Democratic and Popular Algerian Republic

الجمهورية الجزائرية الديمقراطية الشعبية

Ministry of Higher Education and Scientific Research

وزارة التعليم العالي والبحث العلمي

Yahia Fares University of Medea

Faculty of Sciences

Department of Mathematics and Computer Science



Final Master Project

Submitted for Obtaining The Master's Degree in
Computer Science

Speciality: Applied Artificial Intelligence

Multimodal Fake News Detection Using Transformer Architectures and Large Language Models (LLMs)

Presented by:

Mr. Lotfi Hallas

Mr. Mohamed Fouad Cherfi

Proposed and Supervised by:

Dr. Ahmed Cherif Mazari

Ms. Amina Mzaad

2025/2026

شكر وعرفان

الحمد لله رب العالمين، الذي أنعم علينا بنعمة العقل والعلم، وأعاننا بفضله على إتمام هذا العمل العلمي.

نتقدم بخالص الشكر والتقدير وعظيم الامتنان إلى أستاذينا الفاضلين والمشرفين على هذا البحث، **الدكتور أحمد الشريف مزاري** و **الاستاذة امينة مزعد** ، لما قدماه لنا من توجيهات سديدة ونصائح قيّمة. لقد كان لصبرهما وسعة صدرهما، ولجهودهما في المتابعة والمناقشة العلمية، أبلغ الأثر في صياغة هذا العمل وإخراجه إلى النور، فلهما منا كل التقدير والاحترام.

كما نعرب عن امتناننا لكافة أساتذتنا في قسم الإعلام الآلي بجامعتنا، الذين لم يبخلوا علينا بعلمهم وتوجيهاتهم، وكانوا لنا خير قدوة في مسيرتنا الجامعية.

وأخيراً، نخص بالشكر والتقدير عائلتي الكريمة التي كانت السند والداعم الأول، والتي غمرتنا بمحبتها وصبرها ودعواتها الصادقة طوال فترة إنجاز هذا العمل.

شكراً لكل من كان سبباً في إتمام هذا الإنجاز.

الإهداء

بسم الله الرحمن الرحيم

الحمد لله الذي بنعمته تتم الصالحات، وبتوفيقه تُدرك الغايات.

أهدي ثمرة هذا العمل المتواضع

إلى من كان دعاؤهما سر نجاحي ونور دربي

أمي الغالية وأبي العزيز

حفظهما الله وأطال في عمرهما.

إلى **إخوتي** سندي وقوتي، وإلى كل **عائلتي** الكريمة.

إلى كل من علّمني حرفاً وأنار لي طريقاً، إلى **أساتذتي** الأفاضل.

إلى **أصدقائي** ورفقاء دربي، الذين كانوا خير سند في مسيرتي.

إلى كل من ساهم من قريب أو بعيد في إنجاز هذا العمل.

لطفي حلاس

إهـــــداء

بسم الله الرحمن الرحيم

إلى ملاذي الآمن ومصدر قوتي، ونبض قلبها دليلي في الحياة…
إلى من غمرتني بدعواتها الصادقة في جوف الليل فأنارت دربي…
أمي الحبيبة

إلى من أحمل اسمه بكل فخر، وكان لي السند والقدوة في مواجهة الصعاب…
إلى من علمني أن الطموح لا سقف له، وأن النجاح يُصنع بالصبر…
أبي الغالي
(حفظهما الله وأدام عليهما الصحة والعافية وتاج الوقار)

إلى من تقاسمت معهم الأيام والذكريات، وكانوا لي العضد والملجأ الآمن…
إخوتي وأخواتي

إلى عائلتي الكريمة وكل من يربطني بهم دم وانتماء.

إلى طيفٌ يسكن الغد، ونجمةٌ يخبئها لي القدر بين طيات الأيام…
إلى من سأقاسمها تفاصيل العمر، وتكون لي الملاذ والنبض في قادم الأيام…
شريكة الحلم الآتي

إلى ينابيع العلم والمعرفة، الذين لم يبخلوا علينا بتوجيه أو نصح…
أساتذتي الأفاضل

إلى الأوفياء الذين شاركوني رحلة السهر والجهد، وكانوا خير رفقاء في طريق التميز…
أصدقائي الأعزاء

إلى كل من مرّ في حياتي فترك أثراً طيّباً، أو علّمني درساً جميلاً.

محمد فؤاد شرفي

Abstract

The proliferation of false information in digital media has evolved to include manipulated images, synthetic audio, and deepfake videos. Recent advances in generative AI have made robust multimodal detection systems an urgent necessity. This thesis proposes two novel multimodal architectures for fake news detection. The first architecture targets image–text content by extracting visual (SigLIP), textual (Longformer), frequency (DCT/FFT), and Large Language Model (LLM)-based reasoning features (Llama-3), which are fused via a Mixture-of-Experts (MoE) mechanism to produce both veracity predictions and natural language explanations. The second architecture analyzes short videos by extracting eleven feature sets across the visual, acoustic, and textual domains, utilizing cross-modal attention to combine keyframe (SigLIP), audio (Wav2Vec2), text (RoBERTa), and consistency (Phi-3 Mini) representations. Evaluated on the Fakeddit dataset, the image-text system achieves 85.03% accuracy, with visual features providing the most critical detection signal. The video system achieves 91.30% accuracy on the FakeTT dataset, where facial and temporal features prove essential. Ultimately, this work demonstrates that integrating multiple modalities and explicit LLM reasoning is essential for creating reliable, transparent, and interpretable defense mechanisms against modern digital deception.

Keywords: Fake news detection, multimodal learning, deep learning, transformers, large language models, deepfake detection, cross-modal attention, explainability.

Résumé

La prolifération des fausses informations dans les médias numériques s'est étendue aux images manipulées, à l'audio de synthèse et aux vidéos deepfakes. Les progrès récents de l'IA générative font des systèmes de détection multimodaux robustes une nécessité urgente. Cette thèse propose deux nouvelles architectures multimodales pour la détection des infox (*fake news*). La première architecture cible les contenus image-texte en extrayant des caractéristiques visuelles (SigLIP), textuelles (Longformer), fréquentielles (DCT/FFT) ainsi que des caractéristiques de raisonnement basées sur des grands modèles de langage (LLM) (Llama-3), fusionnées via un mécanisme de Mélange d'Experts (MoE) pour générer des prédictions de véracité et des explications en langage naturel. La seconde architecture analyse les vidéos courtes en extrayant onze ensembles de caractéristiques à travers les domaines visuel, acoustique et textuel, utilisant l'attention cross-modale pour combiner les représentations des images clés (SigLIP), de l'audio (Wav2Vec2), du texte (RoBERTa) et de la cohérence (Phi-3 Mini). Évalué sur le jeu de données Fakeddit, le premier système atteint une précision de 85,03%, les caractéristiques visuelles fournissant le signal de détection le plus critique. Le système vidéo atteint une précision de 91,30% sur le jeu de données FakeTT, où les caractéristiques faciales et temporelles s'avèrent essentielles. En fin de compte, ce travail démontre que l'intégration de multiples modalités et du raisonnement explicite des LLM est indispensable pour créer des mécanismes de défense fiables, transparents et interprétables contre la déception numérique moderne.

Mots-clés : Détection des infox, apprentissage multimodal, apprentissage profond, transformeurs, grands modèles de langage, détection de deepfakes, attention cross-modale, explicabilité.

الملخص

لقد تطور انتشار المعلومات المضللة في وسائل الإعلام الرقمية ليشمل الصور المعدلة، والمقاطع الصوتية الاصطناعية، وفيديوهات التزييف العميق (فَكس). وبسبب التطورات الأخيرة في الذكاء الاصطناعي التوليدي، أصبحت أنظمة الكشف المتعددة الوسائط (شُلتِمدَل) والقوية ضرورة ملحة. تقترح هذه الأطروحة بنيتين هندسيتين متعددتي الوسائط للكشف عن الأخبار الزائفة؛ تستهدف البنية الأولى محتوى (الصورة-النص) عبر استخراج السمات البصرية (ضِشيص)، والنصية (شنفرمر)، والترددية (رة\سسة)، بالإضافة إلى سمات الاستدلال القائمة على نماذج اللغات الكبيرة (شلَمَ-٣)، حيث يتم دمجها عبر آلية ْمزيج الخبراءْ (شز) لإنتاج تنبؤات بالموثوقية وتفسيرات باللغة الطبيعية. أما البنية الثانية، فتقوم بتحليل الفيديوهات القصيرة من خلال استخراج إحدى عشرة مجموعة من السمات عبر النطاقات البصرية والسمعية والنصية، مستخدمةً آلية ْالانتباه عبر الوسائطْ (ررسس-مدَل تتنتِن) لدمج تمثيلات الإطارات المفتاحية (ضِشيص)، والصوت (طَ٢ط٢)، والنص (ضزضةَ)، والاتساق (صهِ-٣ شنِ). عند التقييم على قاعدة بيانات سَكددِت، حقق النظام الأول دقة بلغت ٣٠.٥٨٪، حيث قدمت السمات البصرية إشارة الكشف الأكثر أهمية. بينما حقق نظام الفيديو دقة بلغت ٠٣.١٩٪ على قاعدة بيانات سَكدة، حيث أثبتت السمات الوجهية والزمنية أنها أساسية. في الختام، يوضح هذا العمل أن دمج الوسائط المتعددة واستدلال نماذج اللغات الكبيرة (شش‌ش) الصريح يظل أمرًا أساسيًا لإنشاء آليات دفاع موثوقة، وشفافة، وقابلة للتفسير ضد الخداع الرقمي الحديث.

الكلمات المفتاحية: الكشف عن الأخبار الزائفة، التعلم متعدد الوسائط، التعلم العميق، المحولات، نماذج اللغات الكبيرة، كشف التزييف العميق، الانتباه عبر الوسائط، القابلية للتفسير.

Contents

List of Figures **1**

List of Tables **2**

List of Acronyms **3**

General Introduction **4**

1 State of the Art **6**

1.1 Introduction 7
1.2 What is Fake News 7
1.3 Fake News Categories 7
1.3.1 Misinformation 8
1.3.2 Disinformation 8
1.3.3 Malinformation 8
1.4 Related Concepts of Fake News 9
1.4.1 Fabricated content 9
1.4.2 Propaganda 9
1.4.3 Rumors 9
1.4.4 Hoaxes 10
1.4.5 Spam 10
1.4.6 Satire and Parody News 10
1.4.7 Click-Bait 10
1.5 Transition to Multimodal Disinformation 10
1.6 Emergence of Deepfakes and Generative AI 10
1.6.1 Visual-Based Deepfakes 11
1.6.2 Audio-Based Deepfakes 11
1.6.3 Text-Based Deepfakes 11
1.7 Adverse Effects and Impact of Fake News 11
1.7.1 Impact on Public Health 11
1.7.2 Political and Democratic Impact 11
1.7.3 Social and Individual Impact 12

1.7.4 Financial and Economic Impact . . . 13
1.8 Foundations of Fake News Detection . . . 13
1.8.1 Content-Based Approaches . . . 13
a) Knowledge-Based . . . 13
b) Linguistic-Based . . . 14
c) Style-Based . . . 14
d) Visual and Multimodal-Based . . . 14
1.8.2 Context-Based Approaches . . . 14
a) Network-Based . . . 14
b) Credibility-Based . . . 14
c) Social Media Platform Measures . . . 15
1.9 Challenges in Fake News Detection . . . 16
1.9.1 Volume and Velocity of Diffusion . . . 16
1.9.2 Variety of Fake News Content . . . 16
1.9.3 Sophistication of Generative AI . . . 16
1.9.4 Scarcity of Accurate Multimodal Datasets . . . 16
1.10 Natural Language Processing (NLP) . . . 17
1.10.1 Definition and Overview . . . 17
1.10.2 Text Representation Methods . . . 17
1.10.3 Related Works Using NLP for Fake News Detection . . . 17
1.11 Machine Learning (ML) . . . 18
1.11.1 Definition and Overview . . . 18
1.11.2 Classical ML Algorithms . . . 18
1.11.3 Related Works Using ML for Fake News Detection . . . 19
1.12 Deep Learning (DL) . . . 19
1.12.1 Definition and Overview . . . 19
1.12.2 Key Architectures and Feature Extraction . . . 19
1.12.3 Related Works Using DL for Fake News Detection . . . 21
1.13 Transformer-Based Approaches . . . 22
1.13.1 Definition and Overview . . . 22
1.13.2 Key Transformer Models . . . 22
1.13.3 Related Works Using Transformer-Based Methods for Fake News Detection . . . 25
1.14 Large Language Models (LLMs) Integration (2023–2025) . . . 26
1.14.1 Definition and Overview . . . 26
1.14.2 LLM Techniques for Fake News Detection . . . 26
1.14.3 Related Works Using LLMs for Fake News Detection . . . 27
1.15 Theoretical Background . . . 30
1.15.1 Transformer Architecture and Attention Mechanism . . . 30

1.15.2 Long-Document Text Encoding . . . 31
1.15.3 Pre-trained Language and Vision Models . . . 32
1.15.4 Visual Representation Learning . . . 33
1.15.5 Image Captioning . . . 33
1.15.6 Large Language Models for Reasoning . . . 33
1.15.7 Frequency-Domain Image Analysis . . . 34
1.15.8 Face Detection . . . 35
1.15.9 Activation and Output Functions . . . 35
1.15.10 Optimization and Network Regularization . . . 36
1.15.11 Other Neural Network Building Blocks . . . 37
1.15.12 Feature Conditioning Mechanisms . . . 38
1.15.13 Cross-Modal Interaction Mechanisms . . . 38
1.15.14 Loss Functions . . . 39
1.15.15 Similarity and Distance Measures . . . 40
1.15.16 Training and Optimization Techniques . . . 41
1.16 Performance Comparison . . . 42
1.17 Conclusion . . . 42

2 Proposed Approaches **44**
2.1 Introduction . . . 45
2.2 Approach 1 : Image + Text . . . 47
2.2.1 Feature Extraction . . . 47
2.2.2 Visual Feature Extraction . . . 48
2.2.3 Image Captioning . . . 48
2.2.4 Text Feature Extraction . . . 49
2.2.5 Frequency-Domain Feature Extraction . . . 49
a) DCT High-Frequency Features . . . 49
b) Radial FFT Features . . . 49
c) Combination and Normalisation . . . 50
2.2.6 LLM-Based Reasoning . . . 50
a) Prompt Design . . . 50
b) Fallback Response . . . 51
2.2.7 LLM Scalar Feature Vector . . . 51
2.2.8 LLM Reasoning Embedding . . . 52
2.2.9 Summary of Extracted Features . . . 52
2.2.10 Classification Models . . . 53
2.2.11 Modality Projection and Alignment . . . 54
a) Heterogeneous Modality Projection . . . 54
b) Feature-wise Linear Modulation (FiLM) . . . 54

c) Learned Modality Type Embeddings 54
d) Feature Noise and Structured Modality Dropout 55
2.2.12 Classification Model Architecture 55
a) Intra-Modal Hierarchical Self-Attention 55
b) Bidirectional Co-Attention with Inverse Attention 56
c) Coarse-Grained Similarity Representation 56
d) Mixture-of-Experts (MoE) Adaptive Gated Fusion 56
e) Explicit Cross-Modal Dissimilarity 57
f) Residual Classifier Head 57
2.2.13 Multi-Task Learning and Loss Functions 58
a) Focal Loss 58
b) Cross-Modal Consistency Loss 58
c) Contrastive Alignment Loss (InfoNCE) 58
d) Sinkhorn Optimal Transport Loss 58
e) Overall Multi-Task Loss 59
2.3 Approach 2 : Videos 60
2.3.1 Feature Extraction 60
2.3.2 Adaptive Keyframe Selection 61
2.3.3 Textual Feature Extraction 61
2.3.4 Visual Feature Extraction 61
2.3.5 Acoustic Feature Extraction 62
2.3.6 Frequency-Domain Feature Extraction 62
2.3.7 Face Analysis Features 64
2.3.8 Temporal Consistency Features 64
2.3.9 Audio-Visual Synchronization Score 65
2.3.10 Color Distribution Features 66
2.3.11 Metadata Features 66
2.3.12 LLM-Based Cross-Modal Consistency Reasoning 67
a) Model Inputs 67
b) LLM Scalar Feature Vector 68
c) LLM Reasoning Embedding 68
d) Fallback Response 68
2.3.13 Summary of Extracted Features 69
2.3.14 Classification Architecture 69
2.3.15 Feature Projection Layer 70
2.3.16 Cross-Modal Attention 71
2.3.17 Feature Fusion and Classification 72
2.3.18 Training Strategy 73
2.4 Conclusion 74

3 Experimental Results and Discussion **75**
3.1 Introduction . . . 76
3.2 Experimental Environment and Setup . . . 76
3.2.1 Hardware and Cloud Infrastructure . . . 76
3.2.2 Development Environment . . . 76
3.3 Metrics Utilized for Evaluating Model Performance . . . 77
3.3.1 Confusion Matrix . . . 78
3.3.2 Evaluation Metrics . . . 78
3.4 Approach 1: Multimodal Fake News Detection for Image and Text . . . 79
3.4.1 Dataset . . . 79
3.4.2 Hyperparameter Configuration . . . 79
3.4.3 Training and Convergence . . . 80
a) Training Dynamics . . . 80
b) Macro F1 Score Progression . . . 81
c) Overfitting Analysis . . . 82
d) Cross-Validation Stability . . . 83
3.4.4 Test Set Performance . . . 84
a) Per-Class Analysis . . . 84
b) Confusion Matrix . . . 85
3.4.5 Ablation Study . . . 86
a) Leave-One-Out Ablation . . . 86
b) Single-Modality Evaluation . . . 87
3.4.6 Comparison with Published Methods . . . 88
3.4.7 Limitations . . . 88
3.5 Approach 2: Multimodal Deepfake Detection for Videos . . . 89
3.5.1 Dataset . . . 89
3.5.2 Hyperparameter Configuration . . . 89
3.5.3 Training and Convergence . . . 90
a) Training Dynamics . . . 90
b) Macro F1 Score Progression . . . 91
c) Overfitting Analysis . . . 92
d) Cross-Validation Stability . . . 93
3.5.4 Test Set Performance . . . 94
a) Per-Class Analysis . . . 95
b) Confusion Matrix . . . 95
3.5.5 Ablation Study . . . 96
a) Leave-One-Out Ablation . . . 96
b) Single-Modality Evaluation . . . 98
3.5.6 Comparison with Published Methods . . . 99

3.5.7 Limitations 99
3.6 Conclusion 100

General Conclusion **102**

Bibliography **103**

List of Figures

1.1 Fake news categories 9
1.2 DeepFakes Types 12
1.3 Fake news detection approaches 15

2.1 Overview of the proposed multimodal fake news detection framework. . . . 46
2.2 Feature Extraction 47
2.3 Architecture of the proposed five-modality fusion classifier 53
2.4 Feature extraction pipeline for the video-based approach 60
2.5 Fusion and classification architecture for the video-based approach 70

3.1 Training and validation loss curves over 13 epochs. 81
3.2 Training and validation macro F1 score over 13 epochs. 82
3.3 Overfitting monitor: train–validation F1 gap across 13 epochs. 83
3.4 Five-fold cross-validation results on the Fakeddit dataset. 83
3.5 Validation loss (left) and macro F1 (right) curves across all five folds. . . . 84
3.6 Confusion matrix on the Fakeddit test set. 85
3.7 Modality importance measured by macro F1 score change upon removal. . 86
3.8 Single-modality evaluation on the Fakeddit dataset. 87
3.9 Training and validation loss curves on the FakeTT dataset. 91
3.10 Training and validation macro F1 score on the FakeTT dataset. 92
3.11 Overfitting monitor: train–validation F1 gap across training epochs on the FakeTT dataset. 93
3.12 Five-fold cross-validation results on the FakeTT dataset. 94
3.13 Validation loss (left) and macro F1 (right) curves across all five folds on the FakeTT dataset. 94
3.14 Confusion matrix on the FakeTT test set. 96
3.15 Modality group importance measured by macro F1 score change on the FakeTT dataset. 97
3.16 Modality-subset evaluation on the FakeTT dataset showing macro F1 (left) and AUC-ROC (right). 98

List of Tables

1.1 Performance comparison of fake news detection systems. 42

2.1 Summary of Extracted Feature Groups (Approach 1: Image + Text) 52
2.2 Summary of Extracted Feature Groups (Approach 2: Video) 69
2.3 Per-modality projection dimensions 71

3.1 Confusion Matrix Terminology in Fake News Detection 78
3.2 Hyperparameter configuration for the fusion classifier (Approach 1). 80
3.3 Test set performance of the proposed fusion classifier (Approach 1). 84
3.4 Per-class precision, recall, and F1 score on the Fakeddit test set. 85
3.5 Leave-one-out ablation study on the Fakeddit dataset. ΔF1 indicates the change in macro F1 relative to the full model (0.8467). 86
3.6 Single-modality evaluation on the Fakeddit dataset. ΔF1 indicates the change in macro F1 relative to the full model (0.8467). 87
3.7 Comparison of Approach 1 with published fake news detection methods. . 88
3.8 Hyperparameter configuration for the video-based classifier (Approach 2). . 90
3.9 Test set performance of the proposed video-based classifier (Approach 2). . 95
3.10 Per-class precision, recall, and F1 score on the FakeTT test set. 95
3.11 Leave-one-out ablation study on the FakeTT dataset. ΔF1 indicates the change in macro F1 relative to the full model (0.9090). 97
3.12 Modality-subset evaluation on the FakeTT dataset. 98
3.13 Comparison of Approach 2 with published methods on the FakeTT dataset. 99

List of Acronyms

LLM	Large Language Model
CoT	Chain-of-Thought
MoE	Mixture-of-Experts

General Introduction

The Internet has become an integral part of people's lives, with millions of users worldwide sharing and consuming content on a daily basis. While this has provided individuals with unprecedented access to information, it has also opened new avenues for the rapid spread of misinformation and fake news. A recent study found that false news stories are 70% more likely to be retweeted on Twitter than true stories [1], making it increasingly difficult to distinguish between fact and fiction, especially given the rising complexity and volume of digital data. The 2020 US Presidential election serves as a clear reminder of the importance of addressing these issues, as Facebook had to flag and remove over 120,000 posts for sharing false or misleading information [2]. The situation has become even more critical with the advent of generative AI, where malicious actors can now create highly realistic multimodal fake news that easily bypass traditional filters.

The spread of fake news online can have far-reaching consequences for individuals, communities, and even entire nations. Fake news, which refers to intentionally false or misleading information presented as factual news, can be used to manipulate public opinion, spread fear and hate, and undermine trust in institutions and democratic processes. This has resulted in an urgent need to develop effective, multi-layered fake news detection techniques.

