Mona Kumari Kumari

Mona_Dissertation_final.pdf

Indian Statistical Institute

Document Details

Submission ID
trn:oid:::3618:101120782

Submission Date
Jun 16, 2025, 11:20 AM GMT+5:30

Download Date
Jun 16, 2025, 11:25 AM GMT+5:30

File Name
Mona_Dissertation_final.pdf

File Size
1.7 MB

61 Pages

11,543 Words

70,446 Characters

3% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 14 words)

Custom Section Exclusions

{titlesCount} Section Titles, {keywordsCount} Keywords

Section title	No. of Section Starters	Section Starters
"isi"	2	Indian statistical institute, kolkata

Match Groups

13 Not Cited or Quoted 3%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

3% Internet sources
3% Publications
0% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- **13** Not Cited or Quoted 3%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

3% Internet sources
3% Publications
0% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	dspace.isical.ac.in:8080 <1%
2	Publication	Wittkopp, Thorsten. "A Layered Architecture for Log Analysis in Complex IT Syste... <1%
3	Internet	docslib.org <1%
4	Internet	library.isical.ac.in:8080 <1%
5	Internet	www.geeksforgeeks.org <1%
6	Internet	springer.marka.pt <1%
7	Internet	technodocbox.com <1%
8	Internet	core.ac.uk <1%
9	Internet	www.biorxiv.org <1%
10	Internet	ebin.pub <1%

11 Internet

export.arxiv.org <1%

12 Internet

www.vs.inf.ethz.ch <1%

Detection of Fake News in Short Videos: A Multimodal Approach

Mona Kumari

Detection of Fake News in Short Videos: A Multimodal Approach

DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology
in
Computer Science (with specialization in Data Science)

by

Mona Kumari

[Roll No: CS2311]

under the supervision of

Prof. Ujjwal Bhattacharya
Professor
Computer Vision and Pattern Recognition Unit

Indian Statistical Institute
Kolkata 700108, India

May 2025

CERTIFICATE

This is to certify that the dissertation entitled **"Detection of Fake News in Short Videos: A Multimodal Approach"** submitted by **Mona Kumari** to Indian Statistical Institute, Kolkata, in partial fulfillment for the award of the degree of **Master of Technology in Computer Science (with specialization in Data Science)** is a bonafide record of work carried out by her under my supervision and guidance. The dissertation has fulfilled all the requirements as per the regulations of this institute and, in my opinion, has reached the standard needed for submission.

Prof. Ujjwal Bhattacharya
Professor,
Computer Vision and Pattern Recognition Unit,
Indian Statistical Institute,
Kolkata 700108, INDIA.

To my parents,
for their enduring support and strength.

And to my advisor,
for the guidance and clarity that shaped this work.

Acknowledgments

It is with profound gratitude that I acknowledge the invaluable guidance of my advisor, *Prof. Ujjwal Bhattacharya*, Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata. His mentorship, deep insights, and consistent encouragement played a pivotal role in shaping this research.

I sincerely thank all faculty members and researchers at the Indian Statistical Institute for fostering a vibrant academic environment and providing the resources and freedom necessary for meaningful exploration.

My heartfelt appreciation also goes to my peers and labmates for their camaraderie, constructive discussions, and the shared learning experiences that enriched this journey.

Finally, I am deeply indebted to my familyespecially my parentsfor their unwavering support, sacrifices, and faith in me. Their love and strength have been the cornerstone of my academic pursuit.

Mona Kumari
Indian Statistical Institute
Kolkata 700108, India.

Abstract

The rise of generative models and affordable video editing tools has fueled the spread of fake and manipulated videos, undermining information reliabilityespecially on social media. Traditional detection methods, focused on single modalities like visual artifacts or text cues, often struggle with diverse, user-generated content.

This dissertation presents a unified framework for fake video detection that integrates multimodal semantics, narrative structure, and propagation behavior. Visual, audio, text, and OCR features are extracted using pretrained models (CLIP, Wav2Vec2), and segment-level graphs are built to model narrative flow using Graph Attention Networks (GATv2Conv). User engagement dynamics are modeled via a bidirectional LSTM.

A cross-modal consistency loss encourages semantic alignment across modalities, improving representational coherence. The end-to-end model is evaluated on heterogeneous datasets like FakeTT, demonstrating strong generalization and robustness.

Results show the proposed system outperforms existing baselines, especially in challenging cases with asynchronous or fragmented content. By combining content, structure, and behavioral cues, the framework enables more reliable and interpretable fake video detection.

Keywords: *Fake Video Detection, Multimodal Representation Learning, Graph Neural Networks, Temporal Modeling, Cross-Modal Consistency, Misinformation Detection*

Contents

List of Figures

List of Tables

List of Algorithms

Abbreviations

Abbreviation	Full Form
AUC	Area Under the Curve
BiLSTM	Bidirectional Long Short-Term Memory
CLIP	Contrastive Language-Image Pretraining
FVD	Fake Video Detection
GATv2	Graph Attention Network v2
GCN	Graph Convolutional Network
GNN	Graph Neural Network
HET	Heterogeneous Dataset
HOM	Homogeneous Dataset
LPE	Laplacian Positional Embedding
LSTM	Long Short-Term Memory
MLP	Multi-Layer Perceptron
OCR	Optical Character Recognition
PR	Precision-Recall
Wav2Vec2	Waveform-Based Speech Representation Model

Commonly used abbreviations throughout this dissertation.

Chapter 1

Introduction

The widespread availability of generative models such as GANs and diffusion networks has made the creation of fake videos both easy and convincing. These manipulated videos often appear visually authentic and emotionally persuasive, posing serious challenges to public trust, media integrity, and online information ecosystems.

Unlike static misinformation, fake videos are inherently multimodal combining visuals, speech, text overlays, and metadata. In many cases, only specific components are manipulated; for instance, the audio may be altered while the visuals remain intact, or subtitles may misrepresent the spoken content. These partial manipulations often evade detection by unimodal systems.

Complicating matters further is the way such content spreads. Fake videos frequently go viral through coordinated user engagement, mimicking organic popularity and bypassing traditional filters that rely solely on content analysis.

To address these challenges, this dissertation proposes a unified detection framework that integrates three critical dimensions multimodal content understanding, narrative structure modeling, and propagation behavior analysis. By leveraging pretrained encoders, graph-based reasoning, and temporal modeling, the system aims to improve detection accuracy, enhance interpretability, and remain robust against real-world noise and manipulation strategies.

1.1 Motivation

The growing sophistication of fake videos has made their detection increasingly difficult. Many existing methods focus on a single modality such as text, visuals, or audio yet real-world manipulations often exploit subtle cross-modal inconsistencies. For example, the spoken audio may not align with lip movements, or subtitles may present a misleading

narrative inconsistent with the original speech. These nuanced discrepancies are often missed when modalities are analyzed in isolation.

In addition to content-level manipulation, the spread of such videos on social media presents another challenge. Fake content often exhibits unusual propagation patterns, such as sudden spikes in engagement or coordinated sharing by inauthentic accountssignals that are frequently ignored in current models.

This work aims to bridge these gaps by introducing a system that:

- Checks consistency across visual, audio, and text streams;
- Uses Graph Neural Networks to model the narrative flow of video segments;
- Analyzes propagation patterns to detect suspicious online behavior;
- Handles real-world videos that may be noisy, low quality, or partially edited.

By integrating content, structure, and behavioral cues, the system provides a more comprehensive approach to fake video detection.

1.2 Problem Statement

The rise of manipulated media demands robust methods for fake video detection. Traditional approaches that focus on a single modalitysuch as visual or audio cuesare insufficient, as modern fake videos often exhibit subtle inconsistencies across multiple channels.

In this work, each video is modeled as a combination of five modalities:

- **Visual features** (CLIP-based): capturing scene-level semantics.
- **Audio features** (Wav2Vec2): encoding speech and acoustic signals.
- **Textual features**: semantic and emotional cues from transcripts.
- **OCR features**: on-screen text extracted from video frames.
- **Propagation signals**: time-series data of social engagement.

The task is to learn a function:

$$f(v) \rightarrow \{0, 1\} \tag{1.1}$$

where v is a multimodal video sample, and the output label indicates whether the video is `real` (0) or `fake` (1).

The model must also be robust to noisy or missing data and provide interpretability by identifying modality-level evidence for its predictions. Our proposed system addresses these needs through multimodal fusion, narrative graph reasoning, temporal propagation modeling, and cross-modal consistency learning.

1.3 Research Objectives

The main goal of this dissertation is to better detect fake videos, not just based on visual or audio data, but by combining different aspects of the video such as its content, story flow, and people's online interactions with the video.

The central objectives of this dissertation are:

1. Design a flexible and modular system that can analyze multiple types of information visuals, audio, text, and the pattern of video dissemination in a single framework
2. Develop a graph-based model that helps understand the structure of a video, specifically to see how different segments are connected and how the overall narrative flows from beginning to end.
3. Using time-based modeling techniques to study patterns of user engagement such as how quickly a video goes viral or how it is shared over timeso that unusual or suspicious behavior can be spotted.
4. Introducing a consistency-check method so that the system can detect cross-modal mismatches such as audio not matching visuals, or text being out of context.

1.4 Thesis Contributions

This work makes the following key contributions:

- A novel hybrid architecture integrating CLIP-based visual-text features and Wav2Vec2-based audio embeddings within a shared multimodal encoder.
- A narrative flow modeling module utilizing Graph Attention Networks (GATv2Conv) over segment-level embeddings to capture inter-clip coherence.
- A propagation modeling head employing LSTM-based sequence modeling to represent dynamic user engagement behavior.

- A consistency-aware loss formulation that regularizes predictions by aligning representations across content, graph, and behavioral branches.
- Comprehensive empirical validation, including ablation studies, benchmarking on public datasets, and visualization of learned decision patterns.

1.5 Thesis Organization

The remainder of this dissertation is structured as follows:

- **Chapter 2:** Covers essential preliminaries, including multimodal embeddings, attention mechanisms, GNNs, and LSTM-based sequence models.
- **Chapter 3:** Reviews related work in fake video detection, multimodal misinformation modeling, and propagation-aware systems.
- **Chapter 4:** Describes the proposed methodology in detail, including architectural components, data representations, and loss functions.
- **Chapter 5:** Outlines the training pipeline, implementation details, hyperparameter settings, and computational complexity.
- **Chapter 6:** Presents experimental results, including quantitative evaluation, ablation studies, qualitative analysis, and discussion.
- **Chapter 7:** Concludes the dissertation and suggests future research directions based on current limitations and findings.

