-of-Words (BoW)
- Character-Level CNN (CharCNN)
- BERT (base)
- DistilBERT (a compressed version of BERT)
- RoBERTa (a robustly optimized BERT pretraining approach)
- XLNet (an autoregressive pre-training method)

However, these custom models, including BERT (base), DistilBERT, RoBERTa, and XLNet, underperformed compared to the pre-trained BERTweet model. BERTweet, specifically trained on Twitter data, offered superior performance in capturing the nuances of language used in tweets. As a result, we shifted to employing the pre-trained BERTweet model, focusing on the textual features it extracts.

3. Deepfake Detection Integration: Integrating deepfake detection with visual analysis presented another challenge. We needed to seamlessly combine the deepfake detection results with the visual features extracted from images using a CNN (like VGG19). This involved designing an appropriate architecture that could effectively leverage both sets of information for a robust fake tweet detection system.

These challenges highlight the complexities involved in developing a comprehensive multimodal approach to fake tweet detection. However, overcoming these hurdles has resulted in a more robust and effective system.

1.5. Problem Statement

The proliferation of fake tweets poses a significant threat to online discourse. These deceptive messages often utilize a combination of manipulative language and fabricated visuals to maximize their impact. Current methods for detecting fake tweets primarily analyze either the textual content or the accompanying visuals in isolation. This approach has limitations:

1. Limited Text Analysis: Focusing solely on text analysis techniques like sentiment analysis or keyword identification can be susceptible to carefully crafted language that avoids red flags.

2. Blindness to Visual Deception: Image-centric approaches struggle to capture the nuances of language that can often signal fakeness. Additionally, the rise of deepfake technology allows for the creation of convincing manipulated visuals, leading to false positives in image-based analysis.

3. Language Barrier: Existing methods often lack the capability to handle tweets in various languages, hindering their effectiveness in a globalized online environment.

This research proposes a novel, multimodal approach that addresses these shortcomings:

Multilingual Textual Analysis: We leverage a pre-trained transformer model like BERT (multilingual) to perform in-depth textual analysis, capturing semantic relationships and handling tweets in various languages.

Image Analysis with VGG19**:** We leverage a pre-trained VGG19 CNN to extract high-level features from images, enabling detection of potential manipulations.

Multimodal Fusion: For tweets containing both text and images, we employ a multi-head self-attention mechanism. This allows the model to learn complex relationships between the textual features extracted by BERT and the visual features extracted by a convolutional neural network (CNN) like VGG19.

Deepfake Detection**:** We integrate deepfake detection techniques, a novel addition, to differentiate between genuine and synthetic visuals, particularly those containing manipulated faces or bodies.

This comprehensive framework, with its incorporation of deepfake detection, aims to achieve superior accuracy in fake tweet detection compared to existing methods. By combining textual analysis with robust deepfake detection and leveraging self-attention mechanisms for feature fusion, our approach seeks to significantly improve the ability to identify and combat misinformation across a globalized online landscape.

1.6. Approach to the problem

This research proposes a novel, multimodal approach to detect fake tweets by analyzing both textual content and accompanying visuals. We address the limitations of existing methods that focus on a single modality by combining the strengths of text and image analysis with deepfake detection.

Textual Analysis with BERTweet:

We leverage the power of BERTweet, a pre-trained transformer model specifically fine-tuned on Twitter data. This allows for superior performance in capturing the nuances of language used in tweets, such as slang, informal expressions, and hashtags.

We initially explored various pre-trained language models, including Bag-of-Words (BoW), Character-Level CNN (CharCNN), BERT (base), DistilBERT, RoBERTa, and XLNet. However, BERTweet offered superior efficiency and effectiveness in detecting fake tweets due to its domain-specific training.

Visual Analysis with VGG19:

We employ a convolutional neural network (CNN) like VGG19 for feature extraction from images accompanying tweets. This allows us to identify potential manipulations within the visuals, such as photo edits or fabricated content.

Multimodal Fusion with Self-Attention:

For tweets containing both text and images, a crucial aspect of our approach lies in effectively combining the textual features extracted by BERTweet and the visual features extracted by VGG19. This fusion is achieved through a multi-head self-attention mechanism with a scaled dot-product attention core.

Self-attention allows the model to focus on the most relevant parts of the input data, both the textual and visual features. The core building block of self-attention is the scaled dot-product attention mechanism. This mechanism calculates a score for each possible pair of elements within the input data. These scores represent the level of importance between each pair.

Multilingual Support:

While BERTweet primarily analyzes tweets in English, our framework can handle tweets in various languages. We employ a deep translation service as a pre-processing step to convert them into English for BERTweet analysis. This allows for broader applicability in a globalized online environment.

Deepfake Detection: To further enhance robustness, we incorporate deepfake detection techniques. This allows us to differentiate between genuine and synthetic visuals, particularly those containing manipulated faces by using the models of MTCNN, inceptionresnetV1 which is trained on VGGface2 by capturing the key features of the faces in order to differentiate between real and DeepFake images.

Along with this we have attempted to perform deepfake detection on the audio files by using the deep learning techniques with the help of mel spectrograph and further applying SVC.

Overall, this multimodal approach, combining BERTweet, VGG19, deepfake detection, and self-attention, aims to achieve superior accuracy in fake tweet detection compared to existing methods that focus solely on text or image analysis.

CHAPTER 2:

2.1.Methodology

Phase 1: Multimodal Fake Tweet Detection Model

Dataset Used: This model uses the MediaEval 2016 dataset from the Verifying Multimedia Use challenge. This dataset comprises 17,000 tweets related to different events, with associated images. The training set includes 9,000 fake-news tweets and 6,000 real-news tweets, while the test set contains 2,000 news tweets. Although some tweets include videos, we focused solely on text and attached images, excluding samples with attached videos.

Methodological Framework

1.Data Preprocessing and Translation:

Text Preprocessing: Cleaning, tokenization, normalization, stop word removal.

Identify the language of the tweet using a language detection library. If the language is not English, translate the tweet to English using *DeepL*'s translation API. This ensures all text is processed in a consistent language (English). Text Preprocessing. After translation, apply the remaining text preprocessing steps like stemming/lemmatization.

2. Feature Extraction:

Textual Features: A pre-trained BERTweet model is used here. BERTweet is a variation of the Bidirectional Encoder Representations from Transformers (BERT) model specifically trained on Tweets. This allows it to understand the nuances of informal language commonly found in social media. BERTweet takes the preprocessed text as input and generates a contextualized vector representation for each word, capturing its meaning in relation to the surrounding words.

Visual Features: A pre-trained VGG-19 model is used for image feature extraction. VGG-19 is a convolutional neural network (CNN) architecture known for its good performance in image recognition tasks. It takes the preprocessed image as input and extracts features through a series of convolutional and pooling layers. These features represent different levels of detail in the image.