In response to this need, researchers have developed computational methods to automate the process of fake news detection. While manual fact-checking and crowdsourcing can be effective, they are not scalable or cost-effective in the face of millions of daily posts. Machine learning-based approaches, specifically deep learning and large vision-language models (VLMs), have demonstrated tremendous potential in recognizing complex manipulation patterns. Nonetheless, more advanced research is required to improve the accuracy, reliability, and explainability of these techniques when dealing with cross-modal content.

The focus of this thesis is to propose and evaluate advanced multimodal approaches for detecting fake news and digital deepfakes. This study explores the effectiveness of combining state-of-the-art deep learning architectures. Specifically, we investigate the integration of Small Language Models (SLMs) for robust feature representation with Large Language Models (LLMs) for semantic reasoning. The proposed frameworks utilize cutting-edge transformer models, namely Longformer, RoBERTa, SigLIP for visual encoding, and Wav2Vec2 for audio processing, unified through cross-modal attention mechanisms and a

Mixture-of-Experts (MoE) routing system.

To achieve the thesis objectives, two widely recognized datasets are used: the Fakeddit dataset for image-text fake news detection, and the FakeTT dataset for video deepfake detection. These datasets are preprocessed and employed to train the selected models. To evaluate the performance of the architectures, various metrics including Accuracy, Precision, Recall, F1-score, and AUC-ROC are utilized. Furthermore, the proposed approaches are evaluated against existing state-of-the-art methods to assess their effectiveness. The significance of this thesis lies in its potential to contribute to the development of highly accurate, explainable, and scalable methods for combating the next generation of multimodal disinformation.

Thesis Outline

The subsequent sections of this thesis are organized as follows:

- **Chapter 1** introduces the theoretical foundations of fake news, covers its terminology, the emergence of deepfakes, and its societal consequences. It also provides a comprehensive review of prior research and state-of-the-art machine learning methodologies for fake news identification.
- **Chapter 2** details the proposed technical architectures. It explains the dual frameworks developed in this study, detailing the extraction and fusion of multiple modalities (visual, textual, acoustic, and LLM reasoning).
- **Chapter 3** presents the experimental setup, providing information on the datasets and training configurations employed. The findings and performance metrics of the experiments are discussed, followed by a general conclusion and future research possibilities.

Chapter 1

State of the Art

Part I: Definitions and History

1.1 Introduction

In the early days of the Internet, it was widely perceived as a reliable repository of accurate information [3]. However, with the rapid rise of social media and recent advances in artificial intelligence, it has transformed into a fertile environment for the dissemination of misinformation [1]. This shift has accelerated the transition from traditional, text-based fake news to sophisticated multimodal forms of deception, most notably through deepfakes. Deepfake technology leverages generative artificial intelligence to produce highly realistic visual, audio, and textual content that is increasingly difficult to distinguish from reality [4], raising serious concerns and driving the need for robust detection frameworks.

Furthermore, the emergence of advanced multimodal models such as GPT-4V, LLaMA 3, and Gemini 1.5 between 2023 and 2024 has made generating deceptive content easier and more convincing than ever before. Consequently, research in multimodal fake news detection has surged, culminating in a peak of publication volume in 2024 [5].

This chapter is structured into two main parts to provide a comprehensive overview of the field. Part I establishes the foundational concepts, defining the categories of fake news (misinformation, disinformation, and malinformation), exploring its adverse societal impacts across health, political, social, and economic domains, and outlining traditional detection approaches and their core challenges. Part II traces the technological evolution of detection methodologies over time, evaluating how the field transitioned from classical Natural Language Processing (NLP) and Machine Learning (ML) techniques to Deep Learning (DL), Transformer-based approaches, and modern Large Language Model (LLM) integrations. By analyzing what succeeded and what failed, this chapter highlights the open challenges that continue to push the boundaries of research in this domain.

1.2 What is Fake News

Fake news is defined as false information that resembles the content of news media in form but differs in intent [6]. Fake news is distinguished from traditional satire and parody in that it attempts to deceive the viewer for political or ideological gains or for money [7]. Research indicates that fake news is characterized by three distinct phenomena: misinformation, disinformation, and malinformation [8].

1.3 Fake News Categories

All above related concepts of fake news fall under three categories:

1.3.1 Misinformation

Misinformation is inaccurate or false information that arises primarily from human error, misunderstanding, or a failure to verify the source of the information. Crucially, it is published and shared without any malicious intent or deliberate desire to deceive the reader. This phenomenon is particularly common and widely circulated during times of sudden crises, natural disasters, or breaking news events, when individuals rush to share unverified updates out of a genuine desire to be helpful or inform their networks [8]. For example, sharing a satirical article believing it to be a real news report constitutes misinformation.

1.3.2 Disinformation

Disinformation is the deliberate and organized creation and spread of false or misleading information with the explicit intention of deceiving a target audience [8]. Unlike misinformation, disinformation involves a malicious motive, often driven by political, ideological, or financial incentives. It is frequently employed by state actors, coordinated troll farms, or ideologically motivated groups to manipulate public opinion, discredit opponents, or sow societal division. The malicious actors know the information is false, but they weaponize it to achieve a specific agenda.

1.3.3 Malinformation

Malinformation relies on genuine, factual information, but it is deliberately weaponized to inflict harm on individuals, organizations, or institutions [8]. This category often involves the strategic leakage of highly sensitive private data, revenge porn, or the deliberate presentation of truthful information entirely out of its original context. While the foundational facts may be authentic, the malicious intent and the deceptive framing used to cause defamation or reputational damage align it closely with the dangers posed by traditional fake news. The distinction between misinformation, disinformation, and malinformation can be visually summarised through the intersection of truthfulness and malicious motivation, as depicted in Figure 1.1.

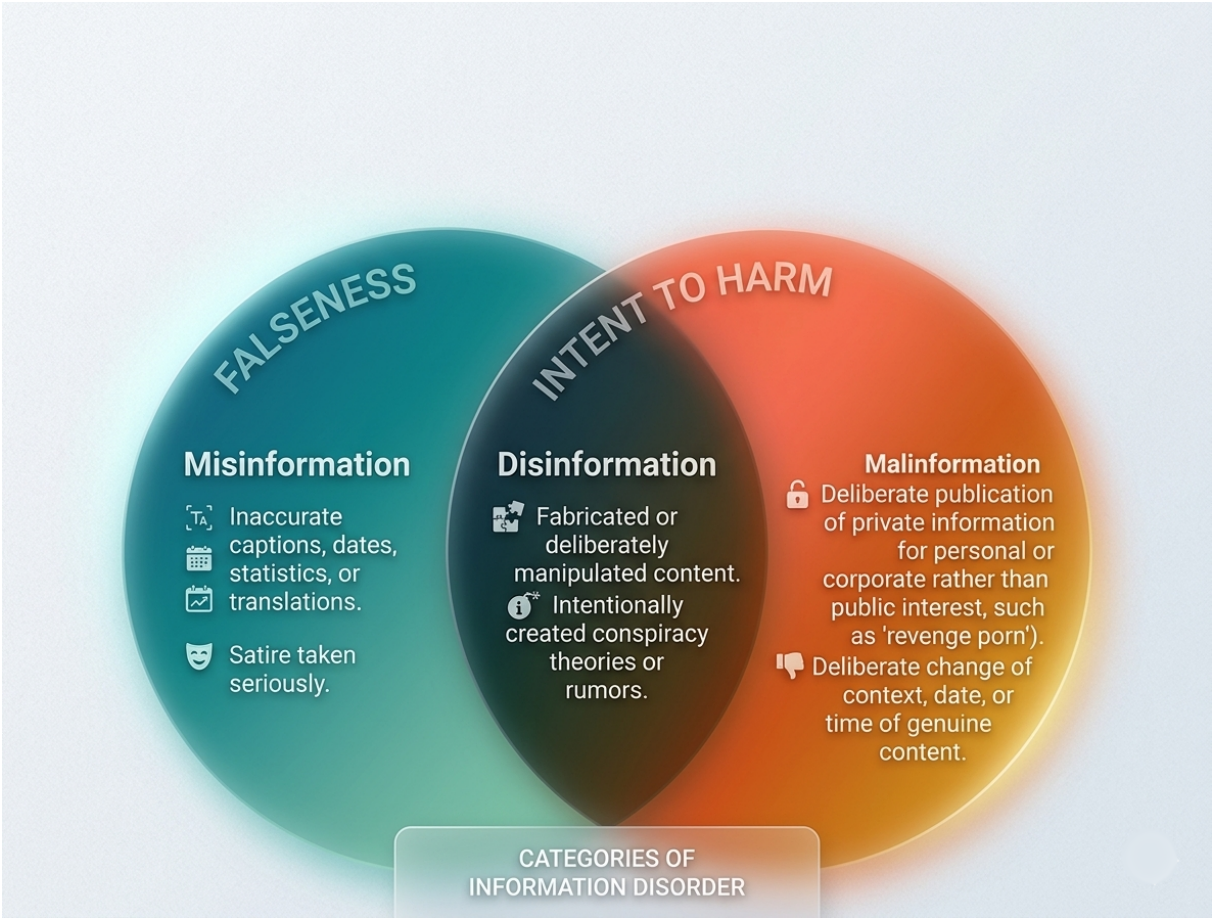


Figure 1.1: Fake news categories

1.4 Related Concepts of Fake News

1.4.1 Fabricated content

This category of news is considered dangerous because it cannot be distinguished from real news, as it provides completely fabricated content for bad motives, which helped it spread widely on social media [8] .

1.4.2 Propaganda

It is misleading information that aims to serve a specific agenda of social, religious, or political interests. The information is manipulated and the appropriate solutions are shown to serve a specific issue [9].

1.4.3 Rumors

A rumor is defined as an unverified claim or story that spreads from one person to another [10]. Unlike traditional news, rumors are often characterized by their informal origin

and rapid spread during periods of public anxiety .

1.4.4 Hoaxes

It is a type of misleading information in which false news is created or published with the aim of convincing others that it is true. It is used in order to influence the public or cause confusion in society [8].

1.4.5 Spam

Spam is unsolicited messages sent to a large number of email or social media users, used to post malicious ads or links for fraudulent activities [11].

1.4.6 Satire and Parody News

It is real news presented in a sarcastic or humorous way in order to entertain or criticize readers, in which current events are commented on in order to shed light on social or political issues [12].

1.4.7 Click-Bait

It is a type of low-quality journalism that uses exaggerated headlines and may be accompanied by images to attract attention in order to entice users to click on them. It aims to generate revenue through advertising by increasing the number of clicks [13].

1.5 Transition to Multimodal Disinformation

Fake news has evolved from simple text-based content to a multimodal form that combines text, images, audio, and video [14]. Unlike written misinformation alone, visual and audio fake news is much harder to verify because it is more convincing to the audience, which has led to its repeated use by fake news publishers and its deployment across various fields [15]. With the advancement of artificial intelligence and the widespread use of social media, fake news has become a real and serious threat to individuals that must be addressed.

1.6 Emergence of Deepfakes and Generative AI

Deepfake media refers to artificial content generated using deep learning techniques to imitate real people or events, often making it difficult to distinguish from authentic media [16]. Deepfakes can be categorized into three primary types depending on the primary

medium they alter: visual, audio, and text-based deepfakes. Each category involves distinct manipulation techniques, as depicted in Figure 1.2, which illustrates the various types and their respective applications.

1.6.1 Visual-Based Deepfakes

Visual deepfakes are synthetic media created using deep generative models. This technology allows the imitation of human faces, including facial expressions and lip movements, which can lead to impersonation of individuals and the creation of fabricated events [4].

1.6.2 Audio-Based Deepfakes

This technology enables the generation of artificial audio using deep learning models, allowing the replication of speech characteristics such as tone, pitch, and speech style, which can lead to the creation of fabricated audio recordings [4].

1.6.3 Text-Based Deepfakes

Deep text-based fakes refer to the generation of written content using large language models such as GPT-4, PaLM, Claude, and LLaMA. These models can imitate human writing styles, producing text that resembles the writing of a specific person [17].

1.7 Adverse Effects and Impact of Fake News

1.7.1 Impact on Public Health

Fake news about public health is misleading health information that is circulated among people. This information influences individuals' behavior, leading to negative consequences such as reliance on unreliable medical treatments that may even lead to death or failure to take officially approved vaccines [18]. Examples include skepticism about vaccines during the Zika and Ebola outbreaks [19], and the spread of false claims about COVID-19 on social media [20]. Such misinformation can lead to a decline in public trust in scientific expertise.

1.7.2 Political and Democratic Impact

Fake political news is defined as a form of "Computational Propaganda" in which public opinion is manipulated during political events, creating biases in favor of a particular party or entity. These actors exploit social media to direct voters toward specific candidates, which could undermine the credibility of legitimate news organizations. The primary goal of these practices is to interfere with the integrity of states' electoral processes in

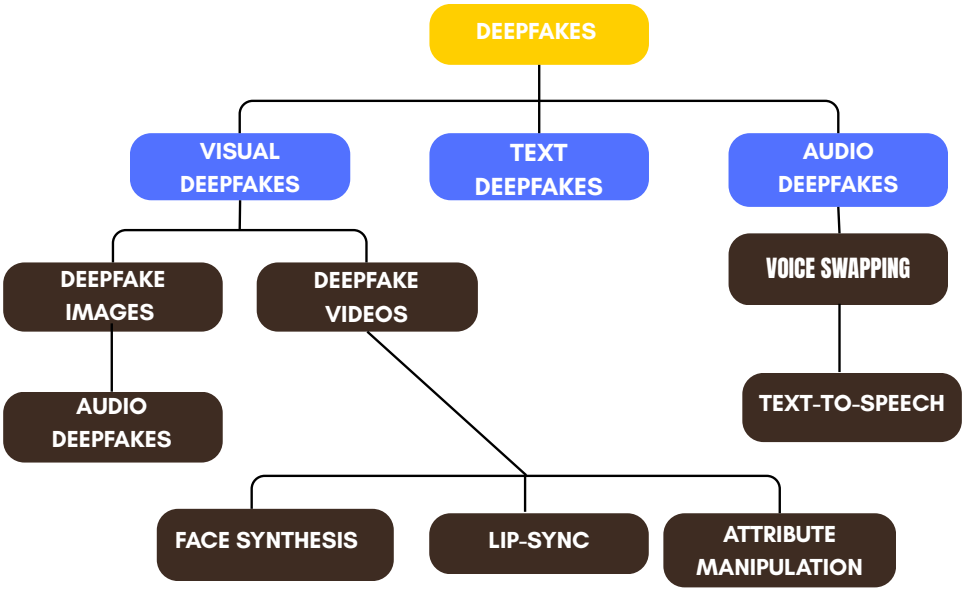


Figure 1.2: DeepFakes Types

order to achieve ideological or financial gains. Research has shown that fake news affected several important events, such as the Brexit referendum in 2016 [21] and the 2017 French presidential election [22]. During the 2016 US presidential election, fake news had a stronger influence on voter opinion than television campaigns [7].

1.7.3 Social and Individual Impact

The spread of false information has negative consequences for individuals and societies, as it affects individuals' decisions and behaviors in several vital areas such as health and politics, causing divisions in society and generating a loss of trust in institutions and official media [1]. For example, in India, rumors spread on WhatsApp claiming the existence of gangs kidnapping children, which resulted in mob violence and the killing of innocent people in several regions [23]. Similarly, during the European migration crisis, fabricated stories led to the spread of racism, leading to increased hatred and social divisions in many European countries [24].

1.7.4 Financial and Economic Impact

Fake news has a significant impact on the economy , as misinformation can lead to market volatility and huge losses for companies that may even lead to bankruptcy. This can create uncertainty in financial markets due to sudden fluctuations in stock prices, leading to constant changes in market value. Fake news reduces consumer confidence in companies and financial institutions [25]. For example, In 2022, an account impersonating a pharmaceutical company called "Eli Lilly" announced that insulin medicine had become free for everyone, which led to a 4.3 % decline in the company's shares and a loss of approximately $15 billion in its market value in just one day [26]. In another example, in 2016, a fake press release was published stating that Vinci's CFO had been fired due to accounting errors that led to an 18% drop in the company's stock price and a loss of about $7 billion in market value within minutes [27]. These examples show how false information can affect companies and investors and cause financial crises and economic turmoil .

1.8 Foundations of Fake News Detection

1.8.1 Content-Based Approaches

a) Knowledge-Based

Knowledge-based methods are based on evaluating news by comparing it with facts extracted from reliable sources [28] .They include two main approaches :

A. Manual Knowledge-Based Verification

- **a) Expert-based methods** These methods rely on human experts in verifying news, who apply specific strategies to assess and judge the credibility of a given piece of news [28] .
- **b) Crowdsourced Methods:** It relies on allowing a large group of non-expert individuals to evaluate and discuss the news, which leads to greater scalability. However, it may result in biases toward certain parties, making it less reliable [29] .

B. Automatic Knowledge-Based Verification

Unlike manual verification, these techniques rely on machines to verify the accuracy of news using tools such as natural language processing (NLP), data mining, machine learning (ML), network/graph theory methods, and many others [30] . They are done in two stages:

- **a) Fact extraction:** At this stage, data is collected and transformed into a structured form that can be processed by the machine.

- **b) Fact-checking:** At this stage, the system determines the accuracy of news articles by comparing them with a pre-existing knowledge base.

b) Linguistic-Based

This method relies on analyzing language characteristics such as grammar, vocabulary, and lexical features. Natural language processing (NLP) techniques are used to extract linguistic features that reveal the possibility that a news item is fake [28] .

c) Style-Based

Style-based methods analyze the writing style of news content to detect fake news. Fake news typically has a writing style that differs from reliable news, as it tends to use provocative and emotionally charged vocabulary to manipulate readers. Furthermore, fake news publishers typically use the clickbait technique, which consists of inserting attention-grabbing headlines designed to entice users to click on a particular web page [28].

d) Visual and Multimodal-Based

With the development of social media, news has evolved from text-only content to multimodal news incorporating images, video, and audio accompanying the article. It has become necessary to rely on visual and multimodal approaches in order to detect manipulation in news content. Features are extracted from each modality and then fused together to obtain a complete representation of the news, allowing the system to assess its credibility [31] .

1.8.2 Context-Based Approaches

a) Network-Based

Fake news differs from real news in the way it spreads. The spread of fake news usually begins with non-influential users and is subsequently followed by opinion leaders. In contrast, real news usually spreads from influencers to ordinary people [32]. These methods use techniques such as graphical neural networks (GNNs) to learn how news spreads and the relationship between users to judge the veracity of news [33] .

b) Credibility-Based

This approach is based on the premise that news from reliable sources is less likely to be fake. Instead of analyzing the text of the news, this technique evaluates the credibility of news based on certain indicators such as account age, number of followers, and the credibility of previously published news, which makes the system better able to identify fake news published by fraudulent accounts [10].

c) Social Media Platform Measures

Social media platforms rely on technical tools to combat the spread of fake news. These tools include placing alerts on unverified content and suspending accounts that repeatedly disseminate false information. Furthermore, algorithmic systems are used to detect and flag misleading posts before they reach a larger segment of users who may further amplify their spread, thus enabling users to distinguish between fake and accurate information [6].

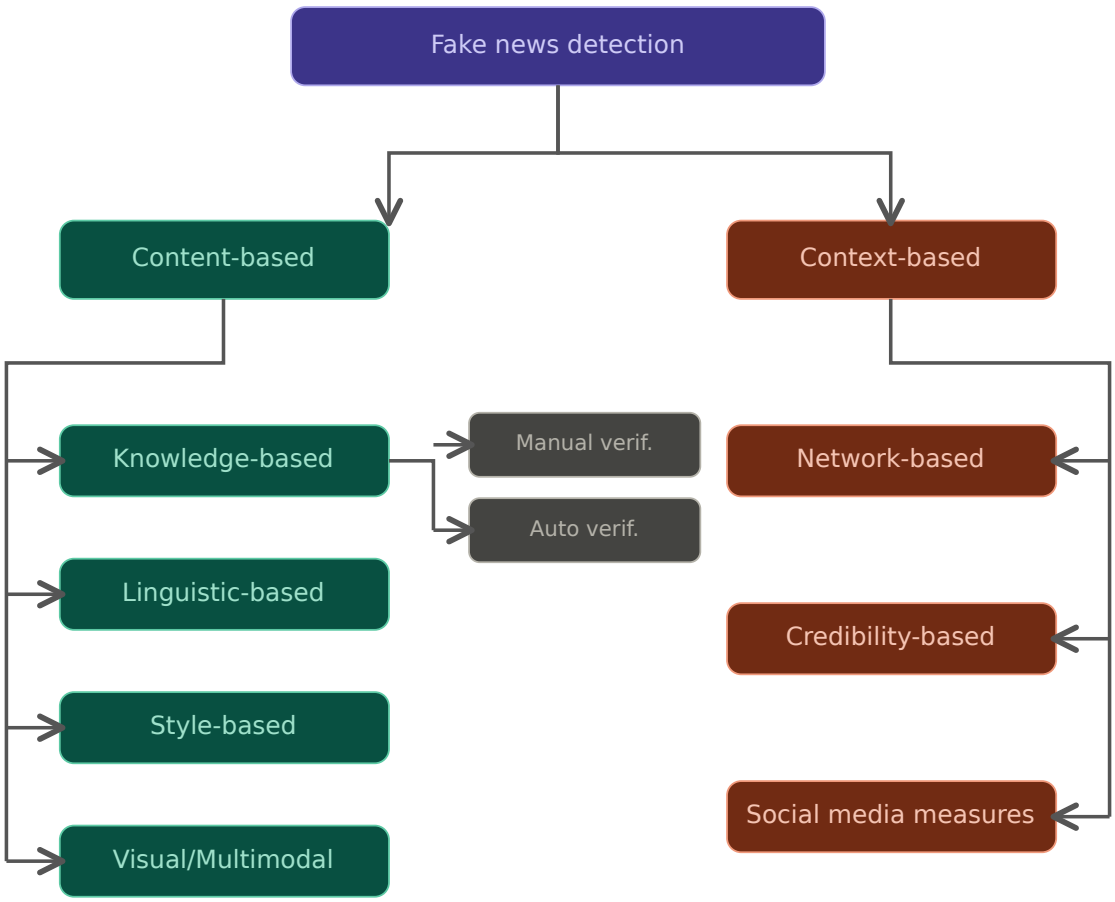


Figure 1.3: Fake news detection approaches

1.9 Challenges in Fake News Detection

1.9.1 Volume and Velocity of Diffusion

Millions of posts are published on many platforms every day, making verifying all shared content an almost insurmountable task. The matter is further complicated by the rapid dissemination of fake news; as a study by Vosoughi et al. [1] showed, lies spread six times faster than the truth. As a result, misinformation reaches a wide audience before it is detected by automated detection systems.