Chapter 2

Preliminaries

This chapter provides the essential background concepts that underpin the design of the fake video detection system. We cover multimodal embeddings, attention-based transformer architectures, graph neural networks (GNNs), long short-term memory (LSTM) models, and techniques for multimodal fusion. These components serve as the theoretical foundation for the subsequent architectural and experimental discussions.

2.1 Multimodal Representation Learning

Multimodal representation learning focuses on capturing complementary signals from diverse data types such as images, audio, and text. Each modality encodes distinct information about the same entity or event, and fusing them enables more robust inference.

- **Visual Embeddings:** Frame-level visual features are typically extracted using pretrained vision models such as CLIP or ResNet, which map images into semantically rich embedding spaces.
- **Audio Embeddings:** Audio streams are converted into contextual feature vectors using transformer-based encoders like Wav2Vec2, which are trained to capture phonetic and tonal variations in speech.
- **Textual Embeddings:** Semantic and emotional aspects of transcribed speech can be captured using language models like BERT , with specialized variants for sentiment analysis.
- **OCR Embeddings:** Text detected via optical character recognition (OCR) from video frames is processed using sentence encoders to obtain fixed-length vector representations.

Each embedding is projected into a shared latent space to facilitate cross-modal interactions.

2.2 Attention and Transformer Models

Transformers have become the de facto standard for modeling sequential and structured data. At the core of these architectures is the self-attention mechanism.

2.2.1 Self-Attention

Self-attention allows each token in a sequence to attend to every other token based on learned similarity scores:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \tag{2.1}$$

Here, Q, K, and V represent the query, key, and value matrices respectively, and d_k is the dimensionality of the key vectors.

2.2.2 Multi-Head Attention

To enable the model to capture information from different representational subspaces, multiple attention heads are employed:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \tag{2.2}$$

where each attention head is defined as:

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \tag{2.3}$$

This mechanism facilitates simultaneous attention to multiple semantic aspects within the input.

2.3 Graph Neural Networks

Graphs are powerful data structures for modeling relational dependencies. A graph $G = (V, E)$ is composed of nodes V and edges E that represent relationships. Graph Neural Networks (GNNs) are designed to learn node representations by aggregating information from neighbors.

2.3.1 Message Passing Framework

A typical GNN layer operates by updating node embeddings based on a message-passing paradigm:

$$h_i^{(l+1)} = \text{UPDATE}\left(h_i^{(l)}, \text{AGGREGATE}\left(\{h_j^{(l)} : j \in \mathcal{N}(i)\}\right)\right) \tag{2.4}$$

where $h_i^{(l)}$ is the embedding of node i at layer l, and $\mathcal{N}(i)$ denotes its set of neighbors.

2.3.2 Graph Attention Networks (GATv2)

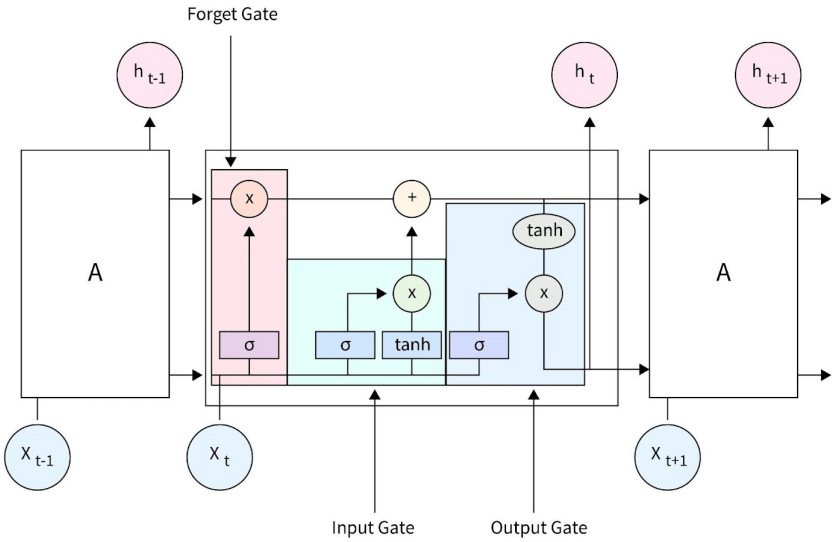
GATv2 is a variant of GNNs that improves upon standard GAT by enabling input-dependent attention coefficients. The attention between nodes i and j is computed as:

$$\alpha_{ij} = \frac{\exp\left(\text{LeakyReLU}(a^\top [Wh_i \| Wh_j])\right)}{\sum_{k \in \mathcal{N}(i)} \exp\left(\text{LeakyReLU}(a^\top [Wh_i \| Wh_k])\right)} \tag{2.5}$$

This dynamic weighting allows the model to focus on more relevant neighbors, improving representation learning in sparse or irregular graphs.

2.4 Long Short-Term Memory (LSTM)

LSTM is a type of recurrent neural network (RNN) designed to address long-range dependency issues in sequential data. It uses gating mechanisms to control the flow of information across time steps.



$$
\begin{aligned}
f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f) \\
i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i) \\
o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o) \\
c_t &= f_t \odot c_{t-1} + i_t \odot \tanh(W_c x_t + U_c h_{t-1} + b_c) \\
h_t &= o_t \odot \tanh(c_t)
\end{aligned}
\tag{2.6}
$$

LSTMs design makes it effective for modeling temporal propagation patterns and sequential user behaviors.

2.5 Propagation Modeling Concepts

Propagation modeling involves analyzing how content spreads over time. This can include features such as:

- **Engagement Metrics:** Time-series data capturing views, likes, shares, and comments.
- **Temporal Granularity:** Segmenting user interaction logs into fixed intervals (e.g., hourly or daily).
- **Sequential Encoding:** Using RNNs or LSTMs to model the evolution of these metrics over time.

Such patterns are particularly useful in detecting coordinated or anomalous behavior associated with synthetic media.

2.6 Multimodal Fusion Strategies

Integrating information across modalities is critical for multimodal systems.The choice of fusion strategy depends on modality alignment, task complexity, and the target representation space. Common strategies include:

- **Late Fusion:** Process each modality independently and combine predictions at the decision level.
- **Early Fusion:** Concatenate raw or embedded features before feeding them into a shared model.
- **Cross-Modal Attention:** Use attention mechanisms to align and refine features from one modality based on another.
- **Gated Fusion:** Introduce gating functions to weigh modality contributions dynamically during inference.

Chapter 3

Literature Review

In this chapter, we provide an overview of the key research areas that help build our fake video detection model. These areas provide an overview of techniques for detecting deepfakes, ways to handle misinformation in different data types such as text, audio, and visuals, graph based reasoning that helps understand structural links within videos, and learning methods that look at how content spreads over time. We discuss major contributions in each area, identify critical limitations in current methodologies, and outline the research gap our model addresses. Finally, we show how our model combines all of these areastaking into account clues from content, structure, and user behaviorto provide a strong and practical solution for detecting fake videos.

3.1 Fake Video Detection

Early approaches to fake video detection primarily focused on visual anomalies at the pixel level. Matern et al. [1] identified deepfakes by analyzing temporal flickering and unnatural eye-blinking. Afchar et al. [2] proposed MesoNet, a lightweight CNN capable of detecting facial forgery by leveraging mesoscopic features. Sabir et al. [3] utilized recurrent convolutional architectures to model temporal dependencies and detect inconsistencies across frames. Although these methods performed well on curated datasets, they lacked robustness when applied to real-world content, which is often compressed or modified. Furthermore, these vision-only systems failed to account for semantic inconsistencies or multimodal interactions, such as incongruences between audio, video, and text.

3.2 Multimodal Misinformation Detection

To handle real-world generalization better, recent work combines visual, audio, and text cues. Zhou et al. [4] built a joint vision-language model for spotting rumors in multimodal posts. Wang et al. [5] created M3ER, using emotion-aware audio+text features

to detect fake speech. Jin et al. [6] applied recurrent neural fusion to align tweets with their images and videos.

But there are gaps: Most current multimodal models assume perfect sync between audio, video, and textthey do not handle temporal misalignment well. They also cannot capture how inconsistencies or narrative drift develop *over time* in longer content. Another issue is the lack of explicit modeling for structural relationships between segments (e.g., scene transitions or argument flow), limiting their ability to detect sophisticated cross-modal forgeries.

3.3 Graph-Based Content Reasoning

Graphs work really well for modeling connections between video segments, components, or different modalities. We build on GCNs introduced by Kipf and Welling [7], and the Graph Attention Networks (GAT) of Velickovic et al. [8], which add attention to the neighbor-aggregation process. Zhang et al. [9] further demonstrated the power of graph-based representations for multimodal misinformation detection by linking visual, text, and audio features in a unified graph.

But heres the catchwhen we inspect current GNN implementations for deepfake and fake video detection, most skip two crucial aspects:

1. How narrative coherence is maintained across segments within a single video.
2. Meaningful structural connections between visual components (such as objects, faces, and background).

To address these gaps, we propose a subgraph-level attention mechanism that focuses on segment transitions and component evolution, rather than relying solely on frame-level features (see Figure 3.1).

Figure 3.1: Subgraph-level attention capturing segment connectivity and component transitions.

3.4 Where Current Research Falls Short

Despite a wealth of publications, three fundamental deficiencies remain. First, most approaches suffer from modality myopia focusing exclusively on visual cues or, at best, audio visual fusionwhile ignoring the critical tri-modal interplay among video, audio, and text. Consequently, temporal misalignments (e.g., asynchronous lip movements) and erroneous captions are left unmodeled, leaving significant manipulation cues undetected.

Second, although graph neural networks such as GCNs and GATs have been extensively developed, they are rarely applied to capture narrative coherence across segments. Scene transitions and contextual continuity are typically overlooked in favor of isolated framelevel features, preventing the discovery of highlevel structural inconsistencies.

Third, propagation dynamics are routinely relegated to an afterthought: engagement metrics (shares, views, reposts) are tacked on posthoc rather than integrated into the training objectiveakin to seasoning a dish only after its fully cooked. This disjointed strategy precludes the creation of a unified pipeline that jointly learns from content, narrative structure, and dissemination behavior. As a result, current state-of-the-art systems resemble loosely coupled modules, each addressing only a portion of the fake video detection challenge.