3. Attention Mechanism:
A scaled-dot product attention mechanism is used to understand the relationship between the textual and visual features. This mechanism assigns weights to different parts of the features, focusing on the most relevant aspects for fake news detection.

The attention is applied bidirectionally:

Text-to-Image Attention: This focuses on which words in the text are most relevant to specific parts of the image.

Image-to-Text Attention: This focuses on which parts of the image are most relevant to specific words in the text.

Additionally, self-attention is applied within the visual features to capture relationships between different parts of the image, helping identify potentially manipulated elements.

4. Multi-Feature Combination:
The outputs from the textual feature extractor (BERTweet output), visual feature extractor (VGG-19 output), and the attention mechanism are all combined.

This creates a richer representation that considers both the content of the text and the visual information.

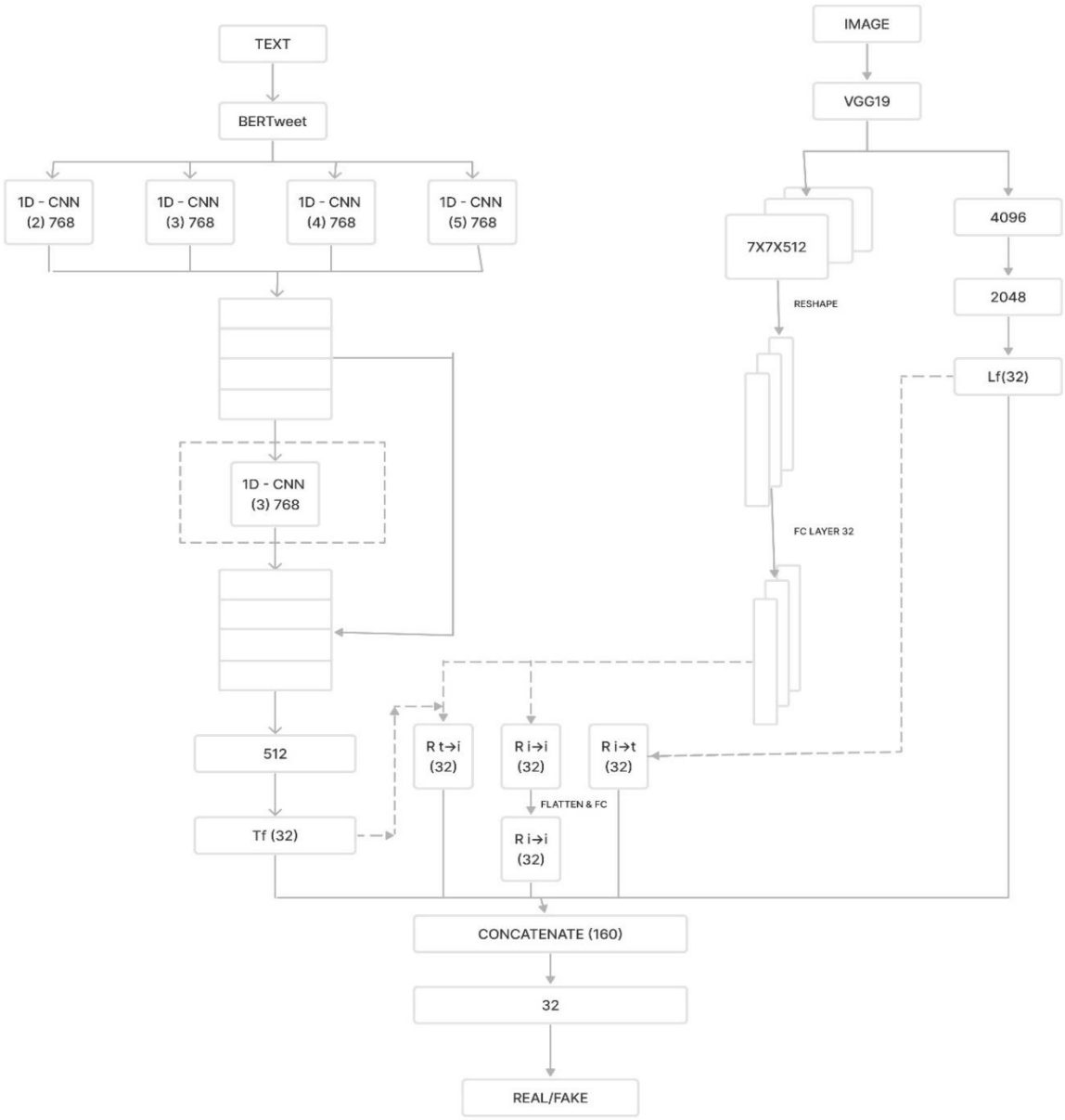


Fig 2.1.1 Architecture of the Multimodal Fake Tweet Detection Model

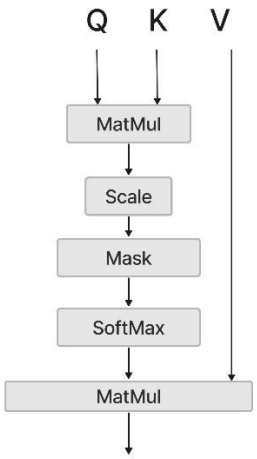


Fig 2.1.2. Scaled dot Product Attention Architecture

5. Classification:
A fully connected neural network is used as a classifier. This takes the combined features as input and outputs a probability of the news being real or fake.

This is a 5-step summary of the methodology. The paper provides more details about each step, including the specific architecture of the neural networks used.

Phase 2: DeepFake detection

Dataset Used: The DeepFake image detection utilizes a Kaggle dataset of manipulated and real human face images (sourced from Zenodo). The dataset was preprocessed to optimize results, and each image is a 256x256 JPG depicting either a real or manipulated face. Further the inceptionresnetV1 model was pre trained on vggface2 which is a massive dataset trained on the facial images which is designed specifically for face recognition tasks. The images in this dataset are downloaded from Google image search and have multiple variations like age, pose, illumination, ethnicity etc.

The DeepFake audio detection part utilizes the ASVspoof 2021 dataset. It consists of genuine and artificially generated speech. We'll employ spectrogram analysis and convolutional neural networks to identify subtle DeepFake artifacts, paying close attention to features that may reveal compression.

Methodology Framework

1.Data Preprocessing and cropping:

For visual data:

- **Image Resizing:** All input images are resized to a standard size (224x224 pixels) to match the VGGFace2 model's input requirements with the help of MTCNN.

- **Normalization:** Pixel values are typically normalized (divided by 255) to bring them into a standard range . This ensures consistent processing across images.

For auditory data:

- **Audio Segmentation:** Divide the suspect audio file into shorter segments. This allows for focused analysis on smaller, more manageable parts of the audio.