1.9.2 Variety of Fake News Content

Fake news has many categories including propaganda, hoaxes, rumors, satire, deepfakes, and clickbait [9, 8], created with the aim of influencing individuals for political, ideological , or financial gain. This diversity poses a major challenge to fake news detection systems [28].

1.9.3 Sophistication of Generative AI

Artificial intelligence has become a powerful weapon in the hands of fake news publishers, becoming capable of mimicking human writing style, leading to the publication of fake articles similar to the style of official articles [17]. With the advent of Deepfakes technology, the matter has become more complicated, as this technology has allowed the creation of images, video clips, and sounds that are difficult to distinguish from real content. This development in artificial intelligence necessitates more advanced, multimodal solutions [14].

1.9.4 Scarcity of Accurate Multimodal Datasets

Text data for fake news detection is widely available because it is inexpensive and can be obtained from multiple sources. In contrast, collecting multimodal data is more costly and time-consuming, as it requires manual verification to ensure accuracy in matching visual content with text [15]. This has resulted in limited data availability. This scarcity creates significant obstacles when training fake news detection models, which has led researchers to adopt advanced techniques such as LLMs and Transformers [14, 31].

Part II: Related Works

1.10 Natural Language Processing (NLP)

1.10.1 Definition and Overview

Early fake news classifiers treated text much like traditional document retrieval systems, relying on word counting and weighting techniques. NLP encompasses key tasks such as text classification, named entity recognition (NER), sentiment analysis, and summarization — all of which form the foundation for text-based fake news detection.

1.10.2 Text Representation Methods

TF-IDF [34] measures a word's importance by comparing its frequency in one document against its frequency across the entire collection. The representations it produces are both easy to interpret and computationally cheap. On the other hand, it ignores word order entirely and treats synonyms as unrelated terms — a serious weakness, given that the same idea can be phrased in many ways.

Word2Vec [35] and GloVe [36] addressed this gap by learning dense vector representations of words from large datasets, positioning semantically related words close together in vector space. A single fixed vector per word type, though, ignores context entirely. The word "bank" receives the same embedding whether it appears in "river bank" or "central bank."

FastText [37] improved on this by representing words as combinations of smaller subword units, making it better at handling rare or unseen words and languages with complex word forms. Sentence-level meaning, though, remains out of reach for all these approaches. Capturing full context across a sentence requires the deeper modelling that Transformer-based architectures later provided.

1.10.3 Related Works Using NLP for Fake News Detection

TF-IDF and static embedding techniques focus on word patterns rather than actual meaning. If someone rephrases a sentence using different words while maintaining the same meaning, these systems can be easily fooled. Fake news creators exploit this weakness to deceive models.

Ahmed et al. [38] established the ISOT benchmark, demonstrating that SVM with TF-IDF achieves near-perfect binary classification on a two-source corpus. The result mattered less for its accuracy than for its framing: it positioned fake news detection as a tractable supervised learning problem and triggered a wave of benchmark-oriented

research. Wang [39] corrected this framing with LIAR, exposing the gulf between source-domain pattern matching and genuine veracity assessment.

1.11 Machine Learning (ML)

1.11.1 Definition and Overview

Classical machine learning approaches applied to fake news classification span supervised, unsupervised, and semi-supervised learning paradigms. These methods rely on hand-crafted or statistically derived features combined with established classification algorithms.

1.11.2 Classical ML Algorithms

Ahmed et al. [38] achieved 96.1% accuracy on the ISOT dataset. Many cite this number as proof that fake news is easy to identify, but the claim is misleading. ISOT draws from only two sources — Reuters for real news and PolitiFact for fake news — so the model learned to recognise the style of these two websites, not to detect actual lies. On different news sources, the same model would struggle.

In contrast, Wang [39] presented a LIAR dataset containing over 12,000 short political statements. In this case, the best model achieved only 27.4% accuracy, a percentage only slightly better than guesswork. While machines easily recognise the writing style of specific news agencies, determining the truth of a short claim poses a much greater challenge. Mathematical formulas alone cannot solve this issue, as evaluating truth requires real-world knowledge. Approaching these two distinct tasks requires entirely different techniques.

Gradient Boosting Machines and Random Forests made systems more reliable by combining many small predictions, uncovering complex connections between data points that single models missed [40]. Castillo et al. [41] introduced a related but distinct idea: the way information spreads can itself signal whether it is true or false. Sharing frequency, poster identity, and audience reactions all carry clues. Fake news tends to spread faster, trigger stronger emotions, and originate from new or inactive accounts.

Popat et al. [42] developed DeClarE, a system based on Bi-LSTM architecture. Instead of just reading the text, DeClarE searches the web for relevant information to verify a claim. Fact-checking relies heavily on finding and using external evidence rather than simply analysing the original words. Retrieving and evaluating outside information became the foundation for the advanced AI detection systems built in 2024.

1.11.3 Related Works Using ML for Fake News Detection

As discussed in Section 1.10.3, Ahmed et al. [38] and Wang [39] established the foundational benchmarks for this field. Beyond those benchmarks, classical ML approaches revealed important structural limitations.

All older methods focus solely on text. As fake news has moved into images and videos, these text-only systems have become unable to detect manipulation in other media types. Models trained on specific datasets, such as political news, do not work with other topics like health, celebrity news, or crisis events. This is a significant problem that traditional machine learning has never been able to solve. These systems are inflexible. Once created, they cannot adapt to new words or emerging topics. To update them, the entire model must be retrained using new, labelled data.

1.12 Deep Learning (DL)

1.12.1 Definition and Overview

Neural architectures enabling richer feature extraction from text and images. Core concepts include neural networks, Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory networks (LSTMs).

1.12.2 Key Architectures and Feature Extraction

CNNs for Text and Image

One-dimensional convolutional filters proved effective at catching local patterns in text. Kim [43] introduced TextCNN, an architecture that passes an embedding matrix through parallel filter banks of width 2, 3, and 4 tokens and applies max pooling over each output to keep the strongest local signal. Fast and parallelisable, it produces a fixed-length representation regardless of document length. It captures only trigram-level patterns, though, and struggles with long-range dependencies — for instance, the relationship between an article's headline and its final paragraph.

For visual data, Simonyan & Zisserman [44] and He et al. [45] established the foundations of hierarchical feature extraction through VGGNet and ResNet respectively, both relying on stacked convolutional blocks. ResNet's residual skip connections were a particularly significant innovation, allowing gradients to pass directly through identity mappings and enabling stable training of networks exceeding 100 layers. ResNet-50 and ResNet-101 continue to appear in modern multimodal detection systems; Xu et al. [46] use ResNet-50 as the image encoder in their MAGIC architecture precisely because the representations these networks produce integrate reliably into downstream fusion pipelines. Tan & Le [47] extended this line of work with EfficientNet, which scales depth, width, and resolution

jointly using a single compound coefficient, delivering competitive performance at a significantly reduced parameter count.

Sequential Context Modeling and Attention Mechanisms

Long Short-Term Memory (LSTM) networks, introduced by Hochreiter and Schmidhuber [48], treat text as a sequence of hidden states passed through "gated" memory cells. These gates regulate data flow at each step, selectively erasing, writing, and reading information to manage the network's stored context. The design prevents the "vanishing gradient" problem — the tendency of earlier models to lose track of information over long sequences — allowing LSTMs to follow a story across many paragraphs rather than just short phrases.

Graves and Schmidhuber [49] extended this idea in 2005 with Bidirectional LSTMs, which process text from both the start and the end simultaneously. Joining these two directions gives the model a complete view of each word's context. Bahdanau et al. [50] later added "soft attention," a mechanism that lets the model assign different levels of importance to different words when forming its final output. Looking at data, weighing it by relevance, and combining the result — this pattern became the starting point for the Transformer architecture [51], which performs the same logic for all words at once in a fully parallel system.

Image-Text Consistency Checking

Placing images in the wrong context is one of the most common tricks in online misinformation. Someone takes a real photo from a particular time or place and pairs it with a misleading caption, changing its meaning entirely. Early detection systems addressed this through cosine similarity. Fang et al. [52], for example, converted both the picture and its accompanying text into mathematical vectors; when the gap between the two grew too large, the system flagged them as a mismatch.

Khattar et al. [53] took a different approach in 2019 with MVAE, a system that maps both the image and the text into a shared latent space and checks whether they fit together. If both describe the same event, the system can reconstruct the information from that shared space with few errors. Mismatched pairs produce higher reconstruction errors, which serves as the detection signal. Contrastive Language-Image Pre-training (CLIP) [54] later eclipsed MVAE by training on 400 million image–text pairs from the internet rather than a few thousand examples, dramatically improving accuracy at spotting out-of-context images.

Early Deepfake Detection Methods

Rossler et al. [55] developed FaceForensics++ in 2019, targeting five specific face manipulation methods — Deepfakes, Face2Face, FaceSwap, NeuralTextures, and FaceShifter — tested across three video quality levels. On high-quality videos, CNN models detected over 99% of fakes. The success was misleading. As Croitoru et al. [56] later showed, the same models dropped below 65% accuracy when confronted with manipulation techniques absent from their training data. Rather than learning general signs of manipulation, these models had memorised the specific digital traces left by particular programs. Overcoming this weakness became a central goal for the next generation of detection models built on Transformers.

1.12.3 Related Works Using DL for Fake News Detection

Qian et al. [57] deployed Generative Adversarial Network (GAN)-based adversarial training to synthesize hard negative examples that push the discriminator toward more generalizable decision boundaries, achieving 87.7% F1 on Twitter16. Popat et al. [42] demonstrated that external evidence retrieval — not richer internal text representation — was the bottleneck for knowledge-intensive claim verification, a finding that directly motivated the 2024 RAG-augmented LLM systems. Khattar et al. [53] showed that joint generative modeling of image and text through a shared latent space provides a principled consistency signal, though the 74.6% F1 they achieved left substantial room for the cross-modal attention mechanisms that followed.

Limitations of Pre-Transformer Deep Learning:

Pre-Transformer models shared several weaknesses that limited their practical use.

Limited Focus: Models like TextCNN examined only a small window of words at once. An article that stated a claim in its opening and contradicted it in its final paragraph would pass unnoticed, since the model could not relate the two.

Slow Processing: LSTM networks processed text one step at a time, preventing any parallel computation. Training was slow, and the maximum input was capped at around 500 words.

Poor Media Integration: Combining text and images by simply joining their vectors at the end did not work well. No mechanism existed to link a specific word to a specific region of an image.

Narrow Detection: Deepfake detectors trained on one manipulation program failed completely against forgeries from other programs. Neither CNNs nor LSTMs could learn a general rule for telling real content from fake.

1.13 Transformer-Based Approaches

1.13.1 Definition and Overview

The Transformer model [51] changed how models process data, replacing step-by-step processing with a system that analyses everything at once. This method is known as self-attention.

For each word (or token) in a sequence, the model creates three vectors: a query (Q), a key (K), and a value (V). To determine the level of emphasis on other words in the sentence, the model calculates a score using the scaled dot product:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1.1}$$

where d_k is the dimension of the key vectors. The result is a weighted average of the value vectors, with weights indicating how relevant each token is to the current position. Unlike older models, every pair of tokens communicates with every other pair simultaneously at each layer, eliminating the need to wait for sequential steps and making the system far more efficient.

Multi-head attention runs several independent attention processes in parallel, each operating on a different subspace of the representation. One head might capture subject–verb agreement while another tracks negation patterns. Residual connections and layer normalisation keep training stable as depth increases. Because the entire computation runs in parallel across the sequence on modern GPUs, training time drops from days to hours compared with LSTMs. This speed is what unlocked pre-training on hundreds of gigabytes of text — and it is in that pre-training that the real advantage builds up.

1.13.2 Key Transformer Models

BERT, RoBERTa & Transfer Learning

BERT [58] presented a method that uses a 12-layer Transformer encoder trained on 3.3 billion tokens. BERT employed Masked Language Modelling (MLM), where 15% of words are masked and the model must predict them using both the preceding and following context. This produces contextual representations in which the same word has a different mathematical embedding depending on the sentence. When BERT is fine-tuned on fake news data, it shifts its focus to veracity assessment while retaining the general linguistic knowledge acquired during pre-training.

RoBERTa [59] improved upon this by addressing four key issues in the original BERT design:

- **Data:** Training on significantly larger datasets (160 GB of text).

- **Objectives:** Removing the Next Sentence Prediction (NSP) task, which proved ineffective.
- **Masking:** Using dynamic masking, where the masked tokens change with each training epoch.
- **Optimisation:** Scaling batch sizes to 8,000 for more stable training.

By training on 160 GB of text, RoBERTa achieved 72.3% accuracy on the LIAR fake news dataset, a significant improvement over older machine learning systems [28]. Recent developments have focused on domain-specific pre-training. For example, AraBERT and CAMeLBERT were designed for the complex morphology of Arabic [60]. BERTweet was trained on 873 million tweets to understand colloquial language and hashtags not typically found in formal news. More recently, HEMT-Fake [61] used a multilingual model (XLM-RoBERTa) to detect fake news in several languages, including Hindi, Telugu, and English, demonstrating that these models generalise across languages provided the initial training data is sufficiently large.

Vision Transformer (ViT), CLIP & BLIP

ViT [62] redefined visual feature extraction by processing images analogously to textual sequences. Instead of analysing pixels individually, it segments an image into 1616 pixel patches, which are subsequently projected into visual tokens. A dedicated `[CLS]` token is appended to capture global image-level representations, and the entire sequence is processed by a standard Transformer encoder. When pre-trained on large-scale datasets such as ImageNet-21K, variant architectures like ViT-Large outperform classical convolutional baselines (e.g., ResNet-152) and demonstrate superior scalability with additional data.

CLIP [54] leverages cross-modal supervision by training on 400 million image–caption pairs sourced from the web. It employs dedicated image and text encoders to map both modalities into a shared embedding space. Through contrastive learning, the model maximizes the cosine similarity of matching image–text pairs while minimizing the alignment of unmatched pairs. In fake news detection, this architecture facilitates zero-shot verification: a low similarity score between a news image and its accompanying caption signals a semantic mismatch or out-of-context manipulation, eliminating the necessity for task-specific labelled datasets.

Bootstrapping Language-Image Pre-training (BLIP) [63] advances this framework by introducing a Querying Transformer (Q-Former), which bridges a frozen visual backbone with a large language model (LLM). The Q-Former utilises learned query vectors to extract bottlenecked visual features, transforming them into soft prompts compatible with the LLM. This multi-modal pre-training strategy was later scaled in BLIP-2 to accommodate significantly larger language architectures.

The TT-BLIP system [64] adapts this paradigm for multimodal fake news detection by concurrently processing three distinct information streams: text, image, and their mutual cross-modal relationship. These streams are fused via a Tri-Transformer architecture that leverages independent attention mechanisms to isolate cross-modal inconsistencies.

Multimodal Fusion Strategies

Early fusion joins text and image features together before classification, allowing the classifier to learn cross-modal relationships from the outset. However, the combined vector tends to be very large, and the modality with stronger activations can easily dominate the result.

Late fusion trains separate classifiers for text and image, then combines their outputs through averaging, weighted scoring, or a small meta-classifier. This works well even when one modality is absent, and each stream can be optimised independently. The drawback is that it cannot detect fine-grained conflicts between modalities, such as when a headline verb contradicts an action shown in the image. This limitation occurs because the two streams never share information during encoding.

Cross-modal attention is the current standard. Text and image representations are cross-attended: text queries image features and image queries text features, and the results are combined through learned projections before classification. MAGIC [46] applies this using an adaptive Graph Attention Network that builds an interaction graph over BERT text nodes and ResNet-50 image nodes, reaching 98.8% accuracy on Fakeddit. GAME-ON [65] demonstrates through ablation studies that detailed within-modality and cross-modality interaction yields a 4.3% accuracy gain over simple concatenation on the same dataset.

Cross-Modal Attention & Text-Image Alignment

Cross-modal attention extends self-attention across modality boundaries. Following the formulation in Equation 1.1, text representations generate queries Q_{text} while image representations generate keys K_{img} and values V_{img}:

$$A = \text{softmax}\left(\frac{Q_{\text{text}} \cdot K_{\text{img}}^T}{\sqrt{d_k}}\right) \tag{1.2}$$

The FND-LLM system [66] implements this as a bidirectional co-attention mechanism. One head treats text as the query and image as key–value, while a second head treats the image as the query and text as key–value. This iterative refinement allows the model to process text in light of the image and vice versa, significantly improving detection accuracy.

Advanced Deepfake Detection (2024)

The M4-BLIP framework (2024) [67] approached deepfake detection as a multimodal alignment problem rather than an artifact classification problem. Global visual features from a CLIP encoder capture high-level semantic content; local forgery features from a dedicated face manipulation detector capture spatial inconsistencies around face boundaries. Fine-grained Contrastive Alignment (FCA) projects both feature sets into a shared space, and a Multi-modal Local-and-Global Fusion (MLGF) module aggregates them before binary classification. Ablation experiments showed that adding local forgery features improved CLIP-only binary classification accuracy by 10 percentage points on held-out manipulation techniques, confirming that global semantic matching alone is insufficient for manipulation-method-agnostic detection.

Croitoru et al. [56] document the structural challenge this creates. Detection architectures trained on 2019-era GAN artifacts are evaluated against 2024-era neural rendering outputs and face systematically near-chance generalization. The arms race has reached a point where artifact-based detection is architecturally inadequate; the field is shifting toward semantic provenance reasoning: not "does this face look synthetic?" but "does the visual content of this video match what it claims to document?"

1.13.3 Related Works Using Transformer-Based Methods for Fake News Detection

Zhou et al. [28] combined RoBERTa with knowledge graph entity embeddings, demonstrating that external factual knowledge materially improves multi-class veracity prediction. Wu et al. [68] showed that co-attention between BERT text and ResNet image features produces a 2.8-point F1 improvement over late fusion on Weibo. Qi et al. [69] established CLIP's zero-shot cross-modal consistency signal as a competitive detection baseline, achieving 76.4% accuracy on NewsCLIPpings without any task-specific fine-tuning, a result that triggered the wave of CLIP-based detection systems that followed. Peng et al. [70] refined this with Contextual Semantic Representation Learning, demonstrating that modeling within-class semantic diversity in fake news categories, not treating all fake news as a monolithic cluster, improves detection accuracy by 3.1 points.

Limitations of Transformer-Based Approaches:

- Self-attention scales quadratically with sequence length. Processing a 512512 image as a sequence of 1,024 patches, combined with a 512-token article, produces attention matrices requiring significant GPU memory, a practical constraint that limits resolution and document length.

- Fine-tuned Transformers exhibit sharp performance degradation under domain shift. A system fine-tuned on Reddit political content (Fakeddit) loses 8–15 percentage

points when evaluated on Arabic news misinformation [60]. The representations are rich but not domain-invariant.

- Attention maps provide local interpretability but not causal explanation. An attention weight of 0.87 on the word "claimed" does not establish that "claimed" caused the model to predict "fake", it establishes correlation within the training distribution, which may or may not generalize.
- LLM-generated content targets Transformer detectors. Adversarially crafted text optimized to suppress Transformer-detectable perplexity patterns evades fine-tuned BERT and RoBERTa classifiers with high success rates [71], a vulnerability that scales with the sophistication of the generating model.

1.14 Large Language Models (LLMs) Integration (2023–2025)

1.14.1 Definition and Overview

GPT-4 [72] set the baseline for multimodal language model reasoning. A single architecture, trained at scale on text and image data together, can answer visual questions, describe image content, and check cross-modal consistency without task-specific fine-tuning. Its context window holds a full news article alongside its associated image, and its response to "does this photograph depict what this headline claims?" draws on factual knowledge built from hundreds of billions of training tokens. The process goes beyond simple feature similarity; it amounts to knowledge-grounded verification.

LLaMA 3 (Meta AI, 2024) offered models at 8B and 70B parameter scales under an open research licence. Researchers can now fine-tune and deploy powerful models on their own datasets without API costs or rate limits. LLaVA-1.6 [73] built on LLaMA-3's language backbone with improved visual encoding, giving the open-source community a strong multimodal baseline. Gemini 1.5 (Google DeepMind, 2024) pushed context length to 1 million tokens, making it possible to feed an entire news report (including article text, image descriptions, source metadata, and related documents) in a single call. Together, these models represent a step change in what detection systems can do.

1.14.2 LLM Techniques for Fake News Detection

Zero-shot and Few-shot Learning

Zero-shot fake news detection with LLMs operates through a simple mechanism: the model has internalized world knowledge during pre-training; the prompt activates that knowledge against the specific claim under evaluation. Beigi et al. [74] prompted GPT-4

with structured verification questions, such as "What specific claim does this headline make? What factual evidence would confirm or refute it? Is the accompanying image consistent with the claimed event?", and obtained 78% accuracy on FakeNewsNet without any labeled training examples. A fine-tuned BERT baseline trained on 5,000 labeled examples from the same distribution achieved 72.3%.

Few-shot prompting provides the model with k labeled examples as in-context demonstrations, allowing it to infer task-specific label conventions and edge-case handling from the examples themselves. GLPN-LLM [75] took this further: use LLM-generated pseudo-labels to bootstrap a semi-supervised label propagation process over news propagation graphs, replacing the need for manual annotation with LLM inference at scale. The propagation mechanism then refines predictions by enforcing consistency among topologically similar nodes. Ablation showed that detailed prompts, which specify the reasoning chain expected of the model, improved accuracy by 5.2 percentage points over minimal prompts on the same benchmark.

Chain-of-Thought (CoT) Reasoning

Chain-of-thought prompting [76] elicits multi-step inference by including examples that show explicit reasoning steps before the final answer. For fake news detection, a CoT prompt might demonstrate: identify the factual claim, then retrieve relevant background knowledge, then assess image-text semantic overlap, then evaluate source credibility, then conclude with a veracity judgment and confidence score. The model learns to generate this reasoning chain before producing its classification output. The chain is not cosmetic: removing it and asking for the classification directly reduces accuracy by 6–8 percentage points on knowledge-intensive benchmarks [76].

Zheng et al. [77] extended CoT to adversarial debate: two LLM agents argue opposing positions on a claim's veracity (one building the prosecution case and the other the defense) while a neutral judge LLM evaluates both arguments and renders a verdict. The debate structure forces the model to generate and then rebut the strongest counterarguments, surfacing evidence that a single-agent CoT chain might rationalize away. On a multimodal fact-checking benchmark, adversarial debate reduced hallucinated justifications by 31% relative to standard CoT while maintaining comparable classification accuracy.