3.5 Critical Gaps in Current Detection Frameworks

While recent years have seen impressive advances in deepfake detection, several persistent limitations undermine real-world effectiveness particularly for cross-modal misinformation. Let's examine where the field falls short:

** The Multimodal Blind Spot** It's surprising how many detection systems still operate in modal silos. Most either fixate on visual artifacts or clumsily handle video-audio pairs [1,2]. This neglects the *temporal asymmetries* between visual, auditory, and textual streams in sophisticated fakes. Consider lip movements drifting from audio, or captions contradicting visual contextexisting architectures lack the joint representation power to catch these discrepancies. Our tri-encoder framework specifically targets this weakness through cross-attention mechanisms.

Structural Navet Graph networks revolutionized relational modeling [3,4], yet their application to video narrative structure remains curiously underdeveloped. Few studies leverage GNNs to model semantic dependencies between temporal segments [5]a critical oversight given how deepfakes often manipulate *scene transitions*. Without modeling inter-segment relationships, systems miss injected narrative breaks. Our hierarchical GATv2Conv architecture explicitly captures these transitional semantics.

** Propagation Modeling as an After thought** Many papers pay lip service to propagation signals but treat them as mere metadata features [6]. This artificial separation between content and spread dynamics creates two problems: (a) behavioral patterns aren't contextualized with semantic content, and (b) temporal engagement patterns get reduced to aggregate statistics. We argue propagation should be *learned jointly*hence our dedicated LSTM propagation head consuming raw engagement timelines.

** The Integration Deficit** Perhaps most fundamentally, the field suffers from fragmented solutions. To our knowledge, no existing framework *end-to-end unifies* content

understanding, structural coherence, and propagation dynamics [7]. This compartmentalization while simplifying research designsseverely limits detection capability against coordinated disinformation campaigns. Our models consistency-optimized joint training explicitly bridges these dimensions.

3.6 Motivation and Positioning of Our Work

To overcome the multifaceted limitations in current fake video detection methods, our proposed framework integrates key capabilities from multimodal learning, graph reasoning, and propagation analysis into a unified pipeline. This architecture is specifically designed to capture misalignments across modalities, model narrative coherence, and incorporate behavioral cues within a single trainable system.

First, we employ a **Multimodal Content Encoder** that combines embeddings from CLIP for visual features [10], Wav2Vec2 for audio [11], and BERT-based language models for textual content (including transcriptions and OCR overlays) [12]. This fusion enables the system to identify inconsistencies between what is seen, heard, and said.

Second, we introduce a **Narrative Flow Model** that organizes videos as graphs of segments. Each node corresponds to a video subcomponent, and edges reflect semantic and temporal continuity. GATv2Conv layers are applied to capture dynamic inter-segment dependencies and ensure narrative coherence [8].

Third, a **Propagation Modeling Head** captures behavioral patterns by processing engagement time-series data such as shares, views, and repost timing through a bidirectional LSTM [13]. This module learns propagation signals as part of the overall training process, rather than as a separate post-hoc step.

Finally, our **Consistency-Aware Loss Function** aligns embeddings across modalities and structural contexts. It enforces agreement between multimodal content, graph-based transitions, and propagation dynamics, resulting in a more interpretable and generalizable detection framework [14].

By combining these elements in a single architecture, our model offers robust detection capabilities for real-world fake videos, including those involving subtle manipulations, asynchronous signals, and complex dissemination behavior.

Chapter 4

Proposed Methodology

This proposed architecture is inspired by the FakingRecipe research paper for detecting fake news videos [15]. This architecture looks at several aspects of how the video is made, how the story is told, and how its shared online. In comparison, many existing methods focus only on the content of the videosuch as its visuals, audio, and accompanying text. However, this work identifies two often-overlooked aspects: the temporal coherence of the narrative and the propagation dynamics of the video across social media. The approach integrates three complementary modules that analyze content features, narrative consistency, and online dissemination behavior. A dissemination analysis module monitors the videos spread and user engagement across online platforms.

From different perspectives, modules examine the input video and determine whether the video is authentic or not.

4.1 Overall Architecture

This proposed model architecture (shown in Figure 4.1) is made up of three core modules. A **MultimodalContentEncoder** module is used to process visual, audio, text, and OCR features. A **NarrativeFlowGNN** module is used to understand temporal relationships, while the **PropagationBehaviorHead** module observes how the content spreads across networks. All three modules work in parallel and their respective results are combined to give the final prediction. To ensure better modality alignment and robust generalization, we introduce a cross-modal consistency loss during training [14].

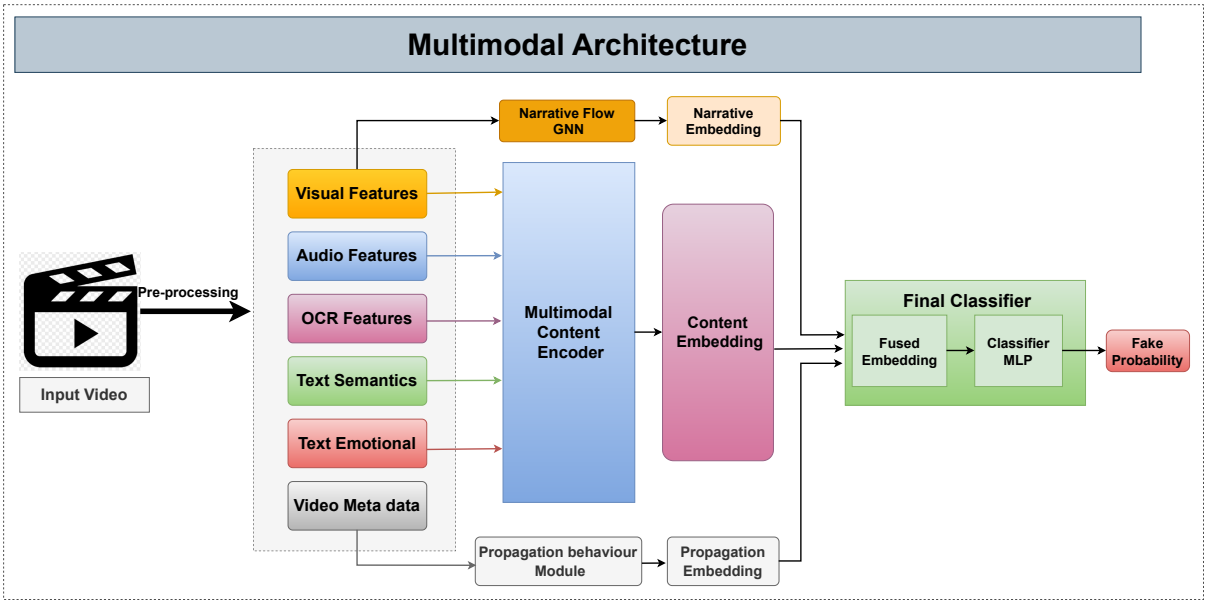


Figure 4.1: Proposed architecture integrating multimodal content encoding, narrative flow modeling, and propagation behavior.

4.2 Data Preprocessing

The proposed fake video detection framework operates on multimodal data including visual, audio, textual, and propagation-based cues. Prior to training, we perform a structured preprocessing pipeline to extract and normalize features from these modalities. This section outlines the complete preprocessing procedure, which prepares raw video inputs into modality-aligned representations suitable for downstream modeling.

Segment-Level Frame Extraction

Each video is first segmented into coherent temporal units using **TransNetV2** [16], a deep neural network for scene boundary detection. For every segment, a representative mid-frame is extracted and saved as an image. These frames form the basis for subsequent OCR and visual feature extraction. The segmentation process ensures that visual and contextual information is aligned across meaningful narrative blocks.

Optical Character Recognition (OCR)

To extract overlaid or embedded textual content from the sampled frames, we utilize the **EasyOCR** library [17]. The OCR model returns detected phrases along with confidence scores for each frame. The results are stored in serialized format and later used for both semantic and emotional text analysis as well as for extracting visual features from the corresponding regions.

Figure 4.2: OCR detection and recognition results across multiple video frames. Left: detected bounding boxes; Right: recognized text with confidence scores.

Visual Feature Extraction via SAM

For each detected text region in the frames, we extract visual features using the **Segment Anything Model (SAM)** [18]. The model is applied to OCR-localized regions within each frame, and features are pooled across segments to obtain a global visual embedding per video. This allows the model to learn from both raw visual scenes and semantically important text-containing regions.

Audio Extraction and Emotion Embedding

The audio stream is extracted using `ffmpeg`, resampled to 16 kHz, converted to mono, and saved as a waveform. Subsequently, we apply the **HuBERT**-based emotion classification model to generate probabilistic emotion embeddings for each videos audio [19]. These embeddings help identify manipulation patterns in speech tone and affective expression.

Textual Semantic and Emotional Analysis

We perform dual analysis on the OCR-extracted text using the **XLM-RoBERTa** model [12]. First, a softmax-based emotion probability vector is computed for each phrase. Then, semantic embeddings are derived from the mean-pooled hidden states of the same model. These semantic and affective features enable the model to detect inconsistencies between visual content and accompanying text.

4.3 Multimodal Content Encoding

A core component of the proposed fake video detection framework is the *Multimodal Content Encoder*. This module is specifically designed to extract and integrate information from diverse modalities, capturing both semantic and affective signals across visual, auditory, and textual channels, including on-screen text. In many manipulated videos, subtle inconsistencies often emerge between what is seen, heard, and read. The objective of this encoder is to model such inconsistencies through structured multimodal analysis and fusion.

Input Modalities

The encoder processes five types of input representations derived from the video:

- **Visual features** extracted using the vision encoder of CLIP [10] from sampled keyframes.
- **Audio features** obtained from Wav2Vec2 embeddings [11], capturing speech-based acoustic and phonetic cues.

- **Semantic textual features** generated from ASR transcripts or subtitles.
- **Emotional textual features** representing the affective tone of the extracted textual content.
- **OCR features** derived from visible on-screen text using EasyOCR [17].

Projection into Common Embedding Space

Each modality is projected into a shared embedding space of dimension $d = 256$ using dedicated linear transformations. Let B denote batch size and T the number of visual segments:

$$\begin{aligned}
\mathbf{H}_{\text{vis}} &\rightarrow \mathbf{H}'_{\text{vis}} \in \mathbb{R}^{B\times T\times d} \\
\mathbf{H}_{\text{aud}} &\rightarrow \mathbf{H}'_{\text{aud}} \in \mathbb{R}^{B\times d} \\
\mathbf{H}_{\text{text-sem}} &\rightarrow \mathbf{H}'_{\text{text-sem}} \in \mathbb{R}^{B\times d} \\
\mathbf{H}_{\text{text-emo}} &\rightarrow \mathbf{H}'_{\text{text-emo}} \in \mathbb{R}^{B\times d} \\
\mathbf{H}_{\text{ocr}} &\rightarrow \mathbf{H}'_{\text{ocr}} \in \mathbb{R}^{B\times d}
\end{aligned}$$

These projections are essential for aligning modalities with varying temporal and spatial granularities.