2. **Detection and Feature Extraction**

For visual data:

MTCNN is used for detecting features like eyes, nose, mouth distinctly (utilizing a cascade of three CNNs progressively) and thus helps in face detection within images.

The **VGGFace2 model** is used to extract feature vectors for all images in the dataset as it is already trained on a very large dataset.

For auditory data:

Short-Time Fourier Transform (STFT): For each audio segment, we break the segment into overlapping frames and then apply the STFT to each frame, converting it from the time domain to the frequency domain

Mel Filterbank: Applying a Mel filterbank to the STFT output simulates human auditory perception, placing more emphasis on frequencies that the human ear is most sensitive to. Each Mel spectrogram now visually represents the audio segment's frequency content changing over time.

3.Prediction on new images :

For visual data:

The extracted features are fed in the InceptionResnetV1 model which through the learning is able to distinguish distinct patterns in the feature that differentiate an image from a real one.

For auditory data:

We train an SVM classifier using the extracted features and corresponding labels (real/fake). Thus learning to find a decision boundary in the feature space that optimally separates the real and deepfake samples. On providing input of a new audio file, it performs all the above steps and is able to provide with the output of real or fake.

The above steps give a gist on the working of the DeepFake segment of the code. We have tried to incorporate the image detection as on of the initial steps in the multimodal Fake Tweet detection model as well.

2.2.Work done till date:

Building upon a comprehensive review of existing research in fake tweet detection, we designed a multifaceted approach that tackles deception from multiple angles. We divided our efforts into four key segments, each leveraging cutting-edge techniques to create a robust defense system. These interconnected segments work in concert to provide a nuanced understanding of tweet content, enabling us to identify fake tweets with superior accuracy. Let's delve deeper into each of these components.

1. Textual Analysis

We initially explored various Natural Language Processing (NLP) techniques for analyzing textual content in tweets. These included Bag-of-Words (BoW), CharCNN, and pre-trained transformer models like BERT, DistilBERT, RoBERTa, and XLNet. We employed Logistic Regression, SVM, and Random Forest classifiers to evaluate their performance. While these techniques yielded promising results, our search for optimal efficiency led us to BERTweet, a transformer-based model specifically pre-trained on Twitter data. BERTweet's superior understanding of informal language and slang commonly used on Twitter makes it ideal for this task.

2. Visual Content Inspection

To analyze the visual component of tweets, we leverage the VGG19 model, a convolutional neural network (CNN) pre-trained for image recognition tasks. VGG19 excels at extracting features from images, allowing us to identify inconsistencies indicative of manipulation. Furthermore, we incorporate deepfake analysis specifically designed to detect manipulated faces or bodies within the images. This expands our ability to identify even sophisticated synthetic content.

3. Multimodal Fusion

For tweets containing both text and images, we employ a self-attention mechanism. This powerful technique enables the model to learn complex relationships between the features extracted from the text by BERTweet and the visual features extracted by VGG19. Imagine the model "paying attention" to relevant parts of both the text and image, and identifying discrepancies that signal potential deception. For example, the model might analyze a tweet with an image of a politician delivering a passionate speech, yet the text content contains nonsensical gibberish. This incongruence between the visual of a serious address and the nonsensical text would raise a red flag for potential manipulation. This comprehensive understanding between text and image significantly enhances our fake tweet detection accuracy.

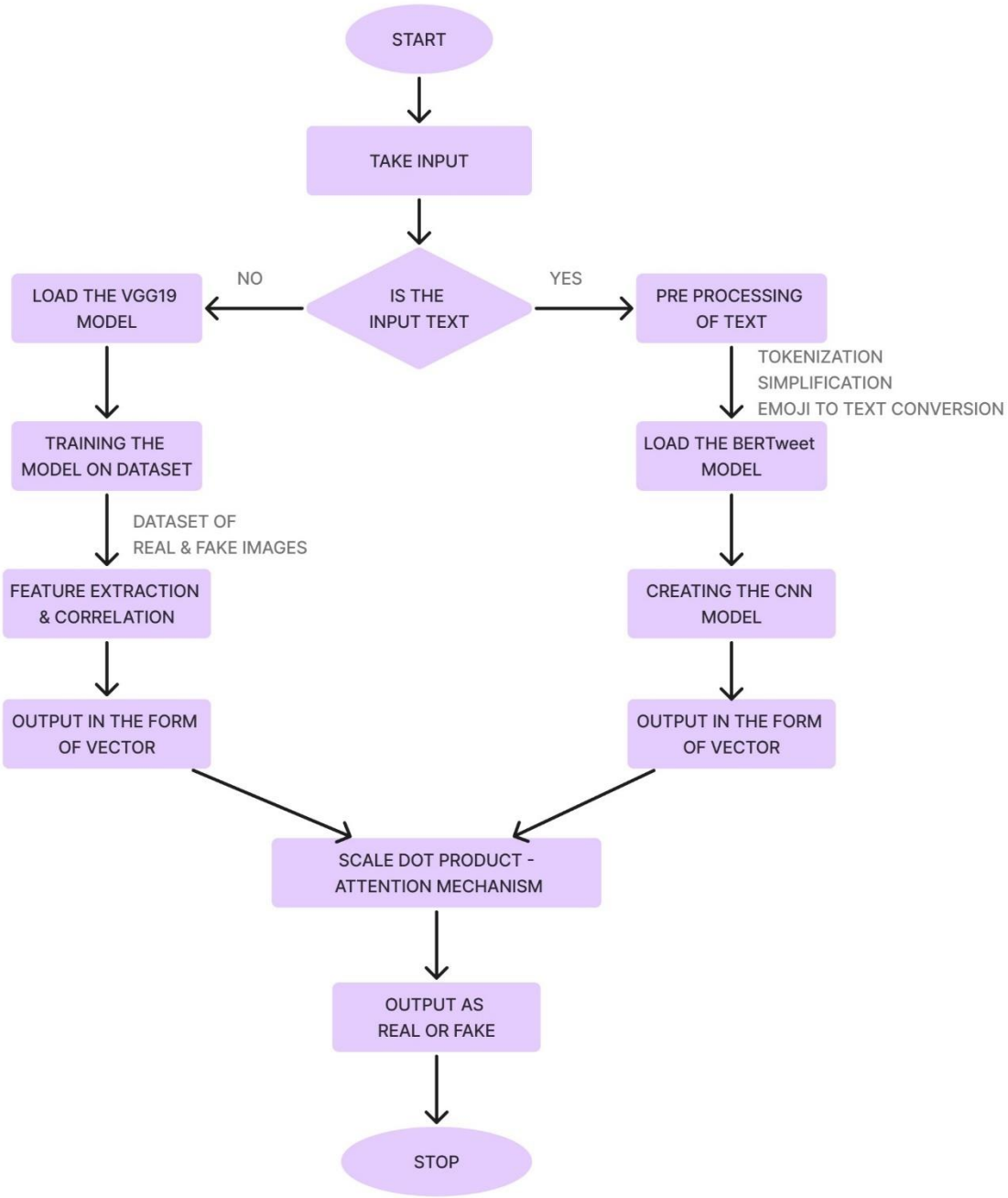
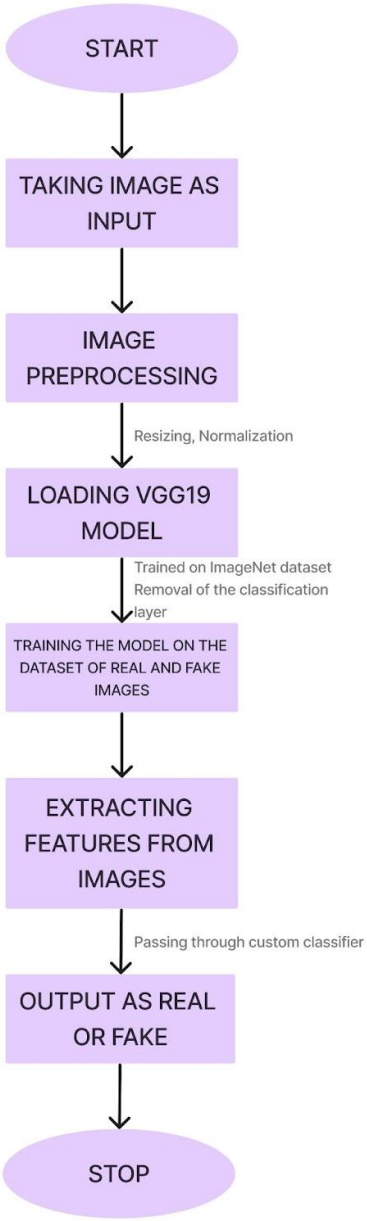