1.14.3 Related Works Using LLMs for Fake News Detection

2024–2025 State-of-the-Art Systems

FND-LLM [66] builds a detection pipeline with eight specialized modules: a BERT textual feature branch, a ViT visual semantic branch, an Extended Autocoded ViT (EAViT)

visual tampering branch, a bidirectional co-attention network, a CLIP cross-modal alignment branch, an LLM reasoning branch that generates article-level authenticity rationales, a Mixture-of-Experts (MMoE) feature integration layer, and a final classifier. The explicit division of visual processing into a semantic branch and a tampering branch is architecturally significant: semantic features assess content plausibility while tampering features detect manipulation artifacts. Combining them via MMoE, which dynamically routes features across expert sub-networks, produced 92.1% F1 on the Weibo benchmark.

SNIFFER [78] adapts InstructBLIP for out-of-context misinformation detection through two fine-tuning stages. Stage 1 aligns the Q-Former to news-domain visual concepts (such as entities, locations, and events) that the original model's generic pre-training underrepresents. Stage 2 fine-tunes on OOC-specific instruction data generated by GPT-4 from real NewsCLIPpings examples, teaching the model to produce natural language explanations alongside binary verdicts. External image retrieval augments each inference call, supplying visual evidence of what the queried image actually depicts. SNIFFER surpassed the best prior multimodal LLM baseline by 40+ percentage points on NewsCLIPpings.

MAGIC [46] constructs an information interaction graph where BERT-encoded text representations and ResNet-50 image representations form nodes, and adaptive Graph Attention Network (GAT) edges encode learned cross-modal compatibility weights. Each graph propagation step updates every node by attending to its neighbors, text nodes attend to related image nodes and vice versa, through an attention mechanism parameterized by the node feature vectors themselves. Tested on Fakeddit (English) and MFND (Chinese), MAGIC achieved 98.8% and 86.3% accuracy respectively. The 12.5-point gap between the two languages directly quantifies the cost of training data domain shift between English Reddit content and Chinese news media.

FakeNewsGPT4 [79] targets the domain-shift problem specifically. Traditional detection models fail on emerging fake news because they learn surface statistical patterns from historical training data; GPT-4V carries world knowledge that does not expire with training-set distribution. The framework uses the LVLM's knowledge base to understand the semantic context of a news claim (specifically what event it refers to and its associated factual record) and grounds the detection decision in retrieved world knowledge rather than training-distribution statistics.

HEMT-Fake [61] addresses multilingual multimodal detection by combining Transformer token embeddings, CNN-BiLSTM text encoders, ResNet image features, and GraphSAGE-propagated metadata signals, fused via multi-head attention across modalities. A hierarchical explainability module generates token-level explanations via attention, sentence-level explanations via SHAP, and modality-level explanations via LIME, producing three levels of interpretable output alongside each classification decision. Evaluated on a new 74,000-article multilingual dataset spanning Hindi, Gujarati, Marathi, Telugu, and English, HEMT-Fake demonstrates that cross-lingual detection is achievable

when the architecture explicitly models language-invariant semantic features rather than language-specific surface patterns.

Multimodal & Deepfake Studies (2024–2025)

The 2024 cohort represents the first generation of multimodal detection systems that explicitly reason about cross-modal consistency rather than computing statistical feature overlap. MAGIC's graph-structured interaction propagation, TT-BLIP's tri-modal encoding, FND-LLM's SLM-LLM hybrid reasoning, and SNIFFER's instruction-tuned cross-modal explanation generation each operationalize this reasoning paradigm through different architectural mechanisms, yet they all converge on the same empirical finding: models that compute explicit interactions between specific text tokens and specific image regions outperform models that fuse modality-level summary vectors. The interaction granularity is the key variable.

The systematic review by Nasser et al. [5], covering 121 studies from 2018–2025, confirms Transformer dominance across modalities and notes that 2023–2024 recorded the highest publication volumes in the field's history. Graph-based methods and LLM-augmented architectures show the strongest performance trends in the most recent cohort. Deepfake detection specifically has shifted from artifact-classification to semantic-provenance reasoning [56], reflecting the reality that 2024-era generation quality has outpaced artifact-level detection capability.

Limitations of LLM-Based Approaches:

- LLMs hallucinate. When asked to justify a veracity verdict, they will generate syntactically fluent, semantically plausible reasoning chains that are factually wrong, and will do so with the same surface confidence as correct explanations. Ji et al. [80] document this systematically across verification tasks; no reliable detection method for LLM hallucination in open-ended generation currently exists.

- LLMs are statically frozen at their training cutoff. A breaking news event that occurred after GPT-4's pretraining cannot be verified from internal knowledge alone; retrieval augmentation is necessary but adds latency and introduces retrieval-quality dependencies.

- GPT-4V inference costs approximately 10–50 more compute than a fine-tuned BERT forward pass. Real-time detection at social media platform scale (processing millions of posts per hour) is economically infeasible with current MLLM inference costs.

- LLM-generated misinformation specifically evades LLM detectors. Wu et al. [71] demonstrate that text generated by GPT-3.5 to mimic credible news style passes GPT-4-based detection at significantly higher rates than human-authored fake news,

because both the generator and detector share overlapping representations of "what credible writing looks like."

1.15 Theoretical Background

1.15.1 Transformer Architecture and Attention Mechanism

Previously, researchers relied on recurrent neural networks (RNNs) and long short-term memory (LSTM) networks to analyze sequences, These networks process elements sequentially, which leads to a bottleneck when capturing long-term dependencies.In order to address this problem, Vaswani et al. [51] introduced a framework called the transformer, which has become the primary standard in sequence processing because it processes all elements of the sequence simultaneously through a self-attention mechanism, and this parallel processing allows it to capture long-term dependencies across the sequence more effectively.

Scaled Dot-Product Attention

The core operation of a Transformer is the scaled dot-product attention, defined as:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V} \tag{1.3}$$

where $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are the query, key, and value matrices respectively, and d_k is the dimensionality of the key vectors. To mitigate the issue of vanishing gradients during model training, the scaling factor $\frac{1}{\sqrt{d_k}}$ is utilized. This prevents the dot products from growing excessively large in higher dimensions, which would otherwise push the softmax function into regions with extremely small gradients.

Softmax Function

The softmax function converts a vector of raw scores (logits) into a probability distribution:

$$\text{softmax}(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}} \tag{1.4}$$

It ensures that all output values lie in the range $(0, 1)$ and sum to 1, making them interpretable as probabilities or normalized weights.

Multi-Head Attention

Instead of performing a single attention operation, the Transformer partitions the queries, keys, and values into multiple distinct subspaces. This design allows the network to use h parallel attention heads at the same time, as each head learns independently to focus on different features within the input sequence. The formula for multi-head attention is:

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\mathbf{W}^O \tag{1.5}$$

where each individual component is formulated as

$$\text{head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^Q,\, \mathbf{K}\mathbf{W}_i^K,\, \mathbf{V}\mathbf{W}_i^V) \tag{1.6}$$

utilizing its own learned projection matrices.

Layer Normalization

Layer normalization (LayerNorm) [81] works on the consistency of values, preventing gradients from exploding or fading, by normalizing activations across the feature dimension for each sample independently:

$$\text{LayerNorm}(\mathbf{x}) = \frac{\mathbf{x} - \mu}{\sqrt{\sigma^2 + \epsilon}} \odot \boldsymbol{\gamma} + \boldsymbol{\beta} \tag{1.7}$$

where μ and σ^2 are the mean and variance of $\mathbf{x}$, and $\boldsymbol{\gamma}$, $\boldsymbol{\beta}$ are learnable scale and shift parameters.

1.15.2 Long-Document Text Encoding

Longformer

Longformer [82] is a transformer model designed for long documents. It solves the memory problem by using two types of attention. The first is sliding-window attention: each token attends only to a fixed number of nearby tokens. This has a cost that grows linearly with the sequence length. The second is global attention, applied to a small set of selected tokens. A token with global attention can attend to every other token in the sequence. Every other token can also attend to it.

The special token placed at the start of every sequence, called the `[CLS]` token, is given global attention. Its final hidden state summarises the entire sequence. This vector is used as the text representation in downstream tasks. Longformer can process sequences of up to 4096 tokens, which is much longer than what standard transformers can handle efficiently.

1.15.3 Pre-trained Language and Vision Models

Text Modeling: RoBERTa

RoBERTa (Robustly Optimized BERT Approach) is an enhanced version of BERT proposed by Liu et al. [59], which improves pre-training by applying dynamic masking where masked tokens change across training epochs. The model was also trained in larger batches using a huge dataset of English text, allowing it to understand the deep semantic meaning of the content.

Vision-Language Modeling: SigLIP

SigLIP (Sigmoid Loss for Language-Image Pre-Training),proposed by Zhai et al. [83],is a visual language model specially trained to compare images and text that uses the sigmoid function,where the model evaluates each pair of images and text separately,unlike CLIP [54] which uses a softmax-based contrastive loss whose denominator requires an aggregate summation across all negative pairs in the batch. This mechanism allows for a significant reduction in computational expenses and taking advantage of huge batch sizes during training, producing highly accurate representations for visual understanding:

$$\mathcal{L}_{\text{SigLIP}} = -\frac{1}{N}\sum_{i,j}\log\sigma(z_{ij}\cdot y_{ij}) \tag{1.8}$$

where z_{ij} is the similarity score between image i and text j, $y_{ij} \in \{-1, +1\}$ is the label, and σ is the sigmoid function.

Speech Representation: Wav2Vec 2.0

Wav2Vec 2.0 [84] is a speech representation model developed by Meta AI, trained on 960 hours of the LibriSpeech dataset [85]. It learns speech representations directly from raw audio through self-supervised training, enabling it to capture and represent complex temporal patterns.

Generative Reasoning: Phi-3 Mini

Phi-3 Mini is a large language model (LLM) developed by Microsoft [86], containing 3.8 billion parameters. It is based on a decoder-only Transformer architecture trained on high-quality curated datasets, which delivers strong reasoning performance while maintaining efficiency during inference.

1.15.4 Visual Representation Learning

Contrastive Vision–Language Pre-training

A visual encoder is a neural network that takes an image as input and outputs a numerical vector representing its meaning in a compact form. Modern visual encoders learn from very large datasets of image–text pairs, producing similar vectors for images and text that describe the same thing while keeping unrelated pairs far apart [54].

The underlying training approach, contrastive learning, uses a loss function that pulls matching pairs closer in vector space and pushes non-matching pairs apart. CLIP [54] is among the most widely used models trained in this way, applying a softmax function over all pairs in a batch to compute the training signal.

1.15.5 Image Captioning

Image captioning is the task of automatically generating a natural-language sentence that describes an image. Early methods used convolutional networks to encode the image and recurrent networks to generate the text one word at a time [87]. Modern methods use large pre-trained models and produce more detailed and accurate descriptions.

BLIP-2

BLIP-2 [63] is a state-of-the-art image captioning model. It connects a frozen image encoder with a frozen large language model using a small trainable module called the Querying Transformer, or Q-Former. The Q-Former learns to extract the most relevant visual information from the image encoder. It then passes this information to the language model. The language model generates a natural-language description of the image word by word.

1.15.6 Large Language Models for Reasoning

Autoregressive Language Models

A large language model (LLM) is a neural network trained to predict the next word in a text sequence. During training, the model reads billions of documents. It learns grammar, facts, reasoning patterns, and general knowledge. At inference time, the model generates text one token at a time [17].

Modern LLMs can also follow instructions written in natural language. Given a prompt describing a task, the model generates a fitting response — even for tasks it was never explicitly trained on, such as fact-checking or evaluating whether two pieces of information agree [88].

Llama-3

Llama-3 [89] is an open-weight LLM developed by Meta AI. The instruction-following variant with 8 billion parameters is used in this work. It is trained to follow structured prompts and produce well-formatted responses.

LLMs of this size require significant memory. To reduce this requirement, quantisation is applied. The NF4 method [90] represents each weight using 4 bits instead of 16. This reduces memory by approximately 75% with only a small reduction in accuracy. Dettmers et al. [90] showed that NF4 quantisation preserves more than 99% of the original model's performance on downstream tasks.

1.15.7 Frequency-Domain Image Analysis

The Discrete Cosine Transform

The Discrete Cosine Transform (DCT) is a mathematical operation that converts a signal from the spatial domain to the frequency domain [91]. When applied to an image, the DCT expresses the image as a sum of cosine waves of different frequencies. Coefficients in the low-frequency region describe broad structures such as large shapes. Coefficients in the high-frequency region describe fine details such as sharp edges and textures.

Image manipulation often changes high-frequency content in ways that are invisible to the human eye. When two image regions are combined (a technique called splicing), the boundary between them creates unusual high-frequency patterns. These patterns can be detected by analysing the DCT coefficients [92]. Yao et al. [92] demonstrated that noise level inconsistencies in the DCT domain reliably identify spliced regions.

Fast Fourier Transform

The Fast Fourier Transform (FFT) is another method for converting an image to the frequency domain. It produces a spectrum that shows how much of each frequency is present in the image. Shifting the zero-frequency component to the centre of the spectrum makes it easier to interpret. A radial profile is then computed by averaging the spectrum values at each distance from the centre.

Images generated by deep learning models, such as generative adversarial networks (GANs), often show distinctive peaks in the radial FFT profile. These peaks come from the regular structure of the convolutional operations used inside the generator. Frank et al. [93] showed that these spectral patterns are consistent across different GAN architectures and can serve as a reliable detection signal. Detecting these peaks can identify AI-generated images.

1.15.8 Face Detection

Haar Cascade Classifier

Haar Cascade [94] is a machine learning algorithm used to detect objects such as faces in images.The process goes through four main stages:

1. Haar features are extracted by calculating the differences in light intensity between adjacent rectangular regions. This produces edge features, line features, and diagonal features.

2. The original image is converted into an integral image to make the calculations easier and faster.

3. Because of the very large number of extracted features, the Adaptive Boosting (AdaBoost) algorithm is applied to ignore weak features. AdaBoost builds weak classifiers, where each classifier $h_j(x)$ depends on only one feature $f_j(x)$, a threshold θ_j, and a polarity p_j:
$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$
.

4. The candidate window passes through a hierarchical structure called the cascade. If the window successfully passes all stages of the cascade, it is finally classified as a detected object. This design ensures high accuracy and very fast processing.

1.15.9 Activation and Output Functions

Sigmoid Function

The sigmoid function, originally introduced by Pierre-François Verhulst [95] in the field of population and biological studies, was later adopted in early neural network research.It is a non-linear activation function widely used in tasks that require interpreting outputs as probabilities, such as binary classification, because it takes values and maps them into a range between 0 and 1.
$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{1.9}$$
where x represents the input to the neuron, and e is the base of the natural logarithm.

One of its biggest problems is the vanishing gradient issue: when the input x becomes very large, either positive or negative, the outputs approach 1 or 0. In these saturation regions, the derivative is close to zero, which makes weight updates almost negligible during the backpropagation process.

ReLU Activation

The Rectified Linear Unit (ReLU) activation function was introduced by Nair & Hinton [96] as a solution to the vanishing gradient problem that affects non-linear saturating functions such as sigmoid and tanh. The mechanism passes the signal as it is when the input is positive and completely stops it (sets it to zero) when the input is negative. This speeds up the training process and makes it more efficient.

$$\text{ReLU}(x) = \max(0, x) \tag{1.10}$$

where x represents the input to the neuron and $\text{ReLU}(x)$ is the output after activation.

GELU Activation

The Gaussian Error Linear Unit (GELU) [97] is an activation function used in modern transformer-based models. It applies a smooth, non-linear transformation to each value. Hendrycks and Gimpel [97] showed that it outperforms the older ReLU activation in many natural language processing tasks.

1.15.10 Optimization and Network Regularization

Batch Normalization

Batch Normalization was introduced by Ioffe & Szegedy [98]. This technique was proposed to solve the problem of Internal Covariate Shift, which refers to the changes in the distribution of input data for each layer, making training more complex and slower. Batch Normalization works by normalizing the outputs of layers for each mini-batch of data, and it is carried out in four steps:

1. **Mini-batch Mean:** The mean of the input values for the current batch $\mathcal{B}$, which contains m samples, is calculated:

$$\mu_{\mathcal{B}} = \frac{1}{m} \sum_{i=1}^{m} x_i \tag{1.11}$$

   where x_i represents the input value for sample number i in this batch.

2. **Calculate Mini-batch Variance:**

$$\sigma_{\mathcal{B}}^2 = \frac{1}{m} \sum_{i=1}^{m} (x_i - \mu_{\mathcal{B}})^2 \tag{1.12}$$

   where $\sigma_{\mathcal{B}}^2$ denotes the empirical variance of the current mini-batch.

3. **Normalization application:**

$$\hat{x}_i = \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \tag{1.13}$$

where ϵ represents a very small constant number.

4. **Scale and Shift:**

$$y_i = \gamma \hat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \tag{1.14}$$

where γ represents scale parameter and β represents shift parameter

Dropout Regularization

Dropout is a technique introduced by Srivastava et al. [99]. It works to prevent overfitting in deep neural networks. Its mechanism is based on randomly identifying some neural units at a certain rate to be temporarily deactivated at each training step, where the model is forced to learn from different units during training. This avoids memorizing data, reduces overfitting, and makes the model more capable of dealing with new data. This mechanism can be expressed as follows:

$$\begin{aligned} r_j &\sim \mathrm{Bernoulli}(p) \\ \tilde{h} &= r \odot h \end{aligned} \tag{1.15}$$

where h represents the original input vector, and r_j denotes the Bernoulli random variable,and p is the activation probability rate,and $\odot$ signifies the element-wise multiplication,and $\tilde{h}$ represents the final regularized vector.

1.15.11 Other Neural Network Building Blocks

Residual Connections

A residual connection, introduced in ResNet [45], adds the input of a layer directly to its output. If a layer computes a transformation $\mathcal{F}(\mathbf{x})$, the output is $\mathbf{x} + \mathcal{F}(\mathbf{x})$. This design allows the gradient to flow back through the network without being reduced at each layer. It makes it possible to train deeper networks without the problem of vanishing gradients.

Attention Pooling

Attention pooling [50] compresses a sequence of vectors into a single vector by learning a scalar importance score for each position. After normalising these scores with softmax

(so they sum to one), the output is computed as the weighted sum of the input vectors, letting the model concentrate on the most useful parts of the sequence.

Convolutional Filters

A one-dimensional convolutional filter slides over a sequence of vectors, multiplying the input values by learned weights and summing the result at each position. Filters of different sizes capture patterns at different scales — small filters pick up local patterns, while larger ones capture wider patterns. Max pooling then selects the strongest activation produced by each filter across all positions, keeping the most prominent signal regardless of where it appears in the sequence [43].

1.15.12 Feature Conditioning Mechanisms

Feature-wise Linear Modulation (FiLM)

Feature-wise Linear Modulation (FiLM), introduced by Perez et al. [100], is a conditioning mechanism that allows one neural network to influence the computation of another by predicting scale and shift parameters. Given a conditioning input $\mathbf{c}$ and a target feature vector $\mathbf{z}$, FiLM generates modulation parameters through a learned function and applies them via an affine transformation:

$$\tilde{\mathbf{z}} = (1 + \boldsymbol{\gamma}) \odot \mathbf{z} + \boldsymbol{\beta} \tag{1.16}$$

where $[\boldsymbol{\gamma}, \boldsymbol{\beta}] = f(\mathbf{c})$ and $\odot$ denotes element-wise multiplication. This mechanism allows the conditioning signal to selectively amplify or suppress individual feature dimensions, providing a more expressive form of feature interaction than simple concatenation [100].

1.15.13 Cross-Modal Interaction Mechanisms

Inverse Attention

Standard attention captures consistency (agreement) between input elements. In tasks where detecting contradictions is equally important, Zhang et al. [101] proposed the Inverse Attention mechanism. Given attention scores $\mathbf{S}$, consistency and inconsistency representations are computed as:

$$\mathbf{A}_{\text{con}} = \text{softmax}(\mathbf{S}), \quad \mathbf{R}_{\text{con}} = \mathbf{A}_{\text{con}}\mathbf{V} \tag{1.17}$$

$$\mathbf{A}_{\text{inv}} = \text{softmax}(\mathbf{A} - \mathbf{S}), \quad \mathbf{R}_{\text{inv}} = \mathbf{A}_{\text{inv}}\mathbf{V} \tag{1.18}$$

where $\mathbf{A}$ is a learnable scalar matrix initialised to 1.0. A sigmoid-gated fusion combines the two streams:

$$\mathbf{g} = \sigma(\mathbf{W}_g[\mathbf{R}_{\text{con}}; \mathbf{R}_{\text{inv}}] + \mathbf{b}_g) \tag{1.19}$$

$$\mathbf{o} = \mathbf{g} \odot \mathbf{R}_{\text{con}} + (1 - \mathbf{g}) \odot \mathbf{R}_{\text{inv}} \tag{1.20}$$

This allows the model to attend to both agreements and disagreements between inputs, which is valuable in verification tasks where inconsistencies are the primary detection signal [101].

Mixture-of-Experts (MoE)

The Mixture-of-Experts (MoE) architecture, proposed by Shazeer et al. [102], uses multiple specialised sub-networks (experts) and a gating mechanism that routes each input to the most relevant experts. For a set of N experts, the output is:

$$\mathbf{y} = \sum_{i=1}^{N} g_i(\mathbf{x}) \cdot \text{Expert}_i(\mathbf{x}) \tag{1.21}$$

where $g_i(\mathbf{x})$ is the gating weight for expert i, computed by a learned gating network. The sparse gating design activates only a subset of experts per input, reducing computation while allowing each expert to specialise on different input patterns. Combining a sparse expert path with a dense residual path improves training stability [102].

1.15.14 Loss Functions

Focal Loss

Focal Loss, introduced by Lin et al. [103], addresses class imbalance by down-weighting well-classified examples and focusing the training signal on hard, misclassified cases:

$$\mathcal{L}_{\text{Focal}} = -\alpha(1 - p_t)^{\gamma} \log(p_t) \tag{1.22}$$

where p_t is the model's estimated probability for the ground-truth class, γ is the focusing parameter, and α is a class-balancing parameter. When $\gamma = 0$, Focal Loss reduces to standard cross-entropy.

InfoNCE Contrastive Loss

The InfoNCE loss, introduced by Oord et al. [104], is a contrastive objective that maximises mutual information between matching pairs while separating non-matching pairs

in the embedding space. The symmetric form is:

$$\mathcal{L}_{\text{InfoNCE}} = \frac{1}{2}(\mathcal{L}_{a\to b} + \mathcal{L}_{b\to a}) \tag{1.23}$$

This objective is widely used for aligning representations across different modalities [104].

Sinkhorn Optimal Transport Loss

The Sinkhorn algorithm, introduced by Cuturi [105], computes an approximate optimal transport plan between two distributions efficiently. The transport loss minimises the Wasserstein distance:

$$\mathcal{L}_{\text{OT}} = \frac{1}{B}\sum_{i,j} P_{ij} C_{ij}^{\text{dist}} \tag{1.24}$$

where $\mathbf{C}^{\text{dist}}$ is the pairwise distance matrix, $\mathbf{P}$ is the soft transport plan computed via iterative Sinkhorn normalisation, and B is the batch size. Unlike pairwise contrastive losses, this loss aligns the global distribution geometry across an entire batch [105].