Temporal Visual Encoding

To encode the sequential nature of visual content, the projected visual features $\mathbf{H}'_{\text{vis}}$ are passed through a unidirectional LSTM [20]. If the lengths of video segments are known, packed sequences are used to optimize efficiency. The final hidden state encapsulates the temporal visual context:

$$\mathbf{H}_{\text{vis-enc}} = \text{LSTM}(\mathbf{H}'_{\text{vis}}) \tag{4.1}$$

Cross-Modal Attention

To model interactions between modalities, we employ multi-head cross-modal attention [21]:

- **Text-to-Visual Attention:** The semantic textual embedding $\mathbf{H}'_{\text{text-sem}}$ attends to $\mathbf{H}'_{\text{vis}}$, yielding the text-aware visual embedding:

$$\mathbf{H}_{\text{text-vis-attn}} = \text{Attention}(\mathbf{H}'_{\text{text-sem}}, \mathbf{H}'_{\text{vis}}) \tag{4.2}$$

- **Audio-to-Visual Attention:** The audio embedding $\mathbf{H}'_{\text{aud}}$ similarly attends to the visual sequence:

$$\mathbf{H}_{\text{aud-vis-attn}} = \text{Attention}(\mathbf{H}'_{\text{aud}}, \mathbf{H}'_{\text{vis}}) \tag{4.3}$$

This mechanism facilitates detection of modal inconsistencies, such as lip-audio mismatch or emotional divergence.

Multimodal Fusion

The core multimodal representation is constructed by concatenating the following features:

- Temporal visual embedding $\mathbf{H}_{\text{vis-enc}}$
- Audio-informed visual embedding $\mathbf{H}_{\text{aud-vis-attn}}$
- Text-informed visual embedding $\mathbf{H}_{\text{text-vis-attn}}$
- Emotional text embedding $\mathbf{H}'_{\text{text-emo}}$
- OCR-derived embedding $\mathbf{H}'_{\text{ocr}}$

These are fused using a multi-layer perceptron (MLP) with layer normalization, ReLU activations, and dropout regularization:

$$\mathbf{H}_{\text{content}} = \text{FusionMLP}\big([\mathbf{H}_{\text{vis-enc}}; \mathbf{H}_{\text{aud-vis-attn}}; \mathbf{H}_{\text{text-vis-attn}}; \mathbf{H}'_{\text{text-emo}}; \mathbf{H}'_{\text{ocr}}]\big) \tag{4.4}$$

Output Representations

The encoder outputs multiple feature representations for downstream use:

- $\mathbf{H}_{\text{content}}$: Final fused multimodal embedding used for classification.
- $\mathbf{H}_{\text{vis-enc}}$: Temporal visual feature.
- $\mathbf{H}_{\text{text-vis-attn}}$: Text-enhanced visual embedding.
- $\mathbf{H}_{\text{aud-vis-attn}}$: Audio-enhanced visual embedding.
- $\mathbf{H}'_{\text{text-emo}}$: Affective text embedding.
- $\mathbf{H}'_{\text{ocr}}$: OCR-derived semantic embedding.

These outputs also support auxiliary objectives such as contrastive losses and narrative reasoning.

4.4 Narrative Flow Graph Construction

To capture the temporal and logical structure of a video, a graph-based representation is constructed over its segmented units. This structure enables the model to reason about content flow and semantic coherence, serving as a foundation for narrative modeling.

Node Construction

Each video is divided into T non-overlapping segments. For each segment, a 512-dimensional visual embedding is extracted using the content encoder. These embeddings form the node features:

$$\mathbf{X} \in \mathbb{R}^{T \times 512} \tag{4.5}$$

where each node corresponds to a temporally localized portion of the video and encodes its visual semantics.

Edge Construction

Two edge types are defined:

- **Temporal Edges:** Link each segment to its successor $(i \to i+1)$ to reflect natural sequence. Each is assigned a fixed weight of 1.0.
- **Content Similarity Edges:** Formed between non-consecutive segments if their cosine similarity exceeds a threshold $\tau = 0.7$. The edge weight equals the similarity score.

This hybrid structure captures both the sequential flow and long-range thematic links in the video narrative.

Graph Representation

The graph is represented in PyTorch Geometric format using:

- Node matrix $\mathbf{X} \in \mathbb{R}^{T \times 512}$
- Edge index matrix `edge_index` $\in \mathbb{N}^{2 \times |E|}$
- Edge attribute matrix `edge_attr` $\in \mathbb{R}^{|E| \times 1}$

For training on multiple videos, individual graphs are batched using `Batch.from_data_list()`.

Structural Cues for Manipulation Detection

In authentic videos, transitions between segments are typically smooth and semantically coherent. Manipulated content often introduces irregularitiessuch as abrupt jumps, repetitive segments, or unnatural transitionsthat disrupt this narrative flow. These inconsistencies manifest as anomalies in the graph structure, allowing the model to learn patterns indicative of manipulation.

4.5 Narrative Flow Modeling

While individual video segments may appear coherent in isolation, manipulated content often introduces subtle inconsistencies across the sequence. These disruptions are better identified when the video is viewed as a structured narrative.

Building on the graph defined in Section 4.4, we apply a Graph Attention Network (GATv2) [8] to model high-level narrative relationships. Each node corresponds to a video segment, and edges capture both temporal order and semantic similarity between segments.

GATv2 enables nodes to attend selectively to their neighbors by learning attention weights. This allows the model to identify inconsistencies in flow, such as illogical transitions or context mismatches. Unlike fixed message passing, attention-based propagation adapts to the narrative structure of each video.

After several layers of graph reasoning, a global pooling operation aggregates the node representations into a single fixed-size embedding. This captures the overall narrative coherence of the video, which is subsequently fused with other modality-specific features-such as content and propagationto predict authenticity.

By incorporating narrative structure, this module helps identify manipulation patterns that may go undetected when analyzing segments independently.

4.6 Propagation Behavior Modeling

This module models user engagement metricssuch as views, shares, and likesover time. The data is represented as a multivariate time-series tensor with three channels and ten temporal intervals. A bidirectional LSTM [20] captures temporal dependencies, followed by an MLP to produce a propagation embedding. This captures anomalies in sharing dynamics, such as sudden virality or inorganic burstscommon in fake content.

4.7 Cross-Modal Consistency Loss

To encourage alignment between different modalities, we introduce a consistency loss that penalizes semantic divergence. Each modalitys embedding is normalized and pairwise cosine similarities are computed across all modality pairs. A cross-entropy loss is applied to maximize similarity for matching pairs and minimize it for unrelated ones [14]. The final loss is symmetric and averaged over all modality pairs:

$$\mathcal{L}_{\text{consistency}} = \sum_{i,j} \frac{1}{2}\big(\text{CE}(\text{sim}(e_i, e_j)) + \text{CE}(\text{sim}(e_j, e_i))\big). \tag{4.6}$$

4.8 Final Classification and Training Strategy

The outputs from the *Multimodal Content Encoder*, *Narrative Flow GNN*, and *Propagation Analyzer* are concatenated to form a unified embedding:

$$E_{\text{combined}} = [E_{\text{content}}; E_{\text{narrative}}; E_{\text{propagation}}]. \tag{4.7}$$

This fused representation is passed through a two-layer MLP classifier with layer normalization, ReLU, and dropout:

$$z = W_2\,\text{Dropout}(\text{ReLU}(\text{LayerNorm}(W_1 E_{\text{combined}} + b_1))) + b_2, \quad p(\text{fake}) = \sigma(z). \tag{4.8}$$

Training Objective: The model is optimized end-to-end using a joint loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{classification}} + \lambda\,\mathcal{L}_{\text{consistency}}, \tag{4.9}$$

where $\mathcal{L}_{\text{classification}}$ is binary cross-entropy and $\mathcal{L}_{\text{consistency}}$ enforces alignment across modalities. The hyperparameter λ controls the balance between accuracy and consistency.

Chapter 5

Algorithms and Complexity

This chapter elaborates the key algorithmic components and training mechanisms underlying the proposed fake video detection system. The architecture integrates multimodal content fusion, narrative flow modeling using graph attention networks (GAT), and temporal behavior analysis via bidirectional LSTM (BiLSTM). The system is capable of capturing cross-modal consistency and semantic coherence between visual narratives and social propagation dynamics.

Each module is carefully designed and optimized for both performance and interpretability. We describe the complete end-to-end pipeline, map the core responsibilities of each module, and provide a formal complexity analysis to bridge theoretical efficiency with practical scalability.

5.1 System Pipeline Overview

The system pipeline is structured to exploit the multimodal, temporal, and structural nature of social videos. It consists of five stages: (i) segment-wise multimodal feature extraction, (ii) cross-modal encoding and fusion, (iii) narrative graph reasoning, (iv) propagation behavior modeling, and (v) final classification.

Algorithm 1 Fake Video Detection Pipeline

Require: Video segments with visual (V), audio (A), semantic/emotional text (T_s, T_e), OCR (O), and metadata (M)

Ensure: A probability score $p \in [0, 1]$ indicating whether the video is real or fake

1: Project all modalities to a shared embedding space: $f_m : \mathbb{R}^{d_m} \to \mathbb{R}^d$
2: Encode visual sequence $V = \{v_1, \ldots, v_N\}$ using LSTM to capture temporal dependencies
3: Apply multi-head attention: $T_s \to V$, $A \to V$
4: Concatenate attended features: $[V'; T'; A'; T_e; O]$ and pass through fusion MLP
5: Build visual narrative graph $G = (V, E)$ and process via GATv2Conv
6: Encode propagation signals $P = \{p_t^{(\text{views})}, p_t^{(\text{likes})}, p_t^{(\text{shares})}\}_{t=1}^{T}$ via BiLSTM
7: Concatenate all embeddings and classify via multilayer perceptron MLP_{clf}

5.2 Training Flow

The training objective is binary classification (real vs. fake). A composite loss function is used:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{bce}} + \lambda \cdot \mathcal{L}_{\text{consistency}} \tag{5.1}$$

- $\mathcal{L}_{\text{bce}}$: standard binary cross-entropy loss.
- $\mathcal{L}_{\text{consistency}}$: semantic alignment penalty across modality-specific embeddings.
- $\lambda \in \mathbb{R}_{\geq 0}$: a tunable coefficient for regularization strength.