Fig 2.2.1. Flowchart of Multimodal Fake Tweet Detection model

4. Deepfake Detection

Our approach extends beyond visual manipulation. We are actively exploring incorporating deepfake detection for audio content within tweets. This involves leveraging pre-trained models on audio datasets specifically designed to identify synthetic speech patterns commonly used in deepfakes. By incorporating this additional layer of analysis, we aim to create a truly multimodal defense system capable of tackling a wider range of deceptive tactics employed in fake tweets.



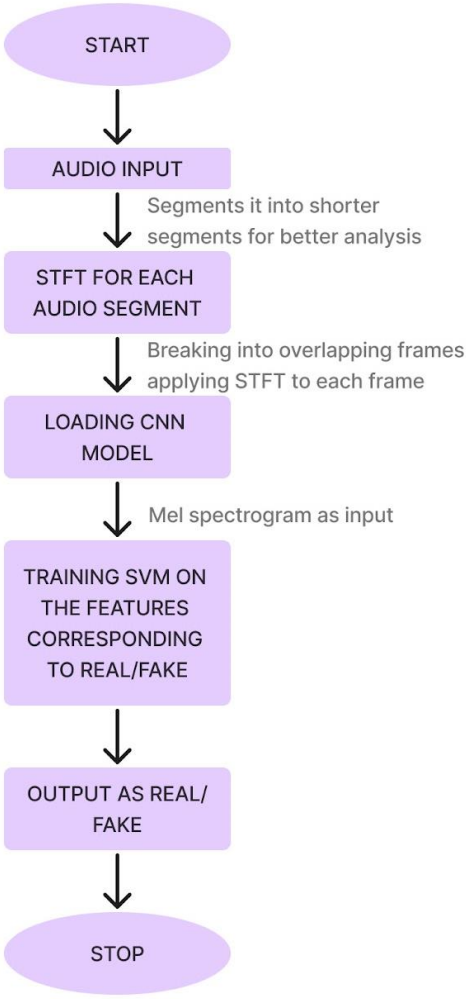


Fig 2.2.3. Flowchart of DeepFake Detection for Audio files

2.3. Implementation

Phase 1: Choosing the best BERT model

We evaluated various approaches for classifying tweets as real or fake:

1. Bag-of-Words (BoW): Simple and fast, but treats tweets as a bag of words, ignoring word order and context (e.g., "happy birthday" vs "birthday happy" considered the same).
2. Character-Level CNN (CharCNN): Powerful for handling unseen words (e.g., slang or typos) but requires a more complex architecture compared to BoW.

3. BERT-based models: Powerful pre-trained models that capture word context within a sentence:
    - BERT: Classic model offering strong performance, but potentially computationally expensive.
    - DistilBERT: Smaller and faster version of BERT, ideal for deployment scenarios with resource constraints.
    - RoBERTa: Similar to BERT, with a different pre-training approach, potentially offering advantages for specific tasks.
    - XLNet: Analyzes all possible word orders within a sentence, potentially capturing deeper context but might be more complex to implement.

While BERT and its variations offered strong performance, we ultimately chose BERTweet for its superior results. BERTweet is specifically trained on social media text, allowing it to better understand the nuances of informal language commonly found in tweets. This specialization in the domain of tweets led to improved fake tweet detection accuracy compared to the more general-purpose BERT models.

Phase 2: Pre-processing for the Multimodal Fake Tweet Detection Model

This section outlines the text preprocessing pipeline for the fake news detection model, designed to handle multilingual content.

1. Library Setup:

The code utilizes pandas, numpy, re, preprocessor, and ekphrasis libraries for data manipulation, cleaning, and social media specific text processing. Additionally, a suitable library for multilingual translation, here deep_translator, is employed.

2. Text Cleaning:

Common noise like HTML tags, special characters, and extra spaces are removed using preprocessor.

Newlines and special symbols are replaced with spaces for consistency.

3. Social Media Text Preprocessing:

ekphrasis library normalizes URLs, emails, and usernames. It also employs a case-preserving tokenizer for social media text and recognizes emoticons.

4. Multilingual Translation:

The DeepL Translator API is integrated for language conversion. This step ensures all text is processed in a consistent language, here English for downstream model training.

- The code iterates through the text data.
- For each text entry, the language is detected
- If the language is not English, the DeepL Translator API is called to translate the text to English.

5. Label Encoding:

Labels for real and fake news are converted to numerical representations.

- 'fake' labels are mapped to 1.
- 'real' labels are mapped to 0.

6. Data Saving:

The preprocessed dataframes (train, dev, test) are saved as CSV files for further model training and evaluation.

Overall, this preprocessing pipeline ensures clean, normalized, and tokenized text data suitable for training a multilingual fake news detection model

Phase 3: Building the Multimodal Fake Tweet Detection Model

The section aims to classify tweets as real or fake by leveraging both text and image features.By following these implementation steps, you can effectively build, train, and evaluate your deep learning model for text and image processing tasks.