Weighted Cross-Entropy Loss

Weighted Cross-Entropy extends the standard cross-entropy loss by assigning different penalty weights to each class:

$$\mathcal{L}_{\text{WCE}} = -\left[w_1 \cdot y \log(p) + w_0 \cdot (1-y)\log(1-p)\right] \tag{1.25}$$

where y is the ground-truth label, p is the predicted probability, and w_0, w_1 are the class weights. Assigning a higher weight to the minority class prevents the model from developing a bias toward the majority class.

1.15.15 Similarity and Distance Measures

Bhattacharyya Distance

The Bhattacharyya distance measures the dissimilarity between two probability distributions. For two normalised histograms h_1 and h_2:

$$d_B(h_1, h_2) = -\ln\left(\sum_k \sqrt{h_1(k)\cdot h_2(k)}\right) \tag{1.26}$$

The inner sum is the Bhattacharyya coefficient, which equals 1 for identical distributions and 0 for non-overlapping ones. The negative logarithm converts this into a distance where larger values indicate greater dissimilarity.

Jaccard Similarity

The Jaccard similarity coefficient measures the overlap between two sets:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{1.27}$$

The coefficient ranges from 0 (no shared elements) to 1 (identical sets). It is commonly used to measure the topical overlap between two pieces of text by treating each as a set of words.

1.15.16 Training and Optimization Techniques

AdamW Optimizer

AdamW, proposed by Loshchilov and Hutter [106], is a variant of the Adam optimizer that decouples weight decay from the gradient-based update. Standard Adam applies weight decay through the gradient, which interacts poorly with the adaptive learning rate. AdamW applies weight decay directly to the parameters after the gradient step, resulting in more consistent regularisation and improved generalisation.

Cosine Annealing Scheduler

Cosine Annealing [107] smoothly decreases the learning rate following a cosine curve:

$$\eta_t = \frac{\eta_0}{2}\left(1 + \cos\left(\frac{t\pi}{T}\right)\right) \tag{1.28}$$

where η_0 is the initial learning rate, t is the current epoch, and T is the total number of epochs. This allows the model to escape local minima early in training and fine-tune precisely during later epochs.

Gradient Clipping

Gradient clipping [108] constrains the gradient norm during training. If the norm exceeds a threshold θ, the gradient is rescaled:

$$\mathbf{g} \leftarrow \frac{\theta}{\|\mathbf{g}\|}\mathbf{g} \quad \text{if } \|\mathbf{g}\| > \theta \tag{1.29}$$

This prevents exploding gradients, which can destabilise training in deep networks.

Early Stopping

Early stopping monitors a validation metric during training and terminates the process when the metric stops improving for a specified number of consecutive epochs (the pa-

tience). This prevents overfitting by selecting the checkpoint with the best validation performance.

1.16 Performance Comparison

Table 1.1 provides a comprehensive overview of the state-of-the-art fake news detection systems developed between 2017 and 2025. The data illustrates the evolutionary trajectory of the field: moving from early text-only machine learning baselines (e.g., ISOT, LIAR) to modern, highly complex multimodal systems that integrate visual, textual, and acoustic data. A clear trend emerges demonstrating that while early text-based models struggled with robust generalisation, the adoption of cross-modal attention mechanisms and LLM-based reasoning (such as in MAGIC and FND-LLM) consistently yields accuracy scores exceeding 90% on challenging multimodal datasets like Fakeddit and NewsCLIPpings.

Table 1.1: Performance comparison of fake news detection systems.

Author / Reference	Year	Modality	Model / Method	Dataset	F1 (%)	Acc. (%)
Ahmed et al.	2017	Text	SVM + TF-IDF	ISOT	95.8	96.1
Wang	2017	Text	Bi-LSTM / SVM	LIAR	27.0	27.4
Popat et al.	2018	Text + Web	Bi-LSTM + Attn. (DeClarE)	Multi	76.1	76.4
Khattar et al.	2019	Text + Image	MVAE	MediaEval	74.6	74.9
Qian et al.	2019	Text + Social	GAN + Hierarchical Attn.	Twitter16	87.7	87.9
Zhou et al.	2020	Text	RoBERTa + KG Embeddings	LIAR	71.8	72.3
Nakamura et al.	2020	Text + Image	BERT + ResNet-50	Fakeddit	88.3	88.7
Wu et al. (MCAN)	2021	Text + Image	BERT + ResNet + Co-Attn.	Weibo	89.2	89.6
Qi et al.	2021	Text + Image	BERT + CLIP Consistency	NewsCLIPpings	86.4	87.0
Papadopoulos et al.	2023	Text + Image	GPT-4V + CoT (VERITE)	VERITE	88.9	89.3
Sharma et al. (GAME-ON)	2024	Text + Image	GNN Cross-Modal Fusion	Fakeddit / PolitiFact	89.6	90.1
Choi & Kim (TT-BLIP)	2024	Text + Image	BLIP + Tri-Transformer	Weibo / GossipCop	91.8	92.3
Wang et al. (FND-LLM)	2024	Text + Image	BERT+ViT+CLIP+Co-Attn+LLM	Weibo / Twitter	92.1	92.7
Liu et al. (FakeNewsGPT4)	2024	Text + Image	GPT-4V + Knowledge Aug.	Multi (MM)	91.5	92.0
Qi et al. (SNIFFER)	2024	Text + Image	InstructBLIP + Q-Former	NewsCLIPpings	90.7	91.2
Xu et al. (MAGIC)	2024	Text + Image	BERT + ResNet50 + Adaptive GAT	Fakeddit / MFND	97.4	98.8
Hu et al. (GLPN-LLM)	2025	Text + Image	LLM Pseudo-Labels + GLP	Multi Benchmarks	91.2	91.6
Nasser et al. (HEMT-Fake)	2025	Multimodal + Graph	Transformer+CNN-BiLSTM+GraphSAGE	Multi (ML)	90.3	90.8

1.17 Conclusion

This chapter has provided a comprehensive overview of the current state of the art in fake news and multimedia deepfake detection. Part I traced the foundational definition of fake news and its distinct categories—misinformation, disinformation, and malinformation—while highlighting the critical shift from simple text-based claims to complex, highly convincing multimodal formats. The severe societal consequences of this phenomenon were examined, illustrating its adverse impacts across public health, democratic processes, social stability, and financial markets. Furthermore, the core approaches to detection, categorized into content-based and context-based methods, were reviewed alongside their fundamental limitations.

Part II surveyed the technological trajectory of the field from 2017 to 2025, detailing how feature engineering gave way to learned representations, unimodal text classifiers evolved into cross-modal architectures, and task-specific fine-tuning transitioned to foundation model adaptation. This evolution has culminated in the highly advanced systems of 2024 and 2025, such as MAGIC, TT-BLIP, FND-LLM, and SNIFFER, which leverage multimodal fusion, instruction tuning, and large language models (LLMs) to achieve unprecedented performance.

Despite these advancements, significant research gaps remain open. Specifically, cross-domain and cross-language generalization continue to be primary failure modes; robustness against adversarially crafted LLM-generated content remains largely unaddressed; and fine-grained, modality-level explainability is missing from most current architectures. To address these challenges, Chapter 2 describes our proposed architectures—a multimodal image-text verification system utilizing deep learning, frequency-domain analysis, and LLM reasoning, alongside a video-based deepfake detection pipeline. Chapter 3 subsequently presents the experimental environment, evaluation metrics, and empirical results validating the effectiveness of these proposed systems.

Chapter 2

Proposed Approaches

2.1 Introduction

Fake news has become a major societal challenge in the digital age. False information spreads rapidly on social media platforms and can cause real harm to individuals, institutions, and public discourse [6, 1]. While early research focused primarily on textual analysis, modern fake news increasingly combines text with images or videos to enhance credibility and reach wider audiences. This phenomenon is known as multimodal fake news [109, 110].

Detecting multimodal fake news is particularly challenging because deception manifests in multiple forms: a real image may be placed in a completely false context, a caption may be fabricated to accompany an unrelated image, or the visual content itself may be digitally manipulated using AI or deepfake techniques [93, 111]. Addressing these threats requires a system that jointly analyses the semantic content of text, the visual properties of images, and the logical consistency between modalities [112, 113].

To handle these distinct threat types, this chapter presents two complementary detection approaches, each tailored to a specific input modality, as illustrated in Figure 2.1.

- **Approach 1 — Image + Text:** A five-stream fusion pipeline that analyses static image–caption pairs from the **Fakeddit** dataset through visual semantic encoding, frequency-domain forensics, text encoding, and LLM-based cross-modal reasoning, fused via a Mixture-of-Experts (MoE) and Co-Attention mechanism.
- **Approach 2 — Video + Metadata:** A multimodal pipeline that processes dynamic video content from the **FakeTT** dataset by integrating keyframe visual features, audio representations, transcribed speech, and structured metadata through a cross-attention fusion architecture.

Both approaches share a common LLM-based reasoning module for cross-modal analysis and converge to a binary classification output: *Real* or *Fake*.

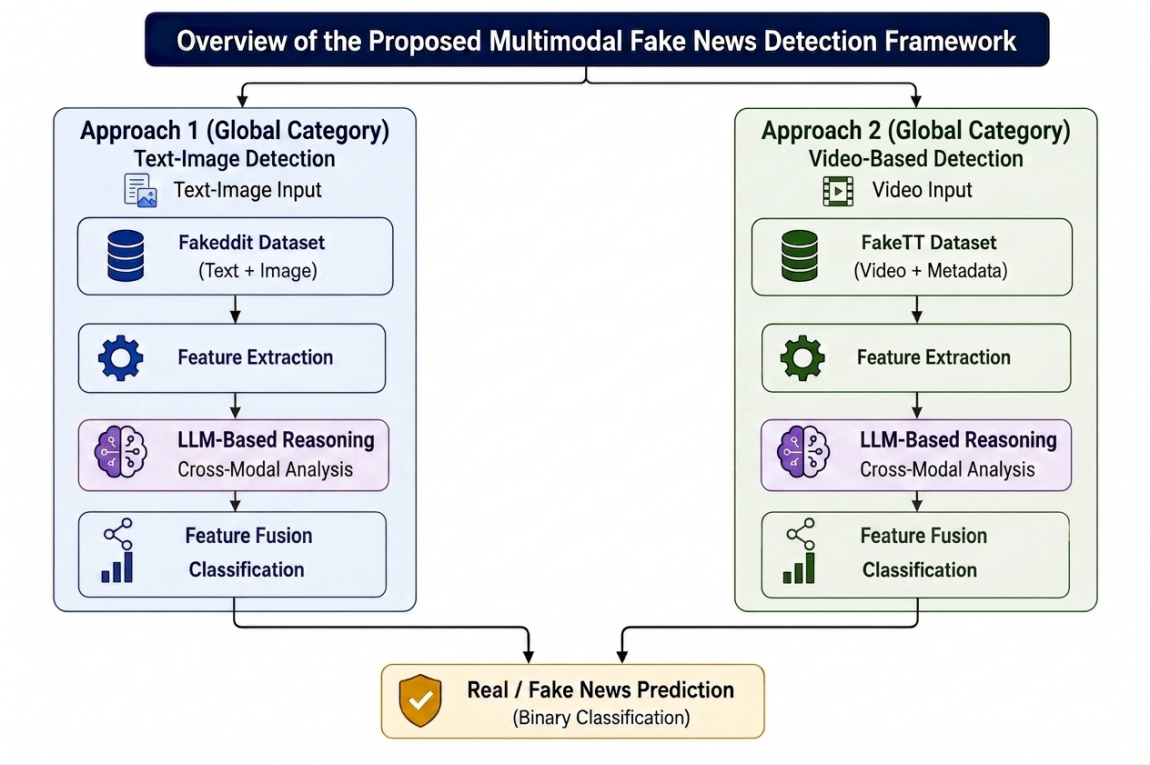


Figure 2.1: Overview of the proposed multimodal fake news detection framework.

As depicted in Figure 2.1, the framework employs a modular design to process different modalities independently before fusing them for a final veracity prediction. For image–text pairs, visual, textual, and frequency features are extracted and explicitly fed into an LLM. The LLM acts as an intermediate reasoning agent, generating a structured consistency analysis. These outputs are subsequently fused via a cross-attention network. Similarly, for short videos, sequential keyframes, audio tracks, and textual metadata are independently encoded and cross-referenced by a specialized LLM. This high-level separation ensures that the system can flexibly handle distinct modalities while maintaining a unified detection objective.

The remainder of this chapter is organised as follows: Section 2.2 details the architecture of Approach 1, explaining the image-text feature extraction and fusion pipeline. Section 2.3 describes the architecture of Approach 2, focusing on the video-based multimodal framework.

Part I: Approach 1 — Image-Text Detection

2.2 Approach 1 : Image + Text

2.2.1 Feature Extraction

The feature extraction stage converts a raw image–caption pair into five numerical vectors. Each vector captures a different type of information about the sample. Together, they provide a complete description that classification models can use to detect fake news.

Figure 2.2 shows how the raw image and caption are transformed into these five feature vectors. An image and a text caption are processed in parallel. The image passes through three branches: the SigLIP encoder (visual features), the DCT and FFT analysis (frequency features), and the BLIP-2 captioner. The generated image description is then combined with the caption and sent to the Llama-3 reasoning model, which produces the LLM scalar and reasoning features. The caption alone is encoded by Longformer to produce the text features.

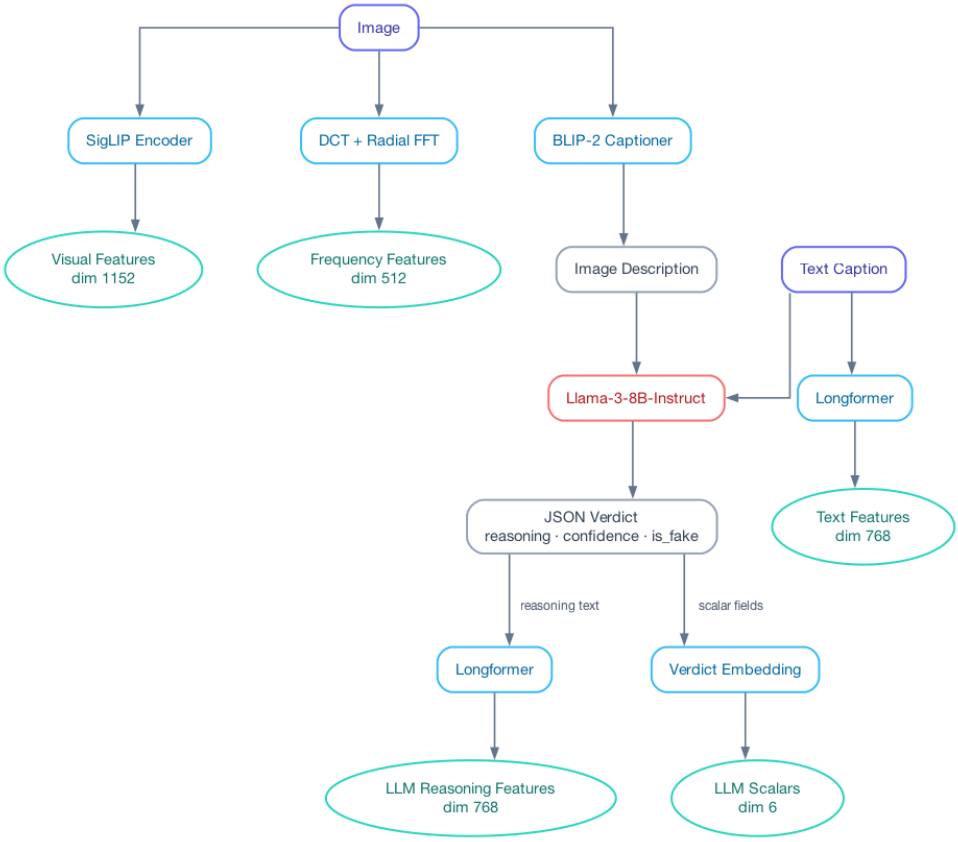


Figure 2.2: Feature Extraction

2.2.2 Visual Feature Extraction

The goal of this step is to represent the semantic content of the image as a single numerical vector. The model used is SigLIP [83], with the `so400m-patch14-384` checkpoint. This model processes images at a resolution of 384384 pixels and divides them into patches of 1414 pixels. The 384384 resolution was chosen over the standard 224224 because the higher resolution preserves fine-grained visual details and background contexts, which are crucial for identifying subtle manipulations. Furthermore, the 1414 patch size divides the image into a dense grid of 784 patches, providing the transformer with a much more granular view of the image compared to the standard 1616 patches. Each patch is encoded by the transformer, and the outputs are combined into a single 1152-dimensional pooled vector.

The resulting vector is normalised to unit length. After normalisation, every visual vector lies on the surface of a unit hypersphere. Two vectors can then be compared using cosine similarity, which measures the angle between them [114]. This comparison is meaningful for detecting whether two images are semantically related.

SigLIP was preferred over CLIP for two practical reasons identified by Zhai et al. [83]. The sigmoid loss evaluates each image–text pair independently, avoiding the memory bottleneck that CLIP's softmax denominator creates during large-batch training. This independent evaluation also yields higher-quality representations on downstream tasks, particularly in vision–language alignment settings. This alignment proves essential for fake news detection, as the relationship between an image and its caption forms the primary detection signal [54].

2.2.3 Image Captioning

The reasoning model used later in the pipeline receives text as input. It cannot process the image directly. This step generates a natural-language description of the image so that the reasoning model can understand its content.

BLIP-2 [63] generates the caption. Given the image, it produces a natural-language sentence using beam search with four beams. Beam search keeps track of the most probable partial outputs at each step, exploring several continuations in parallel. The result is more fluent and accurate than greedy decoding, which simply picks the most probable next token at every step.

BLIP-2 was chosen because it delivers detailed, semantically rich captions at low computational cost. Training only the Q-Former bridge module while keeping both the image encoder and the language model frozen requires far less data and computing power than fully fine-tuned captioning models [63]. The output — a complete sentence describing what the image shows — is passed directly to the reasoning model as the image description.

2.2.4 Text Feature Extraction

Text features represent the semantic content of the news caption. The model used is Longformer [82], with the `longformer-base-4096` checkpoint. The caption is tokenised and encoded. Global attention is applied to the `[CLS]` token. Every other token receives only local window attention.

The final hidden state of the `[CLS]` token is extracted as the text feature vector. This 768-dimensional vector captures the overall meaning of the caption. It is then normalised to unit length, in the same way as the visual feature vector.

Longformer was preferred over standard BERT to handle the significant variation in news caption lengths. The Fakeddit dataset [109] contains everything from brief headlines to extensive paragraphs. Standard transformers truncate inputs exceeding 512 tokens, inevitably causing information loss. Longformer handles sequences of up to 4096 tokens with linear memory cost [82], ensuring the encoder always receives the complete caption text without any truncation.

2.2.5 Frequency-Domain Feature Extraction

The frequency feature extraction step analyses the image at a pixel level. It looks for patterns that may indicate digital manipulation. This analysis is independent of the semantic content of the image. It focuses on low-level statistical properties of the pixel values.

a) DCT High-Frequency Features

The image is converted to greyscale and resized to 256256 pixels. The DCT is then applied to the full image. The resulting two-dimensional matrix of coefficients is arranged so that low frequencies are in the top-left corner and high frequencies are in the top-right corner and lower rows.

The high-frequency region (rows 0 to 63, columns 192 to 255) is extracted from the DCT matrix. Yao et al. [92] showed that this region is most sensitive to manipulation artefacts such as splicing and copy-move, because these operations introduce noise level inconsistencies that are concentrated in the high-frequency coefficients. The values in this region are flattened into a 256-dimensional vector.

b) Radial FFT Features

The FFT is applied to the same greyscale image. The spectrum is shifted so that the zero-frequency component is at the centre. The logarithm of the magnitude is computed. A radial profile is extracted by averaging the magnitude values at each integer distance from the centre.

This profile captures the global frequency structure of the image. It is rotation-invariant, meaning that rotating the image does not change the profile. Frank et al. [93] showed that GAN-generated images produce distinctive peaks in this radial profile that real photographs do not show. These peaks originate from the regular up-sampling operations inside the generator network. The first 256 values of the radial profile are retained.

c) Combination and Normalisation

The 256-dimensional DCT vector and the 256-dimensional radial FFT vector are joined into a single 512-dimensional vector. This combined vector is standardised to have zero mean and unit variance across its dimensions. It is then normalised to unit length.

2.2.6 LLM-Based Reasoning

The reasoning step uses a large language model to compare the news caption with the image description generated in the captioning step. The model produces a structured analysis of whether the two are consistent. This analysis is then converted into two types of features: a small scalar vector and a text embedding.

The decision to use Llama-3 stems from its open-weight licensing, which allows local deployment on research servers without the constraints of paid API services or rate limits. At the same time, the 8-billion-parameter instruction variant delivers robust reasoning capabilities while remaining compact enough to fit within standard GPU memory limits after NF4 quantisation.

a) Prompt Design

The LLM receives a structured prompt. The prompt tells the model to act as a fake news detection expert. It asks the model to compare the news caption with the image description. The model is asked to respond in a specific format that includes five fields:

- A short reasoning explanation in natural language.
- A score between 0 and 1 for how well the caption matches the image description.
- A list of detected manipulation signals. These "signals" are specific semantic traces of deception that the LLM identifies during its cross-referencing. Examples include identifying a pair as "out-of-context" when the image clearly depicts a different event than the caption claims, or noting "splicing" if the text description implies disjointed visual elements.
- A verdict of either REAL or FAKE.
- A confidence score between 0 and 1.

The image description field is filled with the sentence generated by BLIP-2. This directly links the captioning step to the reasoning step.

b) Fallback Response

If the model produces a response that cannot be interpreted, a default response is used instead. The default assigns a match score of 0.5 and a confidence of 0.5. These neutral values avoid introducing a bias in either direction when the response is missing or unclear.

2.2.7 LLM Scalar Feature Vector

The structured response from the LLM is converted into a six-dimensional scalar vector. Each dimension represents one aspect of the model's analysis. This vector is denoted $\mathbf{s} \in \mathbb{R}^6$ and is defined as follows:

$$s_0 = \text{confidence} \in [0, 1] \tag{2.1}$$

$$s_1 = \mathbb{1}[\text{verdict} = \texttt{FAKE}] \in \{0, 1\} \tag{2.2}$$

$$s_2 = \text{caption-image match score} \in [0, 1] \tag{2.3}$$

$$s_3 = \frac{\min(|\text{signals}|,\ 10)}{10} \in [0, 1] \tag{2.4}$$

$$s_4 = \frac{|\{s \in \text{signals} : s \in \mathcal{C}\}|}{|\text{signals}|} \in [0, 1] \tag{2.5}$$

$$s_5 = |s_1 - s_2| \in [0, 1] \tag{2.6}$$

The set $\mathcal{C}$ is the set of known forensic signal categories. It includes: face swap, splicing, copy-move, out of context, lighting inconsistency, metadata mismatch, parse failure, compression artefact, colour mismatch, and text overlay.