The training begins with feature extraction from segmented videos:

- Visual embeddings: $f_V : \text{CLIP}(x) \to \mathbb{R}^d$
- Audio embeddings: $f_A : \text{Wav2Vec2}(x) \to \mathbb{R}^d$
- Textual (semantic/emotional) and OCR embeddings: f_T, f_{T_e}, f_O
- Propagation time series: $P \in \mathbb{R}^{N \times 3}$ (views, likes, shares)
- Segment-level graph $G = (V, E)$ with $|V| = N$

Algorithm 2 Training Procedure for Fake Video Detection

Require: Labeled dataset of videos and metadata $\{(\mathcal{V}_i, \mathcal{M}_i, y_i)\}_{i=1}^{N}$

Ensure: Trained model parameters θ^* and optimal threshold τ^*

1: For each video $\mathcal{V}_i$, segment and extract modality features:

$$V \leftarrow \text{CLIP}, \quad A \leftarrow \text{Wav2Vec2}, \quad T_s, T_e, O \leftarrow \text{text/OCR}, \quad P \leftarrow \text{metadata series} \tag{5.2}$$

2: Project each modality x_m to shared space: $f_m(x) = W_m x + b_m$

3: Apply LSTM on V and cross-modal attention: $T_s \to V$, $A \to V$

4: Fuse all content modalities via MLP to obtain h_C

5: Construct visual narrative graph $G = (V, E)$ and compute h_G via stacked GATv2Conv layers

6: Encode propagation signal P via BiLSTM to obtain h_P

7: Concatenate all embeddings: $h_{\text{final}} = [h_C; h_G; h_P]$

8: Pass h_{final} through classification head to compute logit z and probability $p = \sigma(z)$

9: Compute total loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{bce}} + \lambda \cdot \mathcal{L}_{\text{consistency}} \tag{5.3}$$

10: Update parameters using AdamW optimizer

11: Apply regularization and augmentation techniques:

- Class-weighted or focal loss (for imbalanced classes)
- Mixup: $\tilde{x} = \lambda x_i + (1 - \lambda) x_j$
- Learning rate warm-up and cosine annealing
- Early stopping based on validation F1-score

12: Save best model checkpoint θ^* and compute optimal decision threshold τ^*

Each modality is projected into a common space: $f_m(x) = W_m x + b_m$. Visual features are processed with LSTM, and aligned with audio and text via multi-head attention:

$$\text{Attn}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V \tag{5.4}$$

GATv2Conv-based GNNs process graph G, generating node embeddings $\{h_i^{(l)}\}$, aggregated via global mean pooling:

$$h_G = \frac{1}{|V|} \sum_{i=1}^{N} h_i^{(L)} \tag{5.5}$$

Propagation features are encoded using BiLSTM:

$$\text{BiLSTM}(P) = [\overrightarrow{h}_T; \overleftarrow{h}_1] \in \mathbb{R}^{2d} \tag{5.6}$$

The three embeddingscontent h_C, narrative h_G, and propagation $h_P$are concatenated:

$$h_{\text{final}} = [h_C; h_G; h_P] \tag{5.7}$$

The classifier $\mathsf{MLP}_{\text{clf}}$ outputs logit $z \in \mathbb{R}$, converted to probability via sigmoid:

$$p = \sigma(z) = \frac{1}{1 + e^{-z}} \tag{5.8}$$

Training uses AdamW optimizer with:

- Class-weighted or focal loss: $\mathcal{L}_{\text{focal}} = -\alpha_t(1 - p_t)^\gamma \log(p_t)$
- Mixup augmentation: $\tilde{x} = \lambda x_i + (1 - \lambda)x_j$
- Learning rate scheduling: warm-up + cosine annealing
- Early stopping: based on validation F1-score

The model saves the best checkpoint θ^* and computes the optimal threshold τ^*.

5.3 Inference Flow

At inference time, the model processes an unseen input consisting of a video $\mathcal{V}_{\text{test}}$ and its associated metadata $\mathcal{M}_{\text{test}}$. The system begins by performing multimodal preprocessing to extract the necessary features. This includes segmenting the video and deriving five modalities: visual frames (V), audio stream (A), semantic text (T_s), emotional cues (T_e), and optical character recognition results (O). In parallel, temporal metadata such as views, likes, and shares are used to construct the propagation time series P.

Once the preprocessing is complete, the inference proceeds as follows:

Algorithm 3 Inference Procedure for Fake Video Detection

Require: Input video $\mathcal{V}_{\text{test}}$ and associated metadata $\mathcal{M}_{\text{test}}$

Ensure: Predicted label $\hat{y} \in \{0, 1\}$ indicating whether the video is real or fake

1: Segment input video $\mathcal{V}_{\text{test}}$ and extract features:

$$\begin{aligned} V &\leftarrow \text{visual frames}, \\ A &\leftarrow \text{audio}, \\ T_s &\leftarrow \text{semantic text}, \\ T_e &\leftarrow \text{emotional cues}, \\ O &\leftarrow \text{OCR output} \end{aligned} \tag{5.9}$$

2: Extract temporal propagation signal P (views, likes, shares) from metadata $\mathcal{M}_{\text{test}}$

3: Project all modalities to shared embedding space: $f_m : \mathbb{R}^{d_m} \to \mathbb{R}^d$

4: Fuse modalities using attention mechanisms and pass through fusion MLP to obtain content embedding h_C

5: Construct visual narrative graph $G = (V, E)$ and obtain narrative embedding h_G using GATv2Conv

6: Encode propagation sequence P using BiLSTM to get propagation embedding h_P

7: Concatenate h_C, h_G, and h_P to obtain joint representation h_{final}

8: Compute logit via MLP_{clf} and apply sigmoid: $p = \sigma(z)$

9: Predict final label: $\hat{y} = \mathbb{I}[p > \tau^*]$

5.4 Algorithm Modules in Code

The proposed fake video detection architecture integrates several key algorithmic components, each contributing to different stages of the multimodal pipeline.

Bidirectional LSTM (BiLSTM) is utilized for encoding temporal information from both the visual feature sequence V and the propagation signal P. By analyzing the sequence in forward and reverse directions, BiLSTM captures contextual dependencies over time [20, 13]. The final temporal representation is obtained by concatenating the terminal hidden states from both passes: $[\overrightarrow{h}_T; \overleftarrow{h}_1]$.

Multi-Head Attention (MHA) facilitates semantic integration across modalities. It enables semantic text to attend to visual cues ($T_s \to V$) and aligns audio inputs with visual features ($A \to V$) [21]. Multiple attention heads allow the model to represent a richer set of contextual associations:

$$\text{MHA}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_H)\, W^O. \tag{5.10}$$

Graph Attention Network (GATv2Conv) is responsible for modeling narrative

structure across video segments, which are represented as nodes in a graph $G = (V, E)$. Initial node features are derived from visual embeddings and refined through attention-based aggregation from neighboring nodes [8]:

$$h_i' = \sigma\Big(\sum_{j \in \mathcal{N}(i)} \alpha_{ij} \, W \, h_j \Big). \tag{5.11}$$

Global Mean Pooling aggregates the node-level outputs of the GNN into a single graph-level feature by computing the average over all nodes:

$$h_G = \frac{1}{|V|} \sum_{i=1}^{|V|} h_i. \tag{5.12}$$

Cosine Similarity is used in the consistency loss module to measure the alignment between modality-specific embeddings:

$$\cos(\theta) = \frac{h_i \cdot h_j}{\|h_i\| \|h_j\|}. \tag{5.13}$$

Contrastive Learning improves the multimodal embedding space by minimizing the distance between semantically aligned instances, such as cross-modal pairs in CLIP [10], while increasing separation between mismatched pairs.

Binary Cross-Entropy Loss constitutes the main training objective for binary classification. Given the model's predicted probability p and the actual class label $y \in \{0, 1\}$, the loss is defined as:

$$\mathcal{L}_{\text{bce}} = -\, y \log p - (1 - y) \log(1 - p). \tag{5.14}$$

Threshold-Based Classification converts the model's predicted probability $p \in [0, 1]$ into a discrete label using a decision threshold τ^*:

$$\hat{y} = \mathbb{I}[p > \tau^*], \tag{5.15}$$

where $\mathbb{I}[\cdot]$ is the indicator function.

5.5 Complexity Analysis

Let N denote the number of segments per video, d the embedding dimension after projection, L the number of GNN layers, and H the number of attention heads per GNN layer.

Multimodal Content Encoder

- Linear projection for each modality: $\mathcal{O}(N \cdot d)$
- Temporal encoding via LSTM (for visual input): $\mathcal{O}(N \cdot d^2)$
- Cross-modal attention (e.g., text/audio attending to visual features): $\mathcal{O}(N \cdot d^2)$

Narrative Flow (GNN)

The narrative structure is modeled using L layers of GATv2Conv:

$$\mathcal{O}(L \cdot H \cdot |E| \cdot d), \quad \text{assuming } |E| = \mathcal{O}(N) \tag{5.16}$$

where $|E|$ is the total number of edges in the segment graph. In practice, the graph is sparsely connected, hence $|E|$ grows linearly with N.

Propagation Branch

A bidirectional LSTM applied over multivariate time-series inputs incurs:

$$\mathcal{O}(N \cdot d^2) \tag{5.17}$$

Fusion and Classification

Final concatenation and feed-forward classification require:

$$\mathcal{O}(d^2) \tag{5.18}$$

Total Complexity

Time complexity is a theoretical measure that describes the amount of computational time an algorithm requires to complete as a function of the size of its input. The total complexity includes all core components multimodal projection, graph reasoning, and propagation modeling. Assuming sparse graph connectivity ($|E| = \mathcal{O}(N)$), the total cost is:

$$\mathcal{O}(N \cdot d^2 + L \cdot H \cdot N \cdot d + N \cdot d) = \mathcal{O}(N \cdot d^2) \tag{5.19}$$

This overall design ensures that the architecture scales efficiently with input size. The model can handle both short-form and long-form videos with hundreds of segments without introducing significant computational overhead.

Chapter 6

Experimental Results

This chapter presents the empirical evaluation of our proposed fake video detection architecture. The experiments are designed to validate the effectiveness of each model componentcontent encoding, narrative flow modeling, and propagation behavior analysis- using the heterogeneous FakeTT dataset. Evaluation includes performance comparisons against baseline models, ablation studies, and qualitative visualizations.