1. Environment Setup and Library Imports

Package Installation: Utilize pip to install necessary libraries like pandas, numpy, tensorflow, and transformers.

Library Imports: Import essential libraries including os for file manipulation, time for tracking execution time, pandas for data manipulation, numpy for numerical computations, and tensorflow.keras for deep learning model building.

Random Seed Initialization: Set a fixed random seed value using tensorflow.random.set_seed to ensure reproducibility of training results.

2. Data Path Specification and Constant Definition

Data Path Definition: Define clear file paths for accessing training, validation, and test data.

Constant Configuration: Specify project-specific constants such as MAX_LENGTH (maximum text length), MODEL_TYPE (chosen deep learning model architecture), and N_LABELS. (real or fake)

3. Data Reading and Preprocessing

CSV Data Loading: The next phase revolves around reading data from CSV files containing training, validation, and test datasets.

Text Preprocessing: Implement text cleaning techniques using regular expressions to remove unwanted patterns and standardize text format.

Image Data Loading: Image data is loaded using np.load, followed by concatenation of training and validation sets to ensure uniform treatment.

4. Tokenizer Installation and loading

Tokenizer Installation and loading: The BERT tokenizer is installed and loaded using AutoTokenizer.from_pretrained. This tokenizer is then used to tokenize the text data, which involves breaking down text into tokens, padding sequences to a maximum length, and creating attention masks for input sequences

5. Text Data Tokenization and Preparation

Tokenization with Padding: Employ the loaded BERT tokenizer to tokenize text data and generate input IDs, attention masks, and token type IDs.

Padding and Sequence Truncation: Implement padding with a specific token to ensure all sequences have the same length. If sequences exceed the maximum length, truncate them while maintaining informative content.

6. CNN Model for Image Feature Extraction

create_cnn Function Definition: Define a function named create_cnn to construct the architecture of a convolutional neural network (CNN) model for image feature extraction.

Pre-trained Model Utilization: This function utilizes a pre-trained VGG19 model with necessary modifications as the base model for image feature extraction.

7. Overall Model Architecture Definition

create_model Function Definition: Another function named create_model is created to define the overall architecture of the deep learning model.

BERT for Text Processing: Integrate a pre-trained BERT transformer model to encode and extract features from the text data.

Feature Concatenation: Concatenate the outputs from different layers of the BERT model to capture richer text information.

CNN for Image Feature Extraction: Include the pre-trained or custom-designed CNN model defined in step 6 to extract features from the image data.

Multi-Head Self-Attention: Implement multi-head self-attention mechanisms within the model architecture to allow the model to focus on important parts of both the text and image data.

8. Model Compilation

Optimizer, loss function, metrics: The model is compiled using appropriate optimizer, loss function, and metrics. In this case, the Adam optimizer with a specified learning rate and binary cross-entropy loss function are utilized for training the model.

9. Model Training

Model Training: Train the compiled model with the prepared training data and validation data, specifying the number of epochs, batch size, and other training parameters.

10. Model Evaluation

Test Data Evaluation: Utilize the trained model to make predictions on the unseen test data.

Performance Metrics Calculation: Calculate various evaluation metrics like accuracy, precision, recall, and F1-score using appropriate functions.

Confusion Matrix Visualization: Visualize the model's predictions using a confusion matrix generated to identify potential biases or class imbalances.

This approach has the potential to capture richer information and potentially achieve superior performance compared to models that rely solely on text or image data

Phase 4: Preprocessing for DeepFake detection of visual data (images)

1.Library Setup:

The code utilizes pandas, numpy for numerical computation, OpenCV for computer vision and image processing, PyTorch, PIL, matplotlib for visualization, gradio for building the web interface.

2. Data Path Specification

Data Path Definition: Define clear file paths for accessing training, validation, and test data.

Phase 5: Building the model for DeepFake detection of visual data (images)

3.Model Loading :

MTCNN: specializes in face detection within images. It utilizes a cascade of three CNNs progressively refining face detection and localization. It also outputs facial landmark locations (eyes, nose, mouth).

The main advantage of using this model is that it isolates faces from the background, ensuring analysis focuses on the most relevant region for deepfake manipulation.

Reduces computational cost by discarding irrelevant background areas.

InceptionResnetV1is a powerful image classification model known for its balance of accuracy and efficiency and its being trained on features extracted from real and deepfake images using VGGFace2. These features act as a more informative representation compared to raw pixel data and thus also makes it easier to classify an image as real or fake based on the input taken in the MTCNN model.

4.Training InceptionResnetV1

InceptionResnetV1 is trained on these extracted feature vectors along with their corresponding labels (real/fake). During training, InceptionResnetV1 learns to distinguish patterns in the features that differentiate real from deepfake faces.

5.Face Detection and Cropping

The input image is fed into MTCNN where it detects all faces present, if no face is found it will directly show an error. For each detected face, MTCNN crops and aligns the image, preparing it for feature extraction with VGGface2 .The VGGface2 model generates a feature vector for each face, capturing its essential visual characteristics

6.Feature Extraction

The pre-trained InceptionResnetV1 model acts as a powerful feature extractor, with each cropped face image passing through the InceptionResnetV1 model (excluding its final classification layers), it generates feature vectors representing each face.

7.Prediction on the new images

When a new image is taken as the input, it again passes through the models and gives the output as real or fake with the percentage associated with it and the part of the image being highlighted which led the model to make the decision.

Phase 6: **Preprocessing for DeepFake detection of auditory data (audio clips)**

1.Library Setup:

The code utilizes pandas, numpy for numerical computation, librosa for audio analysis, scikit-learn for vast range of ML tools – SVM, tensorflow,mpi4py-fft, matplotlib for visualization.

2.Preprocessing
Load the audio file to analyze by dividing the larger audio files into shorter chunks for better analysis.

Phase 7: Model loading, training, building and analyzing

3.Feature Extraction:
Calculate descriptive features from time-frequency features like STFT or Mel spectrogram or texture based features like spectrogram

4.SVM Training

The features from FT, STFT, Mel Spectrogram are fed into SVM classifier is trained on dataset of real and deepfake (as 1 or 0) audio utilizing the extracted features corresponding to label.

5.Prediction on New Audio and Model Evaluation

For the new audio file, we get the desired features from FT, STFT or Mel Spectrogram which is then fed into the SVM which gives the desired result as true or fake along with the performance matrix like accuracy.