The feature s_3 measures how many signals the LLM found, normalised to a maximum of ten. The feature s_4 measures what fraction of the detected signals belong to a known forensic category. A high value means the model identified specific, well-defined types of manipulation. A low value means the signals are vague or do not match any known pattern. The feature s_5 captures the disagreement between the binary verdict and the continuous match score. A large value means these two outputs point in different directions, which may itself be informative.

Each of the six features remains mathematically independent, meaning no single feature can be derived from the others. Because redundant variables add no predictive value to a classifier [115], constructing the six-dimensional vector from distinct signals ensures maximum informational density within a compact representation.

2.2.8 LLM Reasoning Embedding

The reasoning field of the LLM response is a short natural-language explanation containing details that the scalar vector cannot represent — the specific reasons behind the verdict, expressed in free text.

To make this information usable by a classifier, the reasoning text is encoded with the same Longformer model used for the news caption. The `[CLS]` hidden state is extracted and normalised to unit length, producing a 768-dimensional vector called the reasoning embedding. It gives the classifier access to the full semantic content of the explanation rather than only the summary numbers in the scalar vector.

2.2.9 Summary of Extracted Features

Table 2.1 summarises all five feature groups extracted by the proposed pipeline, including their dimensionality, extraction method, and semantic role.

Table 2.1: Summary of Extracted Feature Groups (Approach 1: Image + Text)

Feature Group	**Dim.**	**Extraction Method**	**Semantic Role**
Visual	1152	SigLIP so400m-patch14-384	Encodes pooled global semantic content and vision-language alignment features
Text	768	Longformer base-4096 CLS token	Captures overall meaning of long-range news captions without truncation
Frequency	512	DCT (256) + FFT (256)	Detects low-level pixel manipulation artifacts, splicing, and GAN patterns
LLM Scalars	6	Llama-3-8B structured output	Encodes explicit consistency scores, verification confidence, and forensic signals
LLM Reasoning	768	Longformer on LLM text	Encodes the full dense semantic content of the LLM's natural-language justification
TOTAL	**3206**	—	Complete multimodal feature vector per image-caption pair

2.2.10 Classification Models

This section describes the main classification model used in this work. The model combines all five feature modalities through a sequence of carefully designed steps, forming the Unified Refactored BERT+ViT Fusion Architecture. Figure 2.3 shows the full architecture. Each input feature is projected, aligned via FiLM modulation, and combined using hierarchical intra-modal self-attention and bidirectional cross-modal co-attention. The fused features are blended using a Mixture-of-Experts (MoE) gating mechanism and passed, together with a semantic dissimilarity embedding, to a residual classifier block to output the final prediction.

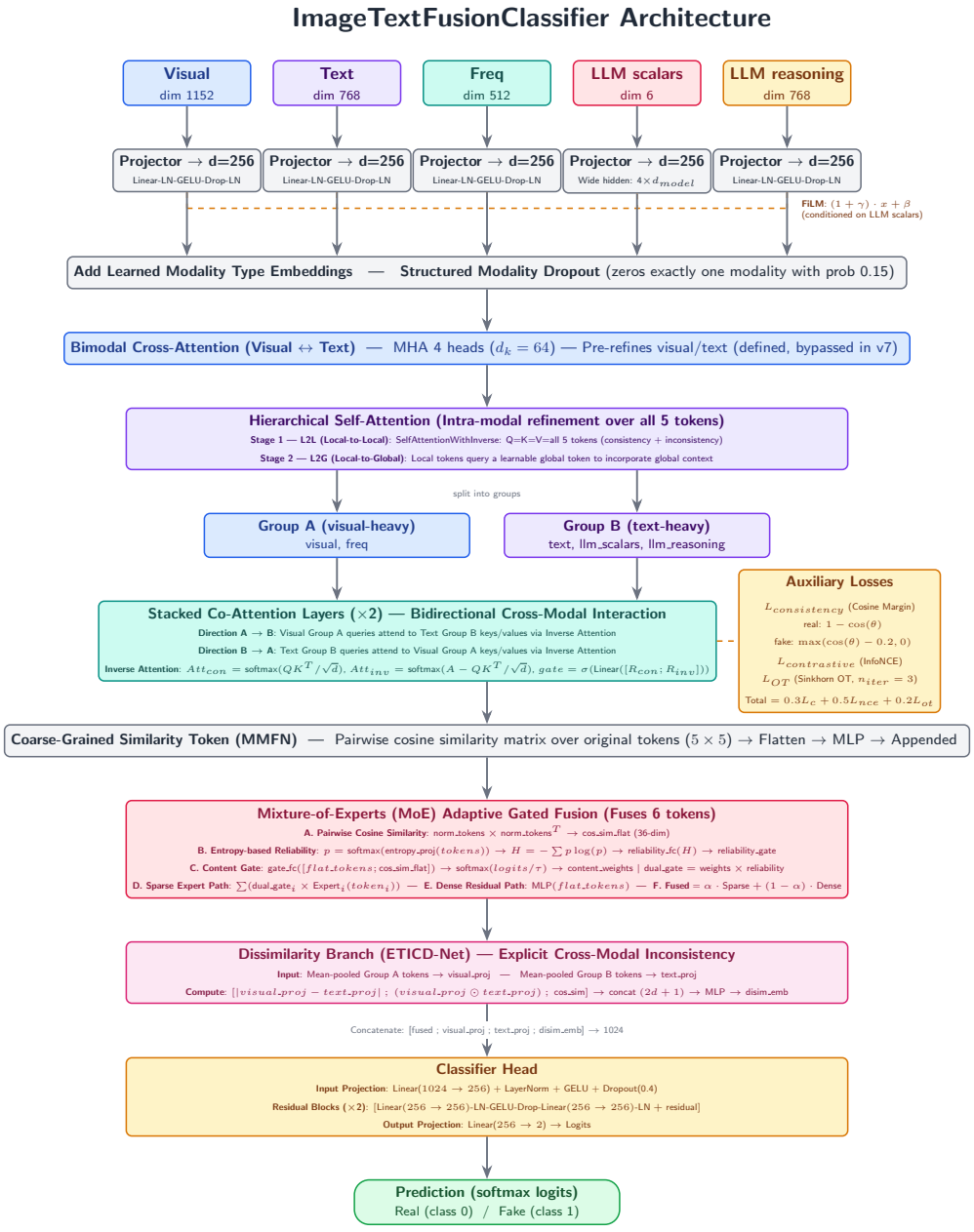


Figure 2.3: Architecture of the proposed five-modality fusion classifier

2.2.11 Modality Projection and Alignment

The five input feature modalities (visual, text, frequency, LLM scalars, and LLM reasoning) originate from different encoders and have highly heterogeneous dimensionalities. To integrate them, they must be projected into a shared latent space and aligned semantically.

a) Heterogeneous Modality Projection

Each input feature vector $\mathbf{x}_m$ is mapped to a shared model dimension $d_{\text{model}} = 256$ using a dedicated modality projector. For high-dimensional inputs ($d_{\text{in}} \geq d_{\text{model}}$), the projection is defined as:

$$\mathbf{z}_m = \text{LayerNorm}(\mathbf{W}_2 \cdot \text{GELU}(\text{LayerNorm}(\mathbf{W}_1\mathbf{x}_m + \mathbf{b}_1)) + \mathbf{b}_2) \tag{2.7}$$

where $\mathbf{W}_1 \in \mathbb{R}^{d_{\text{model}}d_{\text{in}}}$, $\mathbf{W}_2 \in \mathbb{R}^{d_{\text{model}}d_{\text{model}}}$, and GELU denotes the Gaussian Error Linear Unit activation [97]. For low-dimensional inputs (specifically the 6-dimensional LLM scalar features), a wider intermediate hidden layer of size $4d_{\text{model}} = 1024$ is used to prevent representation bottlenecking and allow the network to extract richer representations before final projection to 256 dimensions. This heterogeneous design follows the finding of Baltrušaitis et al. [112] that different modalities require different projection capacities to avoid information loss during dimensionality alignment.

b) Feature-wise Linear Modulation (FiLM)

The projected LLM scalar embedding $\mathbf{z}_{\text{LLM_scalars}} \in \mathbb{R}^{d_{\text{model}}}$ is used as a conditioning signal to modulate the other four modality embeddings $\mathbf{z}_m$ (for $m \in \{\text{visual, text, freq, llm_reasoning}\}$) using Feature-wise Linear Modulation (FiLM), detailed in the Theoretical Background. In our case, the LLM scalar features capture high-level verdict information (confidence, match score, detected signals) that can guide how the model processes the other modalities. The conditioning network generates scale ($\boldsymbol{\gamma}_m$) and shift ($\boldsymbol{\beta}_m$) parameters using a two-layer MLP with GELU activation. FiLM modulation was chosen over simple concatenation because it allows the conditioning signal to selectively amplify or suppress individual feature dimensions, leading to a more expressive integration of the LLM's reasoning into the main classification pipeline.

c) Learned Modality Type Embeddings

To preserve the unique identity of each source modality within the transformer layers, a learned modality type embedding is added to each token. Let $\mathbf{E} \in \mathbb{R}^{5d_{\text{model}}}$ be the learned

type embedding matrix. The initial token representation $\mathbf{h}_m^{(0)}$ is defined as:

$$\mathbf{h}_m^{(0)} = \tilde{\mathbf{z}}_m + \mathbf{E}_m \tag{2.8}$$

where $\mathbf{E}_m$ is the row of $\mathbf{E}$ corresponding to modality m. This design follows the approach used in BERT [58] for distinguishing segment types, adapted here to distinguish modality types.

d) Feature Noise and Structured Modality Dropout

During training, two regularisation mechanisms are applied to prevent overfitting and ensure robust multimodal integration:

1. **Additive Feature Noise:** A small Gaussian noise perturbation $\boldsymbol{\eta} \sim \mathcal{N}(0, 0.05^2\mathbf{I})$ is added to the token representations to improve generalisation.

2. **Structured Modality Dropout:** Instead of applying independent Bernoulli masks, the model drops exactly one entire modality token at random with a probability of 0.15 per sample. This design follows the ModDrop approach proposed by Neverova et al. [116], which showed that dropping whole modalities during training forces the network to learn robust cross-modal correlations. It guarantees that at least four modalities remain active, preventing the model from relying too heavily on a single dominant modality.

2.2.12 Classification Model Architecture

The classification model integrates the aligned tokens using hierarchical attention modules, calculates semantic dissimilarity, and applies an adaptive Mixture-of-Experts (MoE) fusion before binary classification.

a) Intra-Modal Hierarchical Self-Attention

The aligned modality tokens are first processed by a Hierarchical Self-Attention module that refines representations using a two-stage local-to-global attention mechanism:

1. **Local-to-Local (L2L) Stage:** A self-attention pass is applied across the sequence of modality tokens to capture initial contextual associations. This stage uses an inverse attention mechanism to extract internal alignment and contradiction details.

2. **Local-to-Global (L2G) Stage:** A learnable global query token $\mathbf{t}_{\text{global}} \in \mathbb{R}^{d_{\text{model}}}$ is introduced. The local modality tokens attend to this global token, allowing the model to capture the holistic context of the input sequence.

b) Bidirectional Co-Attention with Inverse Attention

To model cross-modal interactions, the modalities are partitioned into two groups: the visual-heavy group (Group A: $\mathcal{G}_A = \{\text{visual}, \text{freq}\}$) and the text-heavy group (Group B: $\mathcal{G}_B = \{\text{text}, \text{llm_scalars}, \text{llm_reasoning}\}$).

Standard attention captures only consistency (agreement) between modalities. In fake news detection, inconsistencies between what the image shows and what the caption claims are the primary detection signal. Therefore, we utilize the Inverse Attention mechanism (described in the Theoretical Background) to capture both consistency and inconsistency features. Stacked bidirectional co-attention layers ($L_{\text{co}} = 2$) are applied. In each direction (Group A attending to Group B, and Group B attending to Group A), the model computes both standard and inverse attention. The co-attended tokens are combined using a sigmoid-gated fusion and then passed through standard Feed-Forward Networks (FFN) with residual connections and layer normalisation.

This mechanism was adopted because capturing cross-modal contradictions improves fake news detection accuracy, as fake samples often have subtle disagreements between the image content and the textual claim that standard attention overlooks.

c) Coarse-Grained Similarity Representation

Before the co-attention stack, a coarse-grained similarity representation is computed from the original tokens $\mathbf{h}^{(0)}$. Let $\mathbf{C} \in \mathbb{R}^{55}$ be the pairwise cosine similarity matrix of the L2-normalised tokens:

$$C_{ij} = \frac{\mathbf{h}_i^{(0)} \cdot \mathbf{h}_j^{(0)}}{\|\mathbf{h}_i^{(0)}\| \|\mathbf{h}_j^{(0)}\|} \tag{2.9}$$

The matrix $\mathbf{C}$ is flattened and projected using a two-layer MLP to form a coarse similarity token $\mathbf{c} \in \mathbb{R}^{d_{\text{model}}}$, which is concatenated to the five co-attended modality tokens, forming a sequence of six tokens.

d) Mixture-of-Experts (MoE) Adaptive Gated Fusion

Feature fusion is the critical process of combining these diverse, heterogeneous representations—visual semantics, textual claims, low-level frequency traces, and high-level LLM reasoning—into a single, cohesive representation that the classifier can analyze. Because not all features are equally reliable for every sample (e.g., an unmanipulated but out-of-context image will have no frequency anomalies but strong semantic mismatch), the fusion mechanism must be adaptive.

To fuse the six sequence tokens intelligently, a Mixture-of-Experts (MoE) Adaptive Gated Fusion mechanism is used. This is adapted to our multimodal setting such that each modality has its own dedicated expert network.

1. **Entropy-Based Reliability Gating:** Each token's uncertainty is measured via the entropy of a projected probability distribution. Tokens with low entropy (high confidence) receive higher gate values, while uncertain tokens are down-weighted.

2. **Content-Based Gating:** A softmax weighting is computed from the concatenated tokens and the coarse similarity matrix.

The final gating weights combine both the reliability and content scores. The model then blends a sparse MoE representation (where each modality is processed by its dedicated expert) with a dense residual projection of the flattened tokens. This sparse-dense blending improves stability: the sparse path allows each modality to be processed by its own specialised expert, while the dense path prevents information loss when the gating weights are highly skewed.

e) Explicit Cross-Modal Dissimilarity

To model the semantic gap between visual and textual content explicitly, the Group A and Group B enhanced tokens are mean-pooled to form visual ($\mathbf{v}_{\text{pool}} \in \mathbb{R}^{d_{\text{model}}}$) and text ($\mathbf{t}_{\text{pool}} \in \mathbb{R}^{d_{\text{model}}}$) projections. A dissimilarity feature vector is computed by:

$$\mathbf{f}_{\text{diss}} = \text{MLP}([|\mathbf{v}_{\text{pool}} - \mathbf{t}_{\text{pool}}| \parallel (\mathbf{v}_{\text{pool}} \odot \mathbf{t}_{\text{pool}}) \parallel \cos_\text{sim}(\mathbf{v}_{\text{pool}}, \mathbf{t}_{\text{pool}})]) \tag{2.10}$$

This explicit dissimilarity signal complements the inverse attention mechanism. While inverse attention detects token-level contradictions, this branch captures the overall semantic gap between the visual and textual modality groups.

f) Residual Classifier Head

The classifier head takes a concatenated representation of the fused features, visual and text projections, and the dissimilarity feature:

$$\mathbf{h}_{\text{classifier}} = [\mathbf{f}_{\text{fused}} \parallel \mathbf{v}_{\text{pool}} \parallel \mathbf{t}_{\text{pool}} \parallel \mathbf{f}_{\text{diss}}] \in \mathbb{R}^{4d_{\text{model}}} \tag{2.11}$$

This 1024-dimensional vector is passed through an input projection layer, layer normalisation [81], GELU activation [97], dropout, and two sequential residual blocks [45]:

$$\text{logits} = \mathbf{W}_o \cdot \text{ResBlock}_2(\text{ResBlock}_1(\text{Dropout}(\text{GELU}(\text{LayerNorm}(\mathbf{W}_i \mathbf{h}_{\text{classifier}} + \mathbf{b}_i))))) + \mathbf{b}_o \tag{2.12}$$

yielding raw logits for the two classes (Real and Fake). The residual connections in the classifier head follow the design of He et al. [45], which allows the gradient to flow directly through skip connections and prevents the vanishing gradient problem in deeper classification networks.

2.2.13 Multi-Task Learning and Loss Functions

The model is trained end-to-end using a multi-task objective that combines classification loss with three auxiliary cross-modal alignment losses.

a) Focal Loss

To address potential class imbalances and asymmetric error rates, Focal Loss (described in the Theoretical Background) is used as the primary classification loss. We use a focusing parameter of $\gamma = 2.0$ and apply label smoothing. Focal Loss automatically down-weights easy, well-classified examples and focuses the training signal on hard, misclassified cases. This is crucial in fake news detection because some samples are obviously real or clearly fake, while others are ambiguous. Standard cross-entropy treats every sample equally, which can slow learning on the difficult cases that matter most.

b) Cross-Modal Consistency Loss

To enforce semantic consistency in the latent space, a cosine-margin loss is applied between pooled visual and text representations:

$$\mathcal{L}_{\text{cons}} = \mathbb{1}[y = 0](1 - \cos(\mathbf{v}_{\text{pool}}, \mathbf{t}_{\text{pool}})) + \mathbb{1}[y = 1] \max(0, \cos(\mathbf{v}_{\text{pool}}, \mathbf{t}_{\text{pool}}) - m) \quad (2.13)$$

where $y \in \{0, 1\}$ (0 for Real, 1 for Fake) and $m = 0.2$ is the margin. This loss encourages real articles to have highly similar visual and text features, while pushing fake articles' representations apart if their similarity exceeds the margin. The margin value of 0.2 was selected to allow some similarity between modalities even in fake samples, since some fake news articles reuse real images with only slightly altered captions.

c) Contrastive Alignment Loss (InfoNCE)

A symmetric InfoNCE loss is employed to align matching visual and text projections in a batch while separating non-matching pairs. To improve robustness, semi-hard negative mining is applied by masking out negative pairs that have a cosine similarity below 0.2. This prevents the loss from being dominated by trivially easy negatives.

d) Sinkhorn Optimal Transport Loss

To align the feature distributions globally within each batch, a Sinkhorn Optimal Transport loss is computed with a regularisation parameter $\epsilon = 0.1$. This loss complements the InfoNCE loss: while InfoNCE aligns individual pairs, the Sinkhorn OT loss ensures that the overall distributions of visual and text features have a consistent geometric structure across the entire batch.

e) Overall Multi-Task Loss

The total training loss $\mathcal{L}_{\text{total}}$ is a weighted combination:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{Focal}} + \lambda_{\text{aux}} \left(w_1 \mathcal{L}_{\text{cons}} + w_2 \mathcal{L}_{\text{cont}} + w_3 \mathcal{L}_{\text{OT}} \right) \tag{2.14}$$

where $\lambda_{\text{aux}} = 0.1$ modulates the auxiliary loss weight, and the internal task weights are set to $w_1 = 0.3$, $w_2 = 0.5$, and $w_3 = 0.2$. The contrastive loss receives the highest weight ($w_2 = 0.5$) because it directly optimises the alignment between individual visual–text pairs, which is the most informative signal for detecting mismatched image–caption pairs.

Part II: Approach 2 — Video-Based Detection

2.3 Approach 2 : Videos

2.3.1 Feature Extraction

This stage is the essential part of the proposed method. We designed a system to extract eleven different features from each sample. The main goal is to create a complete digital representation that includes everything from the meaning of text to visuals and sound.

The reason for extracting such a diverse set of eleven features is that modern fake news in short videos is highly sophisticated. Deceptive creators rarely fake every element perfectly; a video might have a very convincing text description but contain AI-generated visuals or poorly synchronized audio. Relying on just one or two modalities would leave dangerous blind spots. By combining high-level semantic features with low-level technical features, we create a robust safety net. Ultimately, this allows the system to detect *cross-modal inconsistencies*, where different parts of the video contradict each other, ensuring that sophisticated fakes cannot pass undetected [117, 110].

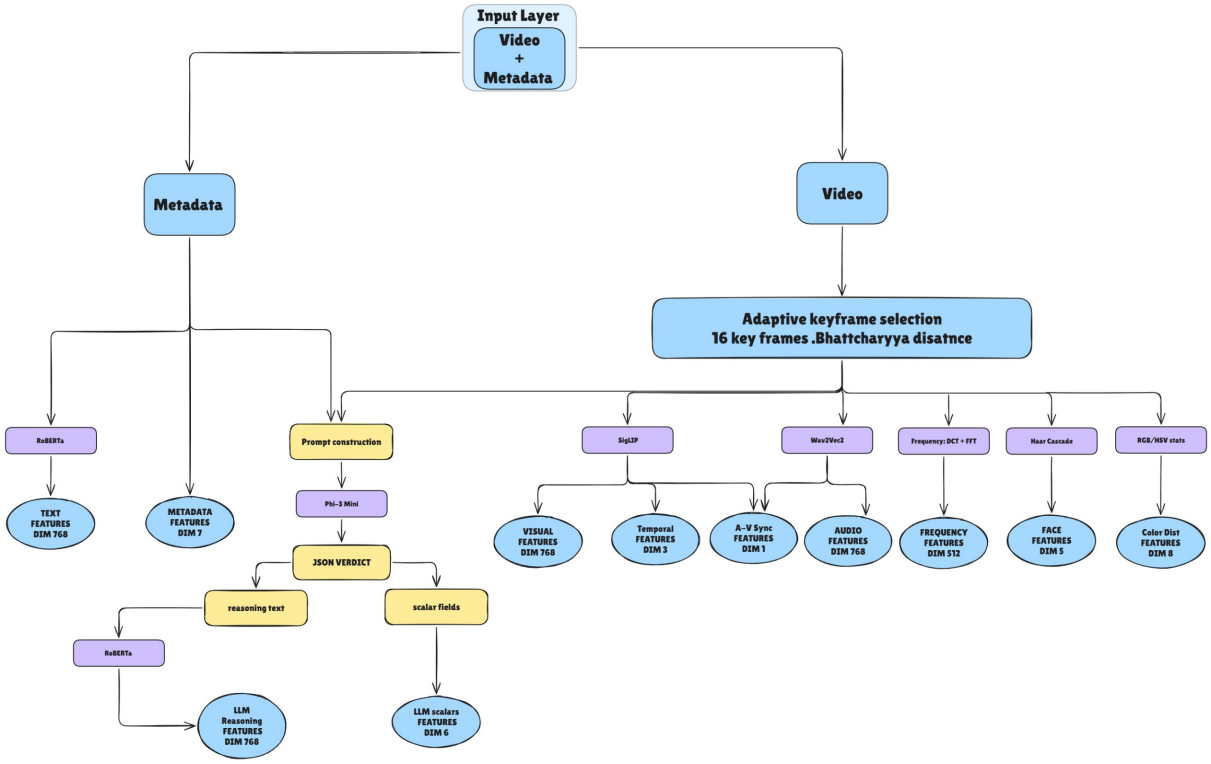


Figure 2.4: Feature extraction pipeline for the video-based approach

2.3.2 Adaptive Keyframe Selection

Key-frame extraction techniques have been extensively studied in the literature as an effective approach for reducing video redundancy by selecting representative frames based on visual similarity measures and clustering strategies [118]. In this work, instead of working on all video frames, the system selects the sixteen most visually useful frames in each video to avoid repetition and save memory. The selection process works by comparing grayscale histograms of successive frames using the Bhattacharyya distance (detailed in the Theoretical Background). Frames are ranked according to these dissimilarity scores, from the most different to the least. Next, the system selects the top sixteen frames and returns them to their original chronological order to maintain the natural sequence of the video.