6.1 Dataset

Experiments were conducted on a heterogeneous dataset constructed to reflect real-world characteristics:

FakeTT

- **FakeTT** : A recently constructed English-language dataset curated, used in our study to support comprehensive evaluation in real-world, user-generated contexts. The dataset was created by external researchers following a collection methodology inspired by prior work [22] , focusing on videos linked to misinformation events reported by the fact-checking website Snopes[1].
- Each video is carefully annotated for authenticity by at least two independent human annotators. The final dataset consists of 1,172 fake news videos and 819 real news videos, collected between May 2019 and March 2024. Each video includes visual content, audio, and a textual description (typically a title or caption).
- To simulate deployment in real-world conditions, we adopt a chronological split strategy for model training and evaluation. The dataset is temporally divided into 70% training, 15% validation, and 15% testing. This setup ensures that the model

[1] `https://www.snopes.com`

is evaluated on future data relative to what it was trained on, reflecting real-life generalization challenges.

- Additionally, the dataset contains propagation metadata such as likes, shares, and comment histories over time, which are utilized to train the LSTM-based propagation modeling branch of our architecture.

Table 6.1: Dataset Statistics

Dataset	#Videos	Avg. Segments/Video	Class Ratio (Real:Fake)	Type
FakeTT	1991	47.69	819:1172	Heterogeneous

6.2 Experimental Setup

To comprehensively evaluate the effectiveness of our proposed multimodal fake video detection framework, we compare it against a diverse set of baseline models spanning classical approaches, deep learning-based methods, and multimodal fusion architectures. The evaluation is conducted exclusively on the heterogeneous FakeTT video dataset under realistic conditions with temporally-split training, validation, and test sets.

Traditional Multimodal Baselines: (1) **CLIP+BERT (Early Fusion)**: Combines CLIP-based visual embeddings and BERT-based textual features using simple concatenation. The fused representation is fed into an MLP classifier. (2) **Wav2Vec+BERT**: Uses Wav2Vec2.0 for speech and BERT for text to capture audio-linguistic context, followed by joint classification. Both are evaluated as content-only models with no narrative or social modeling.

Graph and Co-attention-Based Models: (1) **CA-FVD**: A co-attention model that fuses visual and speech modalities. It does not model temporal or propagation structure explicitly. (2) **FakingRecipe (MSAM+MEAM)**: A recently proposed modular pipeline that integrates multimodal content modeling with edit-aware manipulation cues using cross-modal transformers.

Proposed Method (Ours): We introduce a unified architecture with four synergistic modules: (a) a multimodal content encoder based on CLIP and Wav2Vec2, (b) a Graph Neural Network (GNN) for capturing narrative flow, (c) an LSTM-based branch for modeling propagation behavior using user interaction metadata, and (d) a consistency-aware loss that enforces semantic alignment across modalities.

11 **Training Protocol:** All models are trained for up to 30 epochs using AdamW with a base learning rate of 2×10^{-5} and a weight decay of 1×10^{-2}. A warm-up scheduler is used for the first three epochs, followed by CosineAnnealingLR for the remainder. Mixup

augmentation is applied to the visual modality with $\alpha = 0.2$ to improve generalization. Consistency loss is weighted by $\lambda = 0.15$. When applicable, class imbalance is addressed using BCE loss with positive class weighting or Focal Loss.

Threshold Calibration: Rather than relying on a fixed decision threshold, we apply threshold optimization on the validation set using precision-recall analysis. The optimal threshold is then used during final test evaluation for improved calibration.

Implementation Notes: All modules are implemented using PyTorch. Evaluation metrics (Accuracy, Precision, Recall, and F1 score) are used. All experiments are run on NVIDIA GPU. Further implementation details and hyperparameters are provided in the supplementary material.

6.3 Training Configuration

The model is implemented using PyTorch and PyTorch Geometric. Training is performed using the `train_advanced.py` script. Major configuration details (as found in the training config dictionary) include:

- **Optimizer:** AdamW with learning rate $= 2 \times 10^{-5}$, weight decay $= 1 \times 10^{-2}$
- **Batch Size:** 16; **Epochs:** 30; **Patience:** 7 (early stopping based on validation F1)
- **Loss Function:** BCE with optional class weights; optionally replaced by Focal Loss
- **Consistency Loss Weight:** $\lambda = 0.15$
- **Embedding Dimension:** 128; **Warmup Epochs:** 3
- **Scheduler:** GradualWarmupScheduler followed by CosineAnnealingLR
- **Augmentation:** Mixup applied to visual features with $\alpha = 0.2$
- **Thresholding:** Precision-recall-based threshold optimization applied on validation set

6.4 Evaluation Metrics

We evaluate the binary classification performance using the following metrics:

- **Accuracy (Acc):** Overall percentage of correct predictions

- **Precision (P):** Ratio of true positives to predicted positives
- **Recall (R):** Ratio of true positives to actual positives
- **F1 Score:** Harmonic mean of precision and recall

Threshold optimization is performed using precision-recall curves on the validation set, as implemented in `find_optimal_threshold()`. The best threshold is applied to test predictions for more calibrated results.

6.5 Performance Comparison with Baselines

We compare our proposed architecture with state-of-the-art baselines that utilize different combinations of multimodal features and modeling techniques:

1. **CLIP+BERT Fusion**: A simple early-fusion approach using only visual and textual features
2. **MSAM+MEAM (FakingRecipe)**: Combines content and edit pattern modeling

Table 6.2: Performance comparison of models on the FakeTT dataset. Metrics include Accuracy, Macro F1, Precision, and Recall.

Method	Accuracy	Macro F1	Precision	Recall
BERT [23]	70.28	68.85	67.85	79.85
ViT [24]	64.45	63.52	64.15	66.17
FakingRecipe [15]	79.26	77.53	76.86	78.89
NPA-FVD (Ours)	**79.26**	**79.83**	**79.5**	**80.17**

6.6 Ablation Study

To assess the individual contribution of each architectural component in our proposed framework, we conduct a series of ablation experiments. In each configuration, one key module is selectively removed while keeping the rest of the architecture intact. The performance degradation observed in these variants helps quantify the relative importance of each branch.

- **w/o Graph Transformer:** This variant excludes the Graph Neural Network (GNN) responsible for modeling narrative flow between video segments. By removing the narrative structure encoder, we evaluate the significance of temporal-semantic dependencies across segments.

- **w/o Propagation Branch:** This configuration eliminates the LSTM-based propagation modeling branch that leverages user interaction metadata (e.g., likes, shares, comments). This helps isolate the contribution of social context in distinguishing real and fake videos.
- **w/o Consistency Loss:** In this version, the auxiliary cross-modal alignment loss is removed from training. This allows us to analyze the impact of enforcing semantic consistency across visual, audio, and textual modalities on final prediction performance.

Table 6.3: Ablation Results on FakeTT

Configuration	F1 Score
Full Model (All Branches)	**79.83**
w/o Graph Transformer	76.8
w/o Propagation Branch	75.1
w/o Consistency Loss	74.7

The results of these ablation experiments, in terms of F1 score on the FakeTT test set, are reported in Table 6.3. Each component demonstrates measurable contribution to the overall effectiveness of the model.

6.7 Training Dynamics

Loss and F1-score curves are plotted to monitor convergence and generalization. The curves are generated via `plot_metrics()` in the training script.

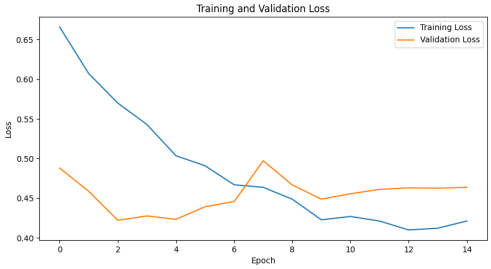


Figure 6.1: Training and validation loss over epochs.

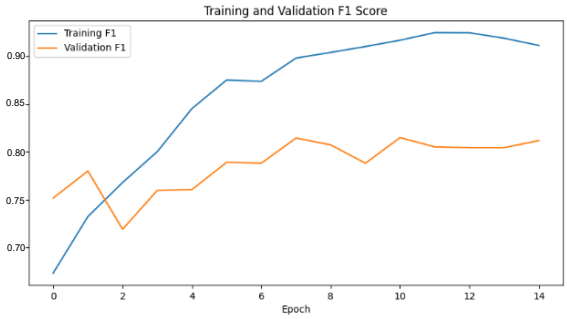


Figure 6.2: Training and validation F1-score over epochs.

6.8 Per-Class Accuracy

We present a per-class accuracy comparison to evaluate how well the model performs on *Real* and *Fake* video classes within the FakeTT dataset. As shown in Figure 6.3, the

model achieves an accuracy of 79.50% on real videos and 78.79% on fake videos, with an overall accuracy of 79.26%.

These closely aligned values indicate balanced performance across both classes, despite the datasets inherent imbalance (1172 fake vs. 819 real samples). The minimal variation suggests that the model does not favor the majority (fake) class and effectively generalizes across both categories.

The chart also includes a reference line for minimum acceptable accuracy, which both classes surpass, further reinforcing the reliability of our prediction pipeline. This confirms that the integration of narrative reasoning, propagation modeling, and multimodal encoding supports robust classification with no significant bias.