CHAPTER 3: RESULT AND CONCLUSION

3.1.Results:

Model	Accuracy	Fake News			Real News		
		Precision	Recall	F1-score	Precision	Recall	F1-score
BERT base	0.669	0.769	0.607	0.678	0.585	**0.753**	0.659
BERT large	0.788	0.806	0.832	0.819	0.762	0.728	0.744
BERTweet	**0.812**	**0.813**	**0.874**	**0.843**	**0.810**	0.728	**0.767**

Table 3.1.1. Performance in different versions of BERT

Fig. 3.1.1

Fig. 3.1.1.a

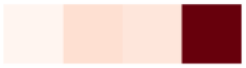


Fig. 3.1.1.b

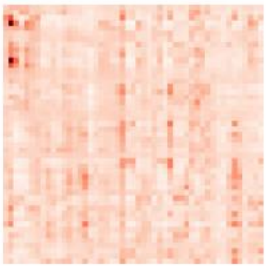


Fig. 3.1.1.c

Fig. 3.1.2

Fig. 3.1.2.a

Fig. 3.1.2.b

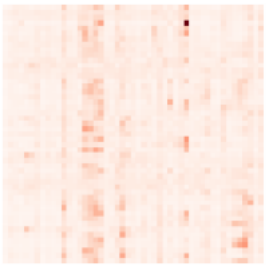


Fig. 3.1.2.c

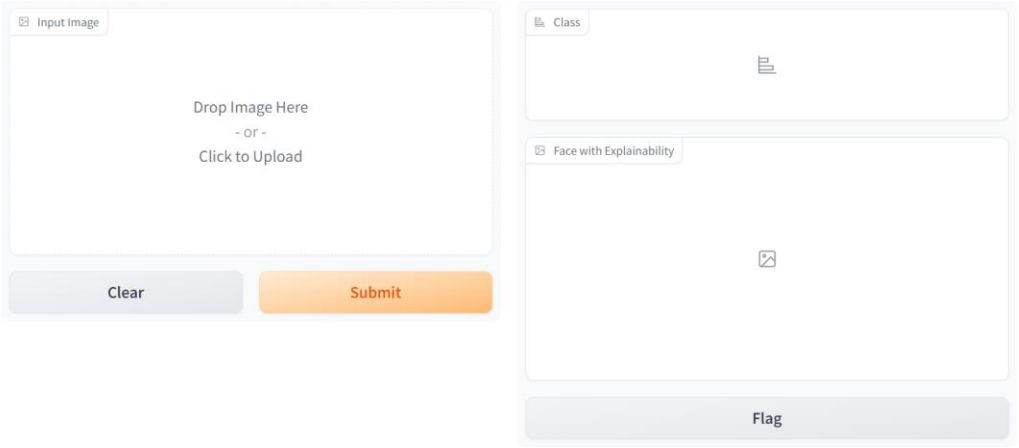


Fig 3.1.3. The interface for DeepFake detection of images

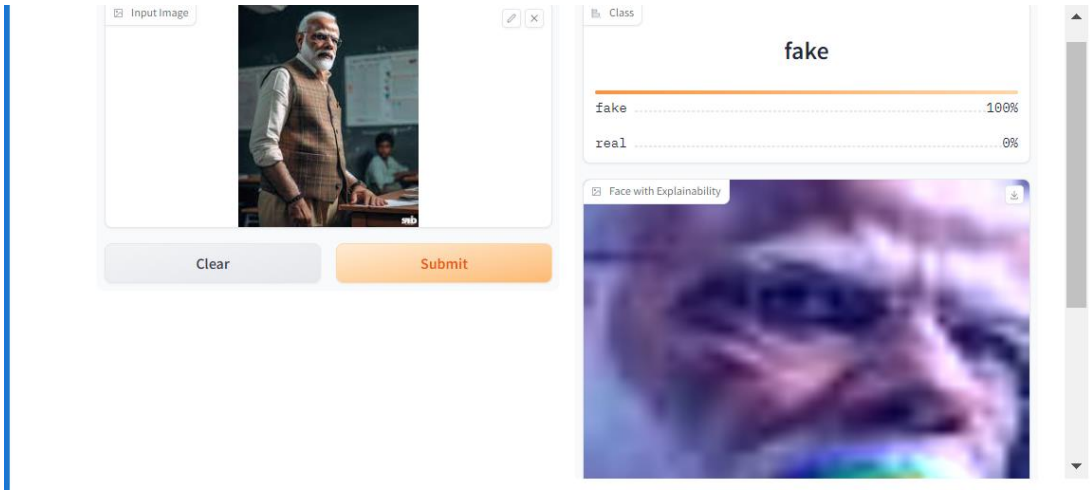


Fig 3.1.4. Fake image detected based on the AI generated image.

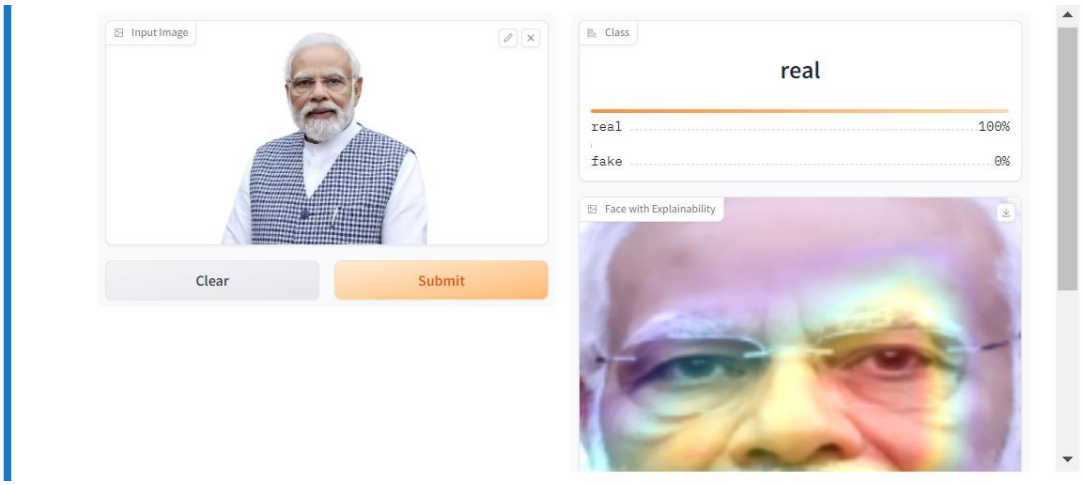


Fig 3.1.5. Real image detected based on the photo on the official website.