2.3.3 Textual Feature Extraction

The system uses RoBERTa-base to encode textual metadata. RoBERTa-base was selected for text encoding due to its superior performance with social media content. Unlike other models that were trained exclusively on formal text, RoBERTa was pre-trained on expansive open web data. This enables the model to effectively analyze informal writing styles, abbreviations, emojis, and common social media slang, which are highly frequent in short-video metadata [119].

The input to the model is a single text string that includes: the video description (with hashtags removed), the event category tag, the user self-description, and the account verification status. Between each piece, the model uses a special separator token [SEP] to separate them:

```
"{description} [SEP] Event: {event} [SEP] User: {user_description}
[SEP] Verified: {yes|no}"
```

The model processes this combined representation and creates a 768-dimensional feature vector, denoted $\mathbf{t} \in \mathbb{R}^{768}$. This vector represents the overall meaning and context of the text in a numerical format suitable for classification.

2.3.4 Visual Feature Extraction

The system extracts visual features from the sixteen previously selected keyframes using SigLIP -base. SigLIP-base was selected for its exceptional capacity to translate visual scenes into high-level conceptual meanings rather than merely identifying isolated objects. Driven by its joint text-image pre-training, the model effectively understands the full story and general meaning of the whole scene [83].

The system converts each keyframe from BGR color format to RGB. Then, it resizes the image to 224224 pixels. The image is then passed through the SigLIP vision encoder,

which produces a vector containing 768 values. This vector is the visual embedding of that single keyframe.

To obtain a final feature vector for the entire video, the system takes the embeddings of all sixteen keyframes and averages them (mean-pooling):

$$V_{video} = \frac{1}{16} \sum_{i=1}^{16} E_i \tag{2.15}$$

where E_i denotes the visual embedding of the i-th keyframe.

This results in a single 768-dimensional feature vector, denoted $\mathbf{v} \in \mathbb{R}^{768}$,capturing the visual content.

2.3.5 Acoustic Feature Extraction

The system extracts acoustic features from the audio segments corresponding to the sixteen selected keyframes using Wav2Vec2-base. Wav2Vec2-base was selected for its exceptional capacity to capture subtle speech nuances and emotional tones rather than merely processing clean, formal audio. As demonstrated in independent speech evaluation benchmarks, the model's self-supervised representations effectively understand informal vocal patterns and robust auditory context across diverse real-world scenarios [120]. The audio processing follows these steps:

Extraction and Preprocessing: The audio stream is separated from each video using FFmpeg. It is then resampled to 16 kHz (mono) to comply with the model's input requirements.

Windowing and Synchronization: A 1-second time window (0.5 seconds before and 0.5 seconds after) is extracted for all sixteen keyframes of the video giving us audio-visual sync that helps detect content manipulation.

Encoding and Aggregation: The Wav2Vec2 model processes each audio window and generates a 768-dimensional feature vector. These embeddings are then combined using mean-pooling to produce a single aggregated feature vector that represents the full audio content of the video:

$$A_{video} = \frac{1}{16} \sum_{i=1}^{16} W_i \tag{2.16}$$

where W_i denotes the acoustic embedding of the i-th keyframe window, and $\mathbf{a} \in \mathbb{R}^{768}$.

2.3.6 Frequency-Domain Feature Extraction

Image manipulation (such as cutting, pasting, or AI generation) leaves traces that are invisible to the human eye, which can be effectively detected through signal processing methods, specifically the Discrete Cosine Transform (DCT) and Fast Fourier Transform

(FFT) [93].

Two complementary transformations are applied to each keyframe:

- **Discrete Cosine Transform (DCT):** The system converts each frame of the video into a small image (6464 pixels) in grayscale, then the Discrete Cosine Transform (DCT) is applied. We obtain a 6464 matrix of numerical coefficients, from which the upper left part (1616) is selected, which represents the low-frequency components, the general and smooth patterns in the image (not the fine details). Then, a logarithmic transformation is applied to reduce the large differences between the values, and in the end, we obtain a 256-dimensional vector that represents a signature that reveals whether the image is natural or contains traces of compression or manipulation. The DCT-based feature is computed as:

$$F_{DCT} = \log(1 + |DCT(I)|^2) \tag{2.17}$$

  where I denotes the grayscale input frame.

- **Fast Fourier Transform (FFT):** It is applied to the same grayscale frame (6464). By applying the Fast Fourier Transform (FFT), the image is converted from the pixels domain into a matrix of values representing the magnitude of each frequency present in the image, and the low-frequency components are shifted from the corners of the matrix to its center so that the spectrum becomes clearer and easier to interpret. Then, a logarithmic transformation is applied to the values to reduce large differences between them, making them more stable for analysis. After that, the spectrum is reduced to a 1616 grid, which gives us a 256-dimensional feature vector representing the fundamental frequency patterns . The FFT-based feature is computed as:

$$F_{FFT} = \log(1 + |\mathcal{F}(I)|^2) \tag{2.18}$$

  where I denotes the grayscale input frame.

The two 256-dimensional vectors are concatenated and L2-normalized to produce a 512-dimensional frequency feature. The features are averaged across all sixteen keyframes to produce the final frequency representation.

$$f_{freq} = \frac{1}{N} \sum_{i=1}^{N} \text{L2-normalize}([F_{DCT}^{(i)} \| F_{FFT}^{(i)}]) \tag{2.19}$$

where $N = 16$ is the number of keyframes, and $\mathbf{f}_{freq} \in \mathbb{R}^{512}$.

2.3.7 Face Analysis Features

Facial regions are the primary targets of modern video manipulation. As demonstrated by Li and Lyu. [121], deepfake models generate faces in fixed resolutions that must be scaled and warped to fit the video frame. This process inevitably introduces artifacts like facial blurring and fluctuating size between keyframes. The face-related features describe whether faces are present, their size, and how clearly they appear across keyframes. The system uses the Haar Cascade frontal face classifier to detect faces.

For each keyframe, the detector identifies all face regions. It then computes the following statistics for those faces:

- **Face presence ratio :**Percentage of keyframes that have at least one face detected.
- **Mean face area:** The average ratio of the largest detected face bounding box to the total frame area across keyframes, normalized to [0, 1].
- **Face blur score:** A measure of how clear the face is. If the face is blurry (low variance), this indicates that the image is compressed or artificial. Normalized to [0, 1] .
- **Face area standard deviation:**Significant change in facial size across frames reveals manipulation of content.
- **Dominant face indicator:** Binary value (0 or 1), if the face is present in more than half of the frames it gives the value 1.

These five statistics are concatenated to form a 5-dimensional face feature vector, denoted $\mathbf{f}_{face} \in \mathbb{R}^5$.

2.3.8 Temporal Consistency Features

Video manipulation operations, such as splicing or frame substitution, disrupt the continuous flow of visual information over time. Although individual frames may appear authentic, modern deepfake generation pipelines frequently fail to maintain long-term temporal consistency, resulting in abrupt shifts between consecutive frames. As demonstrated in the recent temporal learning benchmarks by Zhao et al. [122], tracking these frame-to-frame variations using deep semantic embeddings effectively exposes forgery or content manipulation. To capture these temporal dynamics, the system measures the semantic continuity between adjacent keyframes using the cosine similarity between the embeddings generated by SigLIP:

$$s_i = \frac{E_i \cdot E_{i+1}}{\|E_i\|\|E_{i+1}\|} \tag{2.20}$$

where E_i and E_{i+1} denote the SigLIP visual embeddings of the i-th and $(i+1)$-th keyframes, respectively.

This produces a series of results that show how similar each pair of adjacent frames is. Three statistics are derived from this sequence:

- **Average similarity between frames:** This measures how visually consistent the video is from one frame to another. A low value means there are sudden changes between scenes, while a high value indicates that the video is natural and cohesive.
- **Standard deviation of similarity between frames:** This measures how different the similarity is between successive frames. A high value suggests that the video was compiled from heterogeneous sources, whereas a low value indicates temporal homogeneity.
- **Minimum frame similarity:** This value identifies the most sudden point in the video. Such a sharp change may indicate a break point or inconsistency between successive scenes.

These three statistics form a 3-dimensional temporal feature vector that represents how visual content changes over time, denoted $\mathbf{f}_{temp} \in \mathbb{R}^3$.

2.3.9 Audio-Visual Synchronization Score

In real videos, the way visual content changes over time naturally correlates with the way sound changes, so if the content is manipulated and sounds from different sources are combined, this connection between visual content and sound is disrupted [123]. The degree of synchronization is calculated by measuring the rate of change from frame to frame in both the visual embedding space and the audio embedding space via the L_2 norm, giving us two time series representing the rate of change for each modality.

$$\delta_i^{(v)} = \|E_i^{(v)} - E_{i-1}^{(v)}\|_2, \quad \delta_i^{(a)} = \|E_i^{(a)} - E_{i-1}^{(a)}\|_2 \tag{2.21}$$

where $E_i^{(v)}$ and $E_i^{(a)}$ represent the visual and audio embeddings for the i-th frame, respectively.

These vectors are extracted from different latent spaces, so they are normalized using the Z-score to bring them within a uniform scale.

$$\hat{\delta}^{(m)} = \frac{\delta^{(m)} - \mu^{(m)}}{\sigma^{(m)}} \tag{2.22}$$

where $\mu^{(m)}$ and $\sigma^{(m)}$ denote the mean and standard deviation of the modality m time series, respectively.

Then, the cosine similarity between them is calculated.

$$S_{sync} = \frac{\hat{\delta}^{(v)} \cdot \hat{\delta}^{(a)}}{\|\hat{\delta}^{(v)}\| \|\hat{\delta}^{(a)}\|} \tag{2.23}$$

resulting in a value, denoted $\mathbf{f}_{sync} \in \mathbb{R}$, bounded in $[-1, 1]$, which determines the degree of temporal covariance between the visual and acoustic modalities.

2.3.10 Color Distribution Features

In authentic imaging systems, the acquisition of visual content is governed by the physics of light and sensor characteristics, yielding a consistent and natural color distribution. Conversely, manipulation processes—such as image splicing, generative AI synthesis, or digital editing disrupt this consistency, leaving distinct anomalies in color values, contrast, and saturation [124].

Color statistics characterize the chromatic properties of the video for each keyframe. The following statistics are calculated in the RGB and HSV (Hue, Saturation,Value) color spaces:

- **Average RGB values:** Represents the overall dominant color tone of the video (3 values).
- **Standard deviation of RGB values:** Measures how diverse the colors are in the scene (3 values).
- **Average Saturation:** The frame is converted to HSV color space and the mean saturation is computed. Over-saturated or unsaturated videos may indicate post-processing.
- **Contrast:** The frame is converted to grayscale and the standard deviation of pixel intensities is computed.

The eight statistics are averaged across all keyframes to produce an eight-dimensional color feature vector, denoted $\mathbf{f}_{color} \in \mathbb{R}^8$.

2.3.11 Metadata Features

Deception and misinformation are not confined to editing the video itself, but also appear in the behavior of the account that publishes it, as disinformation actors and malicious profiles have anomalous behavioral patterns on social media platforms to ensure the spread of their rumors [111], and the system leverages this data in order to detect this deception.

Seven features are extracted:

- **Publication hour:** Normalized to [0,1], this feature captures whether there are specific hours during which misinformation is spread.
- **Day of week:** The goal is to determine whether there are certain days when accounts that post misleading content become more active, normalized to [0, 1].
- **Night publication indicator:** Binary value (0 or 1); 1 if the video is posted between 10 PM and 5 AM.
- **Account verification status:** A binary value (0 or 1) indicating whether the account is verified or not.
- **Hashtag density:** The number of hashtags in the description is normalized to a maximum of 30, as disinformation promoters often use a large number of trending hashtags even if they are not related to the content, in order to ensure that the video appears to the largest possible number of people.
- **Description length:** The number of words that do not contain a hashtag in the description, normalized to a maximum of 100 words.
- **Event-description semantic overlap:** Computed using the Jaccard similarity (detailed in the Theoretical Background) between event keywords and description words, to measure how well the text matches the declared topic.

Finally, it produces a feature vector denoted $\mathbf{f}_{meta} \in \mathbb{R}^7$.

2.3.12 LLM-Based Cross-Modal Consistency Reasoning

Large Language Models (LLMs) serve as reasoning engines in multimedia verification, using their language capabilities to evaluate content logic [125]. While traditional feature fusion methods often fail to detect semantic contradictions between text and video, LLMs provide a contextual reasoning layer to address this issue.

The proposed framework uses Phi-3 Mini as a Cross-Modal Consistency Analyzer. The choice of Phi-3 Mini is motivated by its reasoning performance and computational efficiency, which allow local execution without cloud API delays and operational costs [126].

To prevent the model from judging news based on its pre-trained information, it operates strictly as a consistency analyzer rather than an open-domain fact-checking tool. Consequently, the model focuses on evaluating the logical conformity between the extracted visual content and the accompanying video description [127].

a) Model Inputs

The model receives the following inputs:

- **Text Description:** Revised text of the video.
- **Event Category:** News type.
- **Account Type:** Verified or unverified.
- **Light Visual Summary:** A textual description of the five keyframes with the highest brightness and visual complexity scores.

b) LLM Scalar Feature Vector

The model returns a JSON response containing six scores in the range $[0, 1]$:

- **visual_text_consistency:** The extent to which the visual scene matches the text description.
- **temporal_coherence:** The extent of visual consistency between frames over time.
- **source_credibility:** Account credibility based on account type and description style.
- **manipulation_indicators:** Measures the presence of linguistic or syntactic patterns associated with misleading content.
- **cross_modal_alignment:** How well the signals coming from the text and images fit together.
- **overall_suspicion:** Overall degree of doubt about the authenticity of the content.

These six scores form the six-dimensional llm_scalars feature vector, denoted $\mathbf{f}_{llm} \in \mathbb{R}^6$.

c) LLM Reasoning Embedding

The model produces a short explanatory sentence that is converted to a 768-dimensional vector using RoBERTa denoted as $\mathbf{r} \in \mathbb{R}^{768}$.

This representation complements the scalar features by preserving the semantic explanation behind the assigned scores.

d) Fallback Response

If a logical result is not obtained, the model is restarted up to three times for each sample.If the model fails in all attempts or encounters a connection failure or response error, a numerical vector of llm_scalars is given a value of 0.5 for each element and the text embedding is used as the default value for llm_reasoning to prevent it from remaining empty.

2.3.13 Summary of Extracted Features

Table 2.2 summarises all eleven feature groups extracted by the proposed pipeline, including their dimensionality, extraction method, and semantic role.

Table 2.2: Summary of Extracted Feature Groups (Approach 2: Video)

Feature Group	**Dim.**	**Extraction Method**	**Semantic Role**
Text	768	RoBERTa CLS token	Encodes description, event, and account context
Video	768	SigLIP mean pooling	Captures average visual content across keyframes
Audio	768	Wav2Vec2 mean pooling	Encodes acoustic content aligned to keyframe timestamps
Frequency	512	DCT (256) + FFT (256)	Detects compression artifacts and GAN/diffusion patterns
Face	5	Haar Cascade detector	Characterizes facial presence, size, blur, and stability
Temporal	3	SigLIP cosine similarity	Quantifies visual transition smoothness across keyframes
AV-Sync	1	L2 change-rate cosine sim.	Measures audio-visual temporal co-variation
Color	8	RGB/HSV statistics	Captures global color tone, contrast, and saturation
Metadata	7	Structured record parsing	Encodes publication patterns and topical alignment
LLM Scalars	6	Phi-3 Mini JSON output	Cross-modal consistency scores from LLM reasoning
LLM Reasoning	768	RoBERTa on LLM text	Semantic embedding of the LLM justification string
TOTAL	**3614**	—	Complete multimodal feature vector per video

2.3.14 Classification Architecture

The complete feature vector is passed to a classification network that is trained to distinguish between real and fake videos. As shown in Figure 2.5, the classification network

consists of three main stages: a feature projection layer, a cross-modal attention module, and a fully connected classifier head.

Initially, the heterogeneous features (video, audio, text, frequency, and metadata) are projected into lower-dimensional spaces to ensure numerical balance. Subsequently, the text modality acts as a foundational query to cross-attend to the visual and acoustic features. This mechanism explicitly evaluates whether the multimedia content semantically aligns with the textual claims. Finally, all the refined and aligned features are concatenated and passed through a funnel-like Multi-Layer Perceptron (MLP) to output the final binary veracity probability.

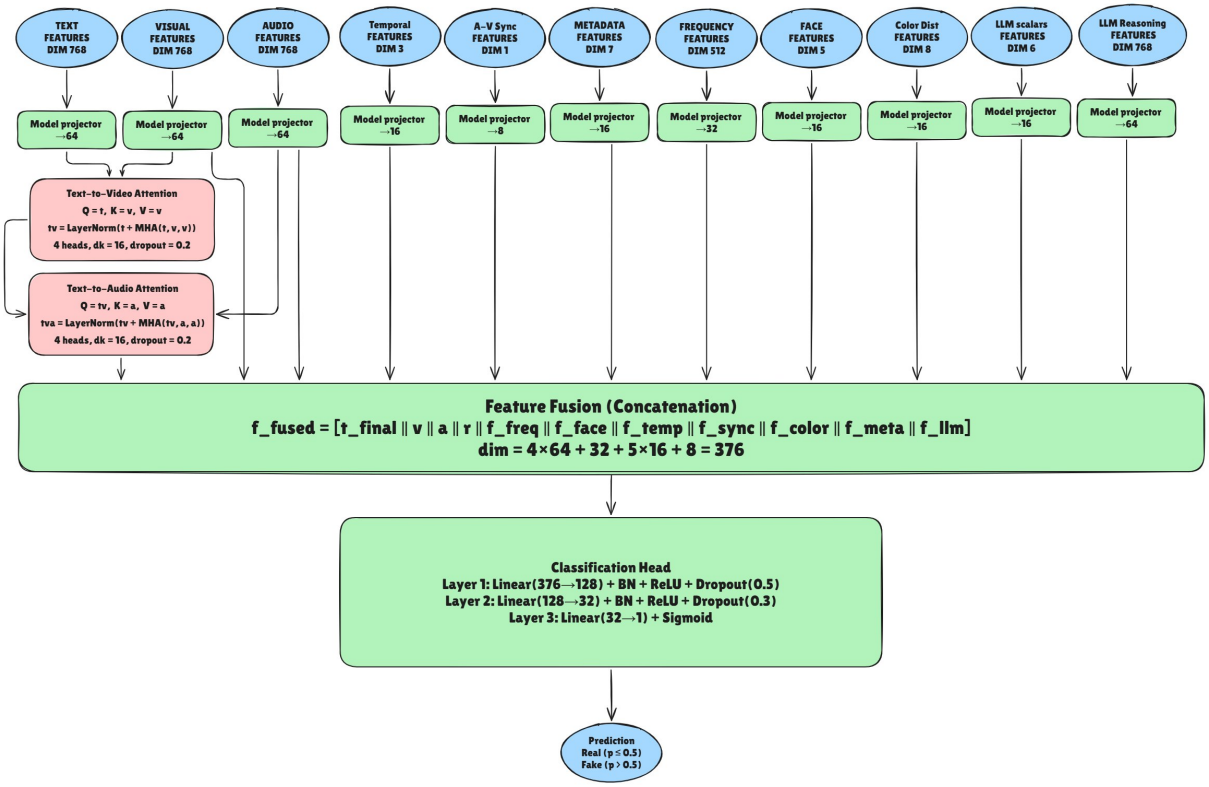


Figure 2.5: Fusion and classification architecture for the video-based approach

2.3.15 Feature Projection Layer

In order to avoid the problem of dimensional imbalance resulting from direct feature merging, each feature set passes through its own linear projection layer, followed by the application of the ReLU activation function. As shown in the table 2.3.

This step brings high dimensions such as text, video, and audio features (768) into a convergent range with low-dimensional features such as the AV-sync score (1), allowing the classifier to learn from all properties without numerically dominating the high-dimensional properties over the low-dimensional properties [128].

Table 2.3: Per-modality projection dimensions

Modality	**Input Dim.**	**Output Dim.**
Text	768	64
Video	768	64
Audio	768	64
LLM Reasoning	768	64
Frequency	512	32
Face	5	16
Temporal	3	16
Color	8	16
Metadata	7	16
LLM Scalars	6	16
AV-Sync	1	8
Fused	3614	**376**

2.3.16 Cross-Modal Attention

In the domain of multimodal fake news detection, the primary objective is to verify cross-modal consistency by determining whether the textual claim semantically aligns with the accompanying visual and acoustic data [129]. To operationalize this verification process, the architecture explicitly designates the text embedding as the query, while mapping the video and audio features as the keys and values:

$$\text{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{2.24}$$

where $\mathbf{Q}$ denotes the query representation, $\mathbf{K}$ denotes the key representation, and $\mathbf{V}$ denotes the value representation. The term d_k corresponds to the dimensionality of the key vectors. The text embedding serves as the query since the objective is to assess whether the visual and acoustic content supports the textual claim.

Text-to-Video Attention: In the first stage, the projected text representation $\mathbf{t}$ attends to the projected video representation $\mathbf{v}$:

$$\mathbf{t}_v = \text{LayerNorm}\left(\mathbf{t} + \text{MHA}(\mathbf{t},\, \mathbf{v},\, \mathbf{v})\right) \tag{2.25}$$

where $\mathbf{t}$ and $\mathbf{v}$ denote the projected text and video representations respectively, and $\text{MHA}(\cdot)$ denotes multi-head attention with query, key, and value inputs.