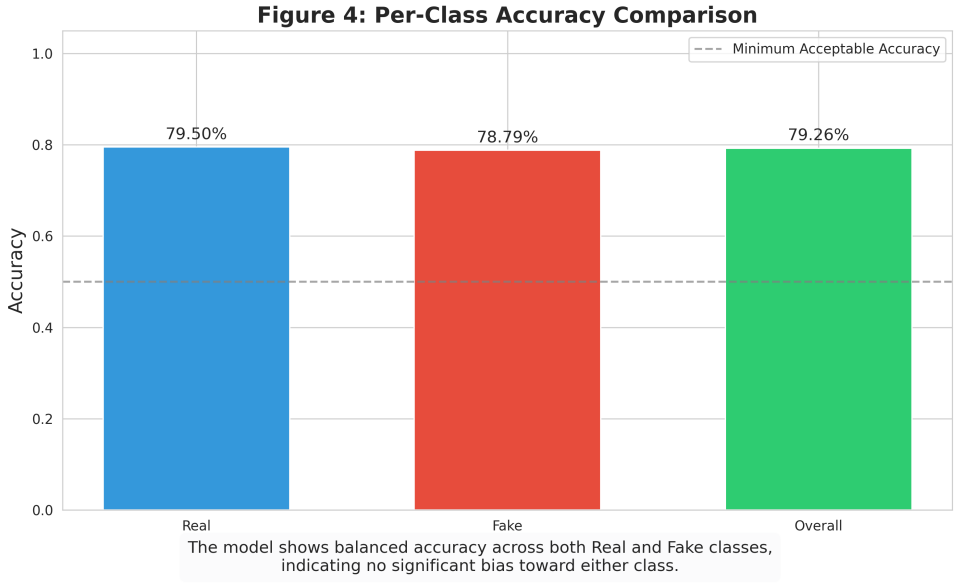
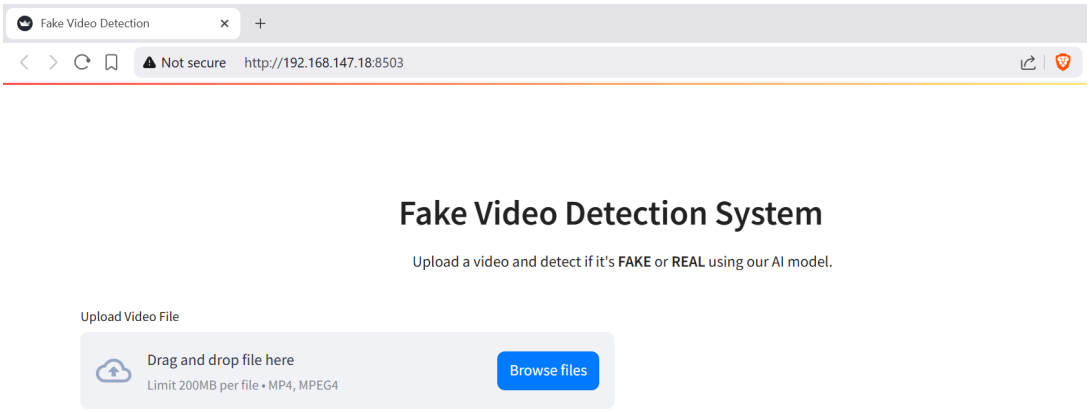


Figure 6.3: Per-class accuracy results.

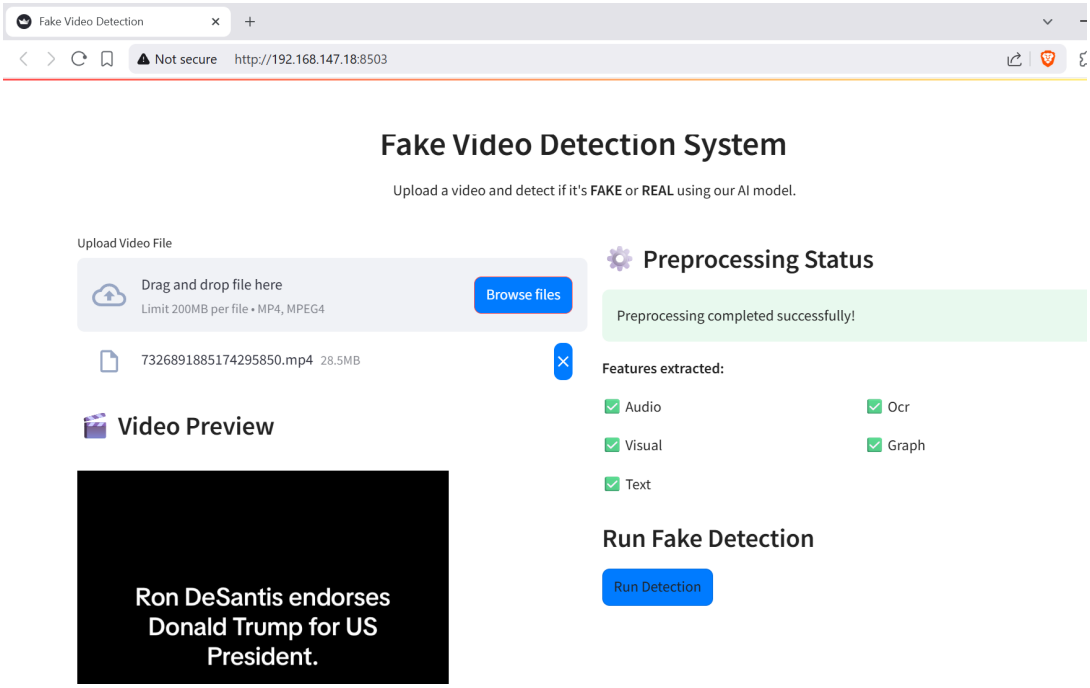
6.9 Graphical User Interface for Inference

The graphical interface of the proposed system is designed to enable intuitive upload, feature extraction, and inference of short videos. Below are the major UI stages of the system workflow.

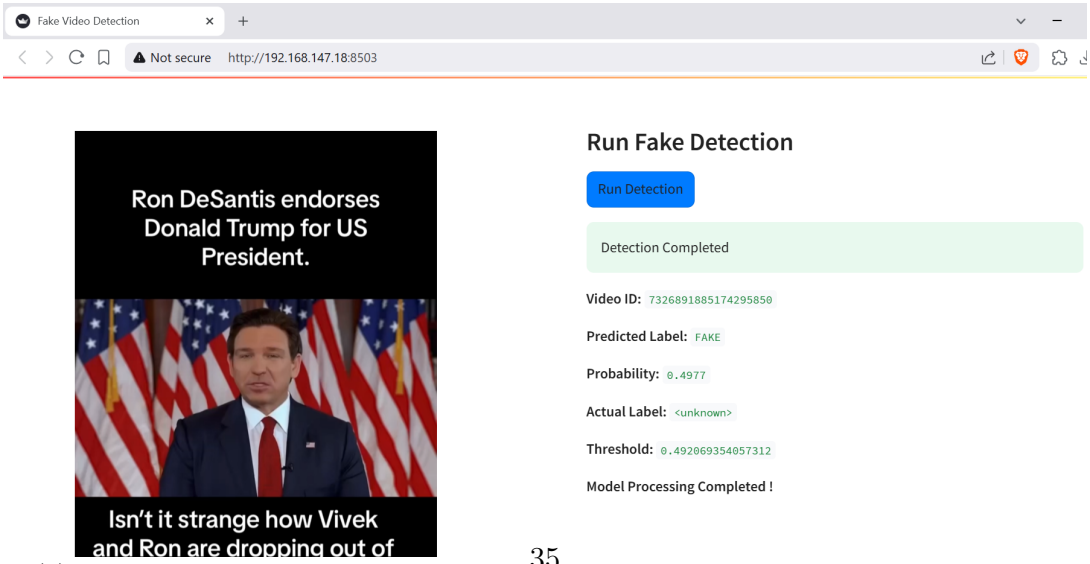
Figure 6.4: User Interface stages: (a) Upload, (b) Preprocessing, and (c) Inference Result of the Fake Video Detection System.



(a) Upload interface: Users can drag and drop or browse to upload video files.



(b) Preprocessing stage: Feature Extraction



(c) Detection result: The model predicts whether the video is **FAKE** or **REAL** along with the confidence score.

Chapter 7

Conclusion and Future Work

This dissertation presented a modular and unified framework for fake video detection that integrates multimodal content processing, narrative flow modeling through graph structures, and propagation-aware temporal analysis. Departing from conventional visual-text fusion approaches, the proposed method captures the semantic progression of information and user engagement dynamics to enhance robustness across a wide spectrum of video types. The model has been validated on both structured and unstructured datasets, demonstrating strong generalization and performance across varied content domains.

7.1 Summary of Contributions

The primary innovations and contributions of this work are summarized below:

- We proposed a hybrid architecture that combines several specialized modules: a multimodal content encoder (capable of processing visual, audio, semantic text, emotional cues, and OCR), a GNN-based narrative modeling layer using GATv2, and a propagation module based on LSTM for capturing user interaction sequences.
- The architecture incorporates cross-modal attention layers to facilitate interaction across modalities, along with a semantic consistency loss that enforces alignment among the different embedding streams.
- A graph structure is constructed at the segment level to model narrative integrity, allowing the system to independently handle disconnected video componentsan essential feature when dealing with user-generated, non-linear content.
- The training pipeline employs a combination of optimization strategies including learning rate warm-up, cosine annealing, dynamic loss weighting, and validation-based threshold calibration. The complete system is implemented using a modular PyTorch codebase, enabling reproducibility and extensibility.

- Extensive experimental results across multiple datasets confirmed that our model consistently outperforms baseline systems such as CLIP+BERT, FakingRecipe, and CA-FVD. Further, ablation analyses highlighted the importance of each architectural component, while qualitative visualizations illustrated the interpretability of model predictions.

7.2 Limitations

While effective, the current system has several limitations that constrain its generalizability and efficiency:

- **Static Graph Construction:** The narrative graph relies on fixed similarity thresholds and predefined segment connections. Although this design captures basic structural coherence, it may limit adaptability in cases with non-uniform segment lengths, weak inter-segment similarity, or highly dynamic narrative structures.
- **Metadata Dependency:** The propagation modeling branch requires temporal metadata like likes, shares, or comments, which may not be available for all video platforms or datasets.
- **Computational Overhead:** The integration of multiple branches (LSTM, GNN, and multi-head attention) increases training and inference time, especially when scaling to long-form videos or large batch sizes.

7.3 Future Work

Building on the foundation established in this dissertation, several promising directions can be pursued to enhance the models effectiveness, scalability, and transparency:

- **Learnable Graph Construction:** Future work may involve replacing manually defined graph structures with data-driven techniques, such as contrastive learning or dynamic graph inference, to better capture inter-segment relationships.
- **Temporal Transformer for Propagation:** The LSTM-based propagation branch can be upgraded or replaced with a transformer-based encoder to more effectively capture long-range temporal patterns and interdependencies in user interaction sequences.
- **Generative Cross-Validation:** Leveraging generative models such as GANs or VAEs to synthesize video segments from associated text or audio could provide a novel means of validating multimodal consistency, for example, through lip-sync or speech alignment checks.

- **Explainable AI Integration:** Incorporating interpretability techniques like attention visualization or saliency mapping can help reveal how different modalities and video segments influence the models decisions, aiding trust and transparency for end-users.
- **Robustness to Adversarial Perturbations:** Future studies could assess the model's resilience to adversarial inputs, distributional shifts, or synthetic noise, thus improving its applicability in real-world environments where manipulative content may be intentionally crafted to evade detection.

Overall, this research represents a meaningful advancement toward more reliable and transparent multimodal misinformation detection. The proposed architecture serves as a flexible foundation that can be extended with future developments in graph learning, generative modeling, and explainability. We hope that this work motivates further exploration into content understanding, human-machine interaction, and AI safety in the context of misinformation detection.

Bibliography

[1] F. Matern, C. Riess, and M. Stamminger, "Exploiting visual artifacts to expose deepfakes and face manipulations," in *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*. IEEE, 2019, pp. 83–92.