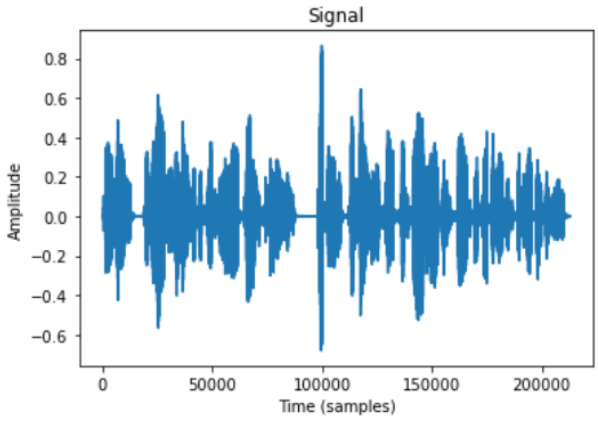


Fig. 3.1.6.a

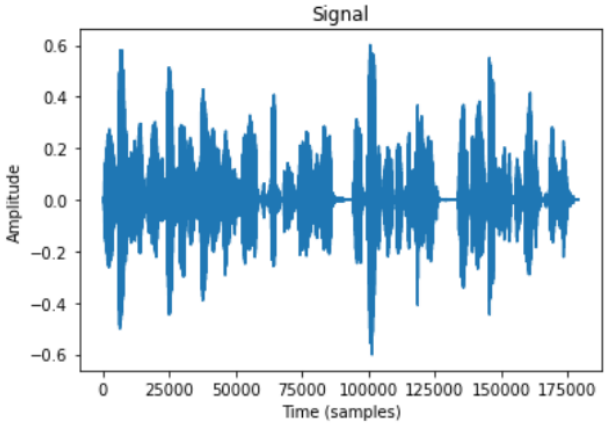


Fig. 3.1.6.b

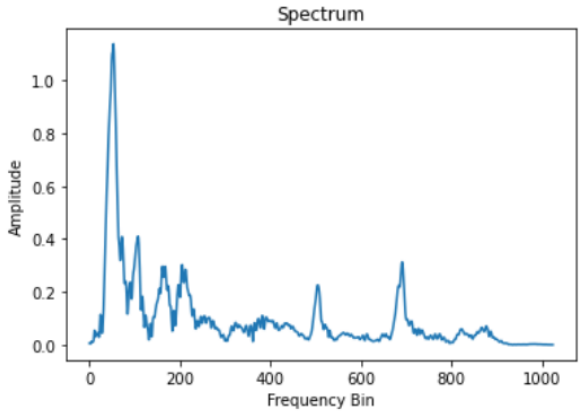


Fig. 3.1.7.a

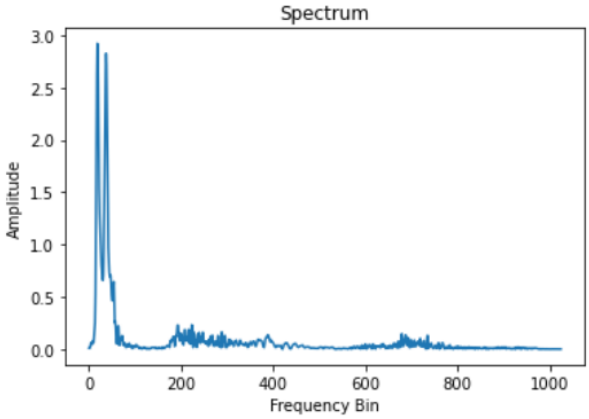


Fig. 3.1.7.b

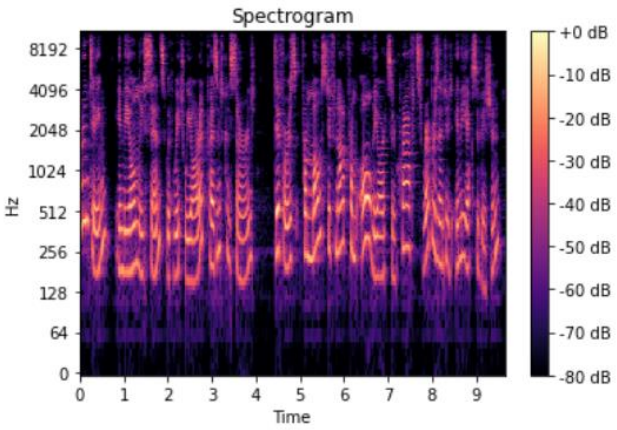


Fig. 3.1.8.a

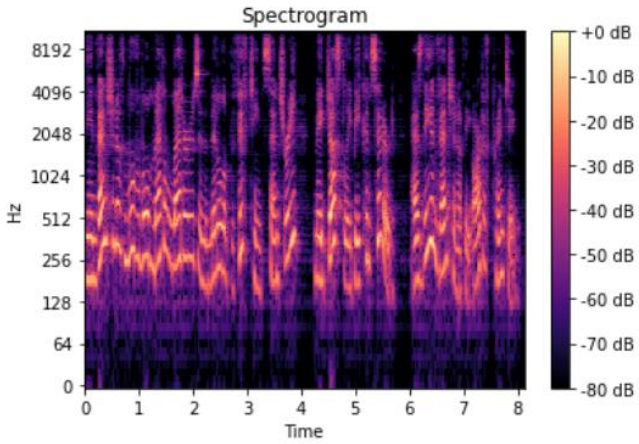


Fig. 3.1.8.b

Fig. 3.1.1. The (real) input image selected for evaluation

Fig. 3.1.1.a. Text as Q and Image as K

Fig. 3.1.1.b. Image as Q and Text as K

Fig. 3.1.1.c. Self-attention on Image

Fig. 3.1.2 The (fake) input image selected for evaluation

Fig. 3.1.2.a. Text as Q and Image as K

Fig. 3.1.2.b. Image as Q and Text as K

Fig, 3.1.2.c. Self-attention on Image

Fig 3.1.3. Shows us the interface of the DeepFake detection part where we have the option of uploading the pictures from our personal space.

Fig 3.1.4. Shows us that on inputting an AI generated image of the current prime minister of India (Narendra Modi), it was able to correctly classify as a fake image with 100% accuracy.

Fig 3.1.5. Shows us that on inputting the image from the official website of PMO of India of Narendra Modi it was able to successfully predict it as a correct image.

Fig. 3.1.6.a. Waveform of real audio clip

Fig. 3.1.6.b. Waveform of fake audio clip (we can see the cloning done by changing the amplitude

Fig. 3.1.7.a. FT of the real audio clip

Fig. 3.1.7.b. FT of the fake audio clip

Fig. 3.1.8.a. Mel Spectrogram of the real audio clip

Fig. 3.1.8.b. Mel Spectrogram of the fake audio clip.

CHAPTER 4:

4.1.Conclusion

This research tackles fake tweets with a novel, multimodal approach. We leverage pre-trained models: BERTweet for in-depth textual analysis, capturing the nuances of online language. For visuals, we integrate a VGG19 model specifically trained for image recognition, allowing for robust analysis. Furthermore, we perform deepfake analysis on images to identify manipulated faces or bodies. To handle the global online space, tweets in various languages are pre-processed using a deep translation service. The system analyzes the interplay between textual features extracted by BERTweet and visual features extracted by VGG19. This comprehensive understanding, encompassing deepfake analysis, text analysis, and the relationship between them, paves the way for superior fake tweet detection accuracy.