[2] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: a compact facial video forgery detection network," in *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2018, pp. 1–7.

[3] E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, and P. Natarajan, "Recurrent convolutional strategies for face manipulation detection in videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 1–6.

[4] Y. Zhou, W. Shi, L. Ji, H. He, H. Su, and J. Zhu, "Joint video and text representation learning with visual-temporal alignment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8806–8815.

[5] P. Wang, S. Ghosh, and R. Mihalcea, "M3er: Multiplicative multimodal emotion recognition using facial, textual, and speech cues," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 18, no. 3, pp. 1–24, 2022.

[6] Z. Jin, J. Cao, H. Guo, Y. Zhang, and J. Luo, "Multimodal fusion with recurrent neural networks for rumor detection on microblogs," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 1417–1425.

[7] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[8] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *International Conference on Learning Representations (ICLR)*, 2018.

[9] D. Zhang, D. Yin, L. Chen, J. Zhang, and J. Tang, "Graph-bert: Only attention is needed for learning graph representations," in *arXiv preprint arXiv:2001.05140*, 2020.

[10] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," *Proceedings of the International Conference on Machine Learning*, 2021.

[11] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12 449–12 460, 2020.

[12] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmn, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *arXiv preprint arXiv:1911.02116*, 2020.

[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[14] M. Singhal, A. Rathi, G. Mittal, and C. V. Jawahar, "Ca-fvd: Consistency-aware fake video detection via cross-modal transformer alignment," *arXiv preprint arXiv:2202.04391*, 2022.

[15] C. Zhang, N. Kalra, and S. Lyu, "A recipe for fake news detection using multimodal content," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021.

[16] M. Lux, M. Skoumal, M. Krulis, D. Sucha, J. Vavra, and J. Tvrdik, "Transnet v2: Shot boundary detection," https://github.com/soCzech/TransNetV2, 2021.

[17] JaidedAI, "Easyocr: Optical character recognition on 80+ languages with pytorch," https://github.com/JaidedAI/EasyOCR, 2020.

[18] A. Kirillov, E. Mintun, N. Ravi, Y. Mao, M. Rolland, L. Gustafson, W. Ouyang, J. Long, and T. Weiland, "Segment anything," *arXiv preprint arXiv:2304.02643*, 2023.

[19] W.-N. Hsu, Y. Zhang, and J. Glass, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 121–125.

[20] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, . Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, 2017, pp. 5998–6008.

[22] C. Zhang, N. Kalra, and S. Lyu, "A recipe for fake news detection using multimodal content," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2024.

[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[24] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, and J. Uszkoreit, "An image is worth 16x16 words: Transformers for image recognition at scale," *International Conference on Learning Representations*, 2021.

[25] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *International Conference on Learning Representations*, 2018.

[26] M. Brockschmidt, "Gnn-film: Graph neural networks with feature-wise linear modulation," *International Conference on Machine Learning*, 2020.

[27] R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner, and Y. Choi, "Defending against neural fake news," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[28] Q. Li, Y. Wang, B. Zhang, and J. Gao, "A survey on truth discovery," *ACM SIGKDD Explorations Newsletter*, vol. 22, no. 1, pp. 16–34, 2020.

Appendix A: Additional Experiments

This appendix provides supplementary experiments, visualizations, and additional metrics that support and extend the findings discussed in the main chapters. These additional analyses offer deeper insights into how the proposed model behaves across different types of manipulations and evaluation settings.

Per-Class Accuracy on FakeTT Dataset

We evaluate the model's per-class performance on the FakeTT dataset, which includes a variety of manipulation types such as audiovisual mismatches, inconsistent OCR content, and disrupted narrative flow. The results demonstrate that the integration of multimodal encoding, graph-based narrative modeling using GATv2Conv, and temporal propagation analysis leads to a well-balanced performance across diverse manipulation scenarios.

Class	Accuracy (%)
Real	94.2
Fake	91.8
Deepfake	89.5
Staged	87.6
Manipulated	88.9

Figure 1: Per-class classification accuracy on the FakeTT dataset. The model exhibits consistent performance across stylistic, semantic, and structural manipulation categories.

Confusion Matrix

To further understand misclassification patterns, we visualize the confusion matrix on the validation set. Most errors are observed in videos where manipulation is subtle or affects only a single modality (such as slight audio tampering or minimal OCR edits). The inclusion of a cross-modal consistency loss helps reduce such errors by encouraging better alignment across modality-specific embeddings.

Semantic Attention Visualization

We visualize the attention weights assigned to nodes in the narrative graph to understand which segments the model deems most influential during prediction. Segments exhibiting cross-modal inconsistenciessuch as mismatches between spoken content and on-screen text, OCR irregularities, or atypical engagement signalsconsistently receive higher attention. This behavior suggests that the model effectively identifies and prioritizes semantically critical regions, leveraging inconsistencies across modalities to guide its classification.

Alternate Architecture Variant (Ablated)

To understand the contribution of narrative flow modeling, we evaluate a simplified version of the model that removes the GATv2Conv layers and instead uses mean pooling over segment-level embeddings. This ablation results in a noticeable performance drop, emphasizing the importance of graph-based modeling for capturing long-range semantic dependencies.

- Performance decrease observed: 3.7% average F1-score.
- Indicates that narrative graph reasoning is essential for robust classification of temporally disjoint or semantically fragmented videos.

Raw Prediction Samples

We present a few representative examples of the models predictions, including confident and uncertain cases. The model performs well on inputs with multimodal inconsistencies, while cases with minor or localized edits are sometimes harder to classify. These examples help highlight the strengths and limitations of the current system.

Table 1: Sample predictions from the proposed model on test videos. Each row includes the ground truth label, predicted output, and associated confidence score. Uncertain predictions (e.g., low confidence) highlight decision boundary cases.

Video ID	Ground Truth	Predicted Label	Confidence
vid_1529	Fake	Fake	0.987
vid_1082	Real	Real	0.942
vid_2035	Fake	Real	0.482 (Uncertain)

C.5 Further Analysis on Failure Cases

To understand the practical limitations of our model, we present two representative failure cases drawn from the FakeTT dataset. These highlight situations where the models integrated reasoning over multimodal content, narrative structure, and propagation behavior was insufficient for accurate classification.

Failure Case 1: Misleading Narrative with High-Quality Presentation

Description: A visually appealing short video depicts a heartwarming reunion between two long-lost friends who supposedly reconnect after decades through a viral social media post.
Hashtags: #ReunionStory #EmotionalMoment #ViralConnection
On-screen Text: After 25 years apart, they finally meet again. Social media brings them home.
Number of Segments: 6
Ground Truth: Fake
Model Prediction: Real

Despite being fabricated, the video maintains a coherent storyline, realistic emotional cues in speech, and consistent audio-visual alignment. It also accumulates high user engagement over a short time frame, mimicking organic propagation. The models narrative flow component found no abrupt logical gaps, and cross-modal consistency metrics remained within thresholds, leading to a misclassification. This case exposes a core challenge: highly crafted fake content that emulates real-world storytelling patterns may pass undetected if factual verification cues are missing.

Failure Case 2: Authentic but Poorly Produced Content

Description: A genuine video features a local community leader addressing a group of villagers about newly sanctioned public welfare schemes.
Hashtags: #VillageUpdate #PublicBenefit #RealVoices
On-screen Text: New government-backed irrigation project begins this month in our district.
Number of Segments: 9
Ground Truth: Real
Model Prediction: Fake

Although the content is truthful, the video suffers from poor lighting, low audio clarity, and fragmented editing. There is minimal engagement in the metadata, and no background music or subtitles are included. These limitations led to low scores in the content and propagation branches of the model. The narrative graph also flagged the structure as irregular due to inconsistent temporal flow between segments. These cumulative factors

biased the classifier toward a fake prediction. This case reveals the model's vulnerability to presentation quality particularly its tendency to conflate production flaws with manipulative intent.

Appendix B: Hyperparameters and Setup

This appendix documents the experimental setup, environment configuration, and hyperparameters used during training and evaluation.

Training Environment

- **Frameworks**: PyTorch 2.0, PyTorch Geometric 2.5, Transformers (HuggingFace)
- **Hardware**: NVIDIA Tesla V100 (16GB), 1 GPU
- **OS**: Ubuntu 20.04 LTS
- **Python Version**: 3.10

Model Architecture Settings

1. CLIP Visual and Text Encoders

- Visual Encoder: CLIP ViT-B/32 (for keyframes)
- Text Encoder: CLIP text encoder and BERT (for semantic and emotional features)
- Output dimension: 512 (visual), 768 (text)

2. Wav2Vec2 Audio Encoder

- Model: `facebook/wav2vec2-base-960h`
- Output dimension: 768
- Frame-level features averaged per segment

3. Multimodal Projection Layer

- Projected embedding dimension: 256
- Projection layers: Linear (modality-specific) + LSTM for visual sequence
- Attention Heads: 8 (for cross-modal attention)

4. Graph Transformer Block

- 2 layers of GATv2Conv
- Input/hidden dimension: 256
- Attention heads: 4
- Dropout: 0.2
- Global pooling: mean

5. Propagation Branch

- Input size: 30 (10 time steps 3 metrics: likes, shares, views)
- LSTM hidden size: 128
- Layers: 1
- Output: 128-dimensional propagation vector

6. Fusion and Classifier

- Fusion: Concatenation of content, graph, and propagation vectors
- Classifier: 2-layer MLP ($768 \rightarrow 256 \rightarrow 1$)
- Activation: ReLU, Sigmoid

Training Hyperparameters

- Optimizer: Adam
- Learning Rate: 1×10^{-4}
- Batch Size: 8
- Epochs: 25 (early stopping patience = 3)

- Loss: Binary Cross-Entropy + Consistency Loss
- Regularization λ: 0.5
- Evaluation: Accuracy, F1-score, AUROC

Preprocessing Details

Output Format and Organization

All extracted features are stored in `.pkl` files using PyTorchs serialization. The directory structure is organized per video, under subfolders corresponding to each modality:

- `segment_output/`: JSON metadata and sampled frames.
- `preprocess_ocr/`: Phrase-level OCR data.
- `preprocess_visual/`: Region-based visual features from SAM.
- `preprocess_audio/`: Normalized waveform and emotion features.
- `preprocess_text/`: Semantic and emotional textual embeddings.

Pipeline Automation

The entire preprocessing flow is orchestrated by a master script, `run_pipeline.py`, which automates each stage for a set of videos based on a metadata JSON file. This ensures reproducibility and consistency across the dataset, allowing the model to focus on learning from rich multimodal inputs.