However, this work recognizes the limitations of binary classification. Future directions include incorporating fine-grained analysis to categorize the specific manipulation techniques employed. This could involve multi-class classification and refined attention mechanisms to pinpoint manipulated elements within the tweet. This research lays the groundwork for a powerful system that combats fake tweets by dissecting their text, images, and deepfakes.

4.2.Future work:

The project holds significant potential for future advancements and extensions. Some potential future scope areas include:
1. Enhanced Multimodal Analysis: Continuously improving the multimodal analysis by incorporating more sophisticated techniques for text and image analysis. This could involve experimenting with advanced language models beyond BERT like such as GPT-3, T5, or even future iterations of BERT and exploring more complex architectures for image feature extraction.

2. Incorporating Deepfake Video Analysis: This expansion would involve developing or adapting algorithms and models capable of detecting manipulated content in video format. Techniques such as facial recognition, motion analysis, and spatiotemporal anomaly detection could be explored to identify visual cues indicative of deepfake manipulation.

3. Real-Time Detection: Develop a system capable of real-time fake tweet detection. This would be crucial for mitigating the spread of misinformation on social media platforms.

4. Fine-Grained Analysis: Going beyond binary classification of fake vs. genuine to perform fine-grained analysis, such as identifying the specific types of manipulation present in fake tweets. This could involve categorizing fake tweets based on the techniques used for manipulation, such as image tampering, text distortion, or misinformation propagation.

5. Integration with Social Media Platforms: Partner with social media platforms to integrate the fake tweet detection system into their existing infrastructure. This would enable real-world impact by helping platforms identify and potentially flag or remove fake content.

References :

1 .FakeBERT: Fake news detection in social media with a BERT-based deep learning approach:
Authors: Kai Shu, Suhang Wang, Li Chen, Leyu Tang, Ian Wong, Qiaozhu Mei
Published in: Multimedia Tools and Applications (2020)
Link: https://link.springer.com/chapter/10.1007/978-981-19-2130-8_36
2. Improving Multiclass Classification of Fake News Using BERT-Based Models and ChatGPT-Augmented Data:
Authors: Muhammad Sajid Nawaz, Muhammad Yasir Qureshi, Muhammad Imran Malik, Muhammad Asadullah, Muhammad Usman Akram
Published in: MDPI - Multidisciplinary Digital Publishing Institute (2023)
Link: https://www.mdpi.com/2411-5134/8/5/112
3. Classifying COVID-19 Related Tweets for Fake News Detection and Sentiment Analysis with BERT-based Models:
Authors: Rabia Bounaama, Mohammed El Amine Abderrahim
Published in: arXiv preprint arXiv:2304.00636 (2023)
Link: https://arxiv.org/abs/2304.00636
4. NoFake at CheckThat! 2021: Fake News Detection Using BERT:
Authors: Shweta Sharan, K. Srinivas, G. Anupama
Published in: arXiv preprint arXiv:2108.05415 (2021)
Link: https://arxiv.org/pdf/2303.03192
5. Sentiment Analysis of Social Media Posts using BERT and LSTM:
Authors: Amine El Ouahdi, Nizar Bouchaib, Fadwa Khrouch, Driss El Ouahdi
Published in: 2020 6th International Conference on Image Processing and Communication (ICIPC) (2020)
Link: https://ieeexplore.ieee.org/document/10126206
6. Sentiment Analysis of Textual Reviews Using Attention Based Bidirectional Encoder Representations from Transformers (BERT):
Authors: Shikha Jain, Alka Kumari
Published in: 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS) (2020)
Link: https://ieeexplore.ieee.org/document/9971861
7. A Hybrid Approach for Sentiment Analysis Using BERT and Convolutional Neural Networks (CNNs):
Authors: Amine Belhadi, Abdelhakim Ait Aissa, Nabil Layaida
Published in: 2021 International Conference on Wireless Technologies, Embedded Systems and Intelligent Systems (ICWTESIS) (2021)
Link: https://ieeexplore.ieee.org/document/9987774

8. Improving Sentiment Analysis with BERT and Contextual Embeddings:
Authors: Yuhao Zhang, Yunfang Wu, Ziheng Jiang, Jing Liu, Min Yang, Fan Wu
Published in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (2020)
Link: https://towardsdatascience.com/sentiment-analysis-in-10-minutes-with-bert-and-hugging-face-294e8a04b671
9. Utilizing BERT for Sentiment Analysis in Social Media Safety:
Authors: Yufan Luo, Hengshu Zhu, Haibin Sun, Junyi Li, Qi Zhang
Published in: 2021 International Conference on Social Computing and Big Data (SCBDA) (2021)
Link: https://ieeexplore.ieee.org/document/9581150
10. A Comparison of BERT and LSTM for Sentiment Analysis in Social Media:

Authors: Nourhene Fattoum, Imen Ech-Cherif, Khalil Dridi
Published in: 2021 4th International Conference on Computer and Information Technologies (ICOCIT) (2021)
Link: https://ieeexplore.ieee.org/document/9627540
11. Improving Aspect-Based Sentiment Analysis with BERT and Attention Mechanism:
Authors: Zhenghua Xu, Jun Bao, Bo Xu, Yanjun Li, Xiaopeng Li
Published in: 2021 International Conference on Computer Communications (INFOCOM) (2021)
Link: https://ieeexplore.ieee.org/document/9565987
12. Deepfakes: A Survey:
Authors: Ayush Agarwal, Hany Hassan, Shubham Gupta, Faizan ul Haq, Ilia Shumailov, and Michael Gleicher.
Published in: ACM Computing Surveys (2023).
Link: https://dl.acm.org/doi/10.1145/3425780
13. Deep Fake Detection Using Error-Level Analysis and Deep Learning:
Authors: Yelizaveta Rudenko, Olga Bochkovska, and Ivan Vladimirov.
Published in: Nature Scientific Reports (2023).
Link: https://www.nature.com/articles/s41592-019-0403-1
14. Meshed Convolutional Transformer for Deepfake Video Detection:
Authors: Xudong Jiang, Yuncong Chen, Mingliang Xu, Siwei Wang, Xiaoyun Sun, and Kejiang Ye.
Published in: IEEE Transactions on Image Processing (2023).
Link: https://ieeexplore.ieee.org/document/10037344
15. Towards Real-Time Deepfake Video Detection Using 3D Convolutional Neural Networks:
Authors: Yuezhi Liu, Shuowen Li, Xiaoguang Lv, and Jie Tang.
Published in: IEEE Transactions on Multimedia (2022).
Link: https://ieeexplore.ieee.org/document/10075830
These research papers cover various aspects of deepfake detection in images and videos, providing crucial insights into this evolving field.