Text-to-Audio Attention: In the second stage, the updated text representation $\mathbf{t}_v$ attends to the projected audio representation $\mathbf{a}$:

$$\mathbf{t}_{va} = \text{LayerNorm}\left(\mathbf{t}_v + \text{MHA}(\mathbf{t}_v,\, \mathbf{a},\, \mathbf{a})\right) \tag{2.26}$$

Following each cross-modal attention operation, a Layer Normalization (LayerNorm) is applied,each attention layer uses 4 heads with $d_k = 16$, a dropout rate of 0.2, and an embedding dimension of 64.
The resulting representation $\mathbf{t}_{final} = \mathbf{t}_{va}$ serves as the final text encoding that carries the visual and audio context.

2.3.17 Feature Fusion and Classification

After the projection and cross-modal attention alignment phases, all eleven feature representations are integrated using a straightforward concatenation strategy, also known as early fusion. Concatenation is preferred over methods like vector addition or averaging because it preserves the unique information of each individual modality. This prevents information loss and helps the classifier find complex relationships between the features [112].

$$\mathbf{f}_{fused} = [\mathbf{t}_{final} \| \mathbf{v} \| \mathbf{a} \| \mathbf{r} \| \mathbf{f}_{freq} \| \mathbf{f}_{face} \| \mathbf{f}_{temp} \| \mathbf{f}_{sync} \| \mathbf{f}_{color} \| \mathbf{f}_{meta} \| \mathbf{f}_{llm}] \tag{2.27}$$

To compute the final multi-modal veracity prediction, $\mathbf{f}_{\text{fused}}$ is propagated through a deep Multi-Layer Perceptron (MLP) classifier. The network adopts a multi-layer funnel architecture characterized by a gradual dimensionality reduction pipeline:

- **Layer 1:** Linear$(376 \to 128)$ + BatchNorm + ReLU + Dropout(0.5)
- **Layer 2:** Linear$(128 \to 32)$ + BatchNorm + ReLU + Dropout(0.3)
- **Layer 3:** Linear$(32 \to 1)$

This architectural configuration and its associated hyperparameters are justified based on empirical deep learning principles:

1. **Gradual Information Compression:** The transitional compression $(376 \to 128 \to 32 \to 1)$ prevents the formation of an abrupt information bottleneck. Compressing the high-dimensional multimodal vector directly into a single scalar output would compromise the structural abstract features learned hierarchically during the previous stages [130].
2. **Sequential Layer Ordering:** Each classification block follows a strict order:(Linear $\to$ BatchNorm $\to$ ReLU $\to$ Dropout). Placing Batch Normalization immediately after the Linear layer standardizes the data before it reaches the non-linear activation. This stabilizes the training process and prevents neurons from becoming permanently inactive, solving the dying ReLU problem.

3. **Progressive Regularization Scaling:** The dropout rate is stochastically scaled down from $\sigma = 0.5$ in Layer 1 to $\sigma = 0.3$ in Layer 2. Because Layer 1 operates on a massive parameter space, it exhibits a high susceptibility to overfitting, requiring aggressive regularizing constraint ($\sigma = 0.5$). As the dimension contracts in Layer 2, the severe risk of co-adaptation decreases, allowing for a lower dropout rate ($\sigma = 0.3$) to preserve signal transmission stability.

The result of the third layer is passed through the sigmoid function in order to convert the logit into a value between 0 and 1 that represents the probability that the video is fake.

$$p = \sigma(z) = \frac{1}{1 + e^{-z}} \tag{2.28}$$

where $p \in [0, 1]$ is the expected probability that the video is fake. If the value is greater than 0.5, it is classified as fake, and if the value is less than or equal to 0.5, the video is classified as real.

2.3.18 Training Strategy

To ensure stable convergence and robust evaluation, the proposed multimodal model is trained using a carefully configured optimization and regularization pipeline:

- **Loss Function:** A Weighted Cross-Entropy loss (see Theoretical Background) was adopted to address the issue of class imbalance within the dataset. This approach assigns a higher penalty weight to the minority class, ensuring that the model does not develop a bias toward the majority class. By doing so, the model is forced to learn discriminative features for both real and fake samples. Consequently, this loss function improves the overall accuracy and enhances the model's ability to distinguish between different classes in a balanced manner.

- **Optimizer:** The network is optimized using AdamW (see Theoretical Background) with a weight decay of 110^{-4}. AdamW is deliberately selected over standard Adam because it decouples weight decay from the gradient updates, which significantly improves regularization and generalization performance [106].

- **Gradient Clipping:** To mitigate the risk of exploding gradients in deep multimodal architectures, gradient norm clipping (see Theoretical Background) is enforced with a maximum threshold of 1.0 [108]. This ensures that gradient steps remain bounded, preserving training stability.

- **Learning Rate Scheduler:** The training begins with an initial learning rate of 510^{-4}, which is dynamically adjusted using a Cosine Annealing scheduler (see Theoretical Background) over 50 epochs [107]. This scheduling approach smoothly

decreases the learning rate, allowing the model to safely escape local minima early in training and fine-tune its weights precisely during later epochs.

- **Early Stopping:** To prevent overfitting and avoid unnecessary computational overhead, an early stopping mechanism (see Theoretical Background) is implemented. Training is automatically terminated if the macro F_1 score on the validation set fails to improve for 5 consecutive epochs. The macro F_1 score is selected as the tracking metric because it evaluates performance equitably across imbalanced classes.

- **Data Split:** The dataset is partitioned into 70% training, 15% validation, and 15% test sets. Stratified sampling is strictly applied during the split to ensure that the original class distribution is perfectly preserved across all three subsets.

2.4 Conclusion

This chapter presented the methodologies of the two proposed multimodal fake news detection approaches. Approach 1 processes static image–caption pairs from the Fakeddit dataset through five complementary feature streams, namely visual semantics, text encoding, frequency-domain forensics, and LLM-based reasoning, fused via a hierarchical co-attention and Mixture-of-Experts architecture. Approach 2 processes dynamic video content from the FakeTT dataset by integrating keyframe visual features, audio representations, transcribed speech, and structured metadata through a cross-modal attention framework. Both pipelines leverage pre-trained transformer encoders and large language model reasoning to capture cross-modal inconsistencies that are characteristic of fake news content. The experimental evaluation and a comprehensive analysis of the results obtained by both approaches are presented in the following chapter.

Chapter 3

Experimental Results and Discussion

3.1 Introduction

This chapter presents a comprehensive experimental evaluation of the two proposed multimodal fake news detection systems described in Chapter 2. The evaluation is structured to provide rigorous empirical evidence for the design choices made in each approach, covering training dynamics, generalisation stability, test set performance, modality contribution analysis, and a comparison with published state-of-the-art methods.

Approach 1 addresses static image–caption pairs using the Fakeddit benchmark, combining semantic visual features, frequency-domain analysis, text encoding, and LLM-based reasoning through a hierarchical fusion architecture. Approach 2 addresses dynamic short-video content using the FakeTT benchmark, integrating eleven complementary modalities, including facial, temporal, acoustic, and metadata signals, through a cross-modal attention pipeline.

The remainder of this chapter is organised as follows. Section 3.2 describes the hardware and software environment. Section 3.3 defines the evaluation metrics. Sections 3.4 and 3.5 present the detailed results for each approach. Section 3.6 synthesises the findings and draws comparative conclusions.

3.2 Experimental Environment and Setup

3.2.1 Hardware and Cloud Infrastructure

All model training and feature extraction were performed on Kaggle's cloud computing platform, which provides free access to GPU-accelerated virtual machines. The primary training hardware consisted of NVIDIA Tesla P100 GPUs with 16 GB of VRAM, which were sufficient for training the proposed fusion classifiers and running inference with the pre-trained feature encoders. Kaggle notebooks were used for their reproducibility and ease of dependency management.

3.2.2 Development Environment

The development workflow involved the following tools and technologies. These tools were deliberately selected to ensure a robust, scalable, and reproducible pipeline, covering everything from complex multimodal model training in the cloud to the deployment of a real-time mobile application for end-users:

- **Kaggle:** Used as the primary training and experimentation platform. All datasets, model checkpoints, and training logs were managed within the Kaggle environment, enabling consistent and reproducible experiments.

- **Visual Studio Code (VS Code):** Used as the primary code editor for developing and debugging the feature extraction pipeline, classification models, and evaluation scripts.
- **Python 3.10:** The entire machine learning pipeline was implemented in Python, using the following key libraries:
  - **PyTorch** for model implementation and training.
  - **Transformers (Hugging Face)** for loading pre-trained models (SigLIP, Longformer, RoBERTa, BLIP-2, Wav2Vec2).
  - **scikit-learn** for evaluation metrics and cross-validation.
  - **NumPy and SciPy** for numerical computation and frequency-domain analysis (DCT, FFT).
  - **OpenCV** for image preprocessing, face detection, and colour analysis.
  - **Matplotlib** for generating training curves and evaluation plots.
- **FastAPI:** Used as the backend framework for the mobile application's server-side API, handling prediction requests and interfacing with the trained models.
- **PostgreSQL:** Used as the primary relational database for storing user profiles, authentication data, and historical prediction records.
- **Cloudinary:** Integrated as a cloud-based media management solution for uploading, storing, and retrieving user-submitted images and videos.
- **Firebase Cloud Messaging (FCM):** Integrated to deliver real-time push notifications to mobile users upon detection completion.
- **React Native:** Used to develop the cross-platform mobile application serving as the user-facing interface.

3.3 Metrics Utilized for Evaluating Model Performance

To evaluate the effectiveness of the proposed multimodal architectures in detecting fake news, several standard evaluation metrics are employed. These metrics are derived from the confusion matrix, which provides a comprehensive summary of correct and incorrect predictions.

3.3.1 Confusion Matrix

The confusion matrix serves as the fundamental basis for our performance metrics, categorizing the model's outcomes into four distinct groups as illustrated in Table 3.1:

Table 3.1: Confusion Matrix Terminology in Fake News Detection

Symbol	Name	Meaning in Fake News Detection
TP	True Positives	Fake content correctly identified as fake.
TN	True Negatives	Real content correctly identified as real.
FP	False Positives	Real content incorrectly classified as fake (False Alarm).
FN	False Negatives	Fake content incorrectly classified as real (Missed Detection).

3.3.2 Evaluation Metrics

1. Accuracy

Accuracy represents the proportion of total correct predictions made by the model. While it provides a general overview, it may be misleading if the dataset is highly imbalanced:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.1}$$

2. Precision

Precision measures the reliability of the positive predictions, indicating the proportion of articles identified as "fake" that are truly fake:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.2}$$

3. Recall (Sensitivity)

Recall evaluates the model's ability to capture actual positive cases, representing the proportion of real "fake" articles that were successfully detected by the model:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.3}$$

4. F1 Score

The F1 Score is the harmonic mean of Precision and Recall. It is a vital metric in fake news detection, as it provides a balanced measure that accounts for both False Positives and False Negatives, especially in scenarios with uneven class distribution:

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.4}$$

5. Macro F1 Score

To ensure equitable evaluation regardless of dataset size, the Macro F1 Score computes the F1 score independently for each class and averages them, treating both "Real" and "Fake" classes with equal importance:

$$\text{Macro F1} = \frac{1}{C}\sum_{i=1}^{C} \text{F1}_i \tag{3.5}$$

where C is the number of classes.

6. AUC-ROC

The Area Under the Receiver Operating Characteristic (AUC-ROC) curve measures the model's discriminative ability across various classification thresholds. A value closer to 1.0 indicates superior performance in distinguishing between real and fake content.

3.4 Approach 1: Multimodal Fake News Detection for Image and Text

3.4.1 Dataset

The Fakeddit dataset [109] was used for training and evaluating the image–text approach (Approach 1). This dataset contains over one million Reddit submissions with paired images and fine-grained labels across six categories. For the binary classification task, all non-real labels were merged into a single "Fake" class, following the standard evaluation protocol. The dataset was split into 70% training, 15% validation, and 15% test sets, with stratified sampling applied to preserve the class distribution across all splits. The final test set comprised 1,396 samples (728 Real and 668 Fake), reflecting a near-balanced binary distribution.

3.4.2 Hyperparameter Configuration

Table 3.2 summarises the complete hyperparameter configuration used for training the fusion classifier (Approach 1).

Table 3.2: Hyperparameter configuration for the fusion classifier (Approach 1).

Hyperparameter	**Value**
Model dimension (d_{model})	256
Number of attention heads	4
Co-attention layers (L_{co})	2
Dropout rate	0.1
Modality dropout probability	0.15
Feature noise standard deviation	0.05
Optimizer	AdamW
Initial learning rate	510^{-4}
Weight decay	110^{-4}
Learning rate scheduler	Cosine annealing
Maximum epochs	50
Early stopping patience	5 epochs
Gradient clipping (max norm)	1.0
Focal loss γ	2.0
Label smoothing	0.05
Auxiliary loss weight (λ_{aux})	0.1
Consistency loss weight (w_1)	0.3
Contrastive loss weight (w_2)	0.5
Optimal transport loss weight (w_3)	0.2
Cosine margin (m)	0.2
Data split	70% / 15% / 15%
Cross-validation folds	5

3.4.3 Training and Convergence

a) Training Dynamics

The model was trained for 13 epochs before early stopping was triggered, indicating that the architecture converged efficiently without requiring the full 50-epoch budget. Figure 3.1 shows the training and validation loss curves over the training period.

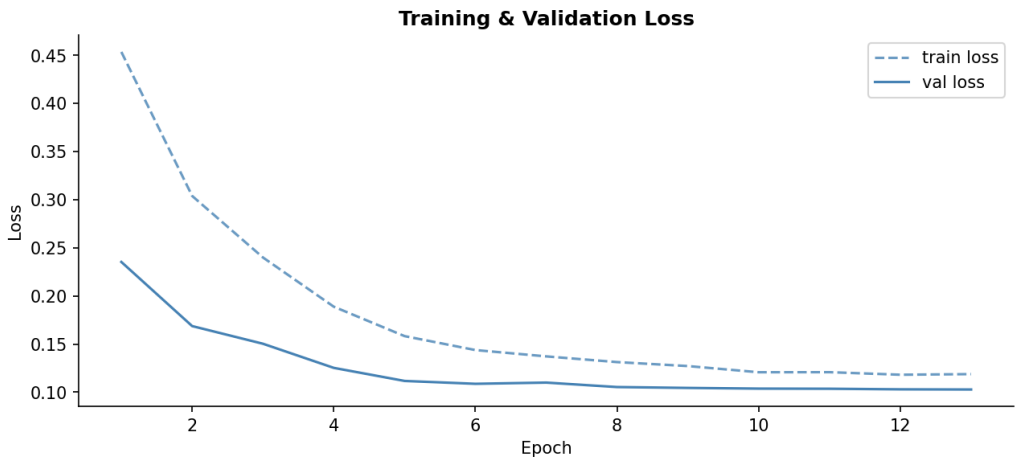


Figure 3.1: Training and validation loss curves over 13 epochs.

Both curves decrease monotonically during the first five epochs, with the validation loss dropping from 0.235 at epoch 1 to 0.112 at epoch 5, before reaching a final value of 0.103 at epoch 13. The rapid initial descent reflects the model's ability to quickly learn the primary discriminative features, particularly the SigLIP visual representations, which provide a strong and consistent signal from the first training steps. The cosine annealing scheduler contributes to this behaviour by maintaining a relatively high learning rate in the early epochs, enabling fast exploration, before gradually reducing it to allow fine-grained weight adjustment in later epochs.

Crucially, the generalisation gap between training and validation loss remained narrow throughout training, staying within ± 0.03 for both F1 and AUC-ROC. This controlled gap confirms that the regularisation mechanisms, including modality dropout (probability 0.15), feature noise ($\sigma = 0.05$), label smoothing (0.05), and dropout (0.1), effectively prevented overfitting despite the architectural complexity of the five-modality fusion system.

b) Macro F1 Score Progression

Figure 3.2 presents the macro F1 score progression during training.

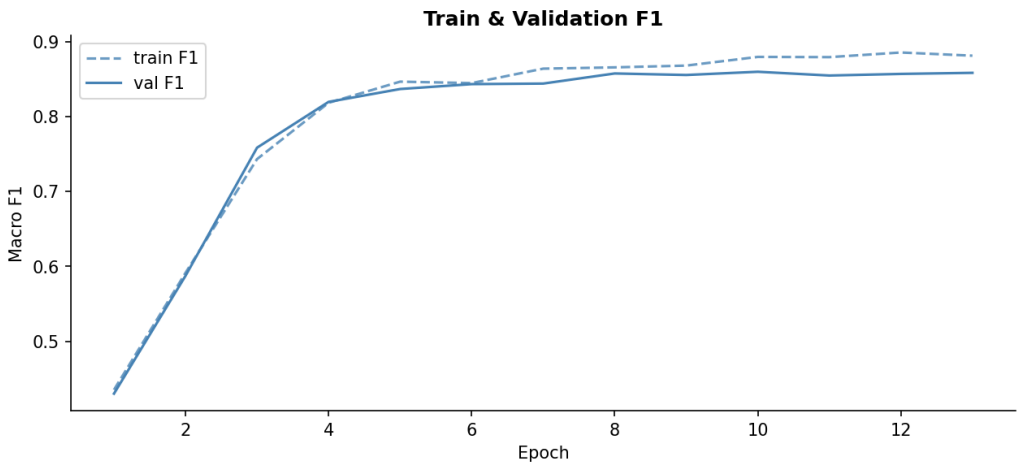


Figure 3.2: Training and validation macro F1 score over 13 epochs.

The validation F1 increases steeply during the first four epochs, rising from 0.430 at epoch 1 to 0.819 at epoch 4, a gain of 0.389 points in just four steps. This rapid improvement is attributable to the dominant role of the SigLIP visual encoder, which encodes rich semantic representations that the classifier can exploit immediately. After epoch 4, the validation F1 stabilises in the range of 0.855–0.860, indicating that the model has saturated the discriminative capacity available from the current feature set. The training F1 continues to rise slightly, reaching 0.881 at epoch 13, producing a small but stable generalisation gap of approximately 0.02 points.

c) Overfitting Analysis

Figure 3.3 presents the train–validation F1 gap across all 13 training epochs.

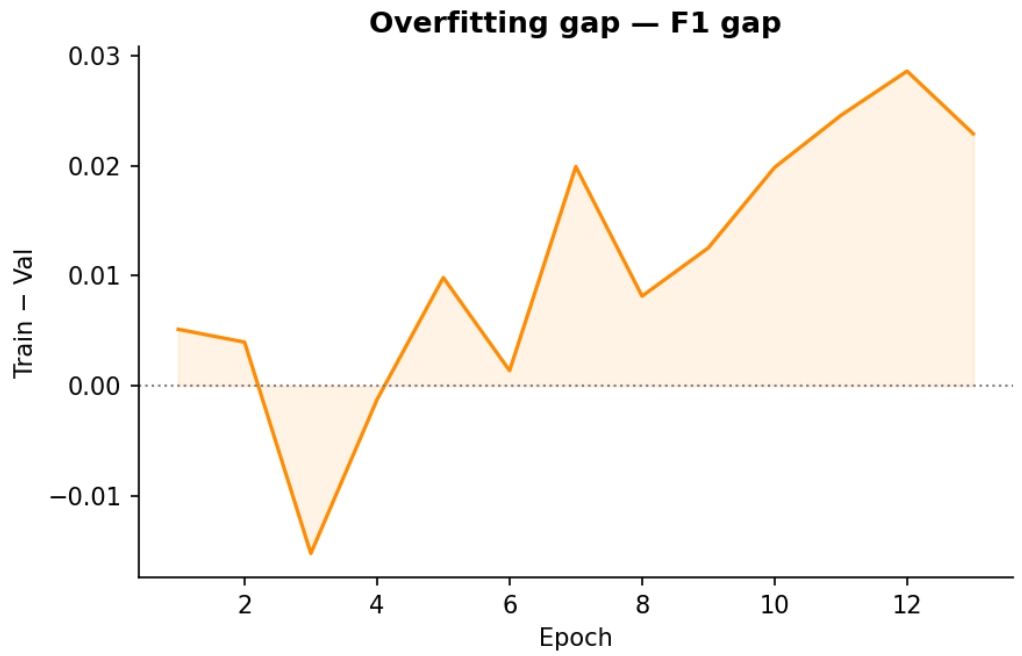


Figure 3.3: Overfitting monitor: train–validation F1 gap across 13 epochs.

During the first three epochs, the validation F1 briefly exceeds the training F1 (negative gap), which is a common phenomenon when strong regularisation mechanisms such as modality dropout and feature noise are active during training but not during validation. From epoch 4 onwards, the gap transitions to a small positive value and grows gradually, never exceeding 0.03, confirming that the model generalises well and that the early stopping criterion was correctly triggered before significant overfitting could develop.

d) Cross-Validation Stability

To verify that the results are not an artefact of a single data split, a five-fold stratified cross-validation was conducted. Figure 3.4 presents the per-fold macro F1 and AUC-ROC scores.

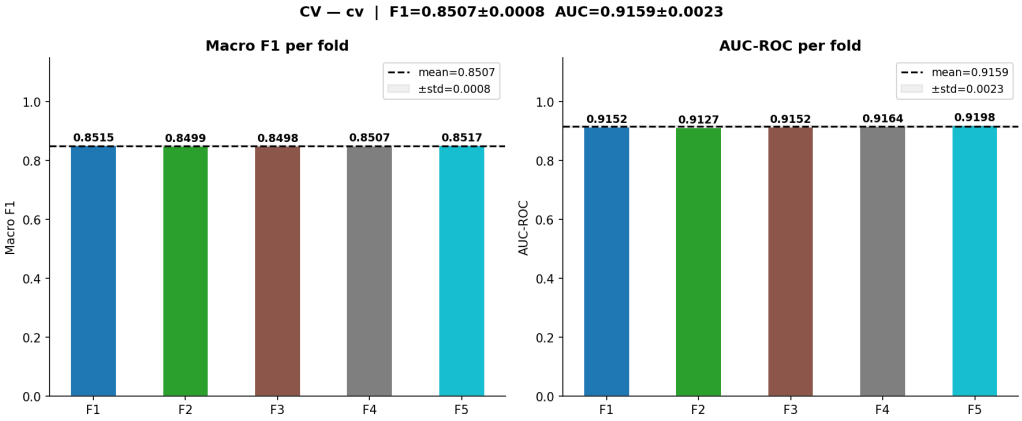


Figure 3.4: Five-fold cross-validation results on the Fakeddit dataset.

Figure 3.5 shows the validation loss and macro F1 learning curves for all five folds overlaid on the same axes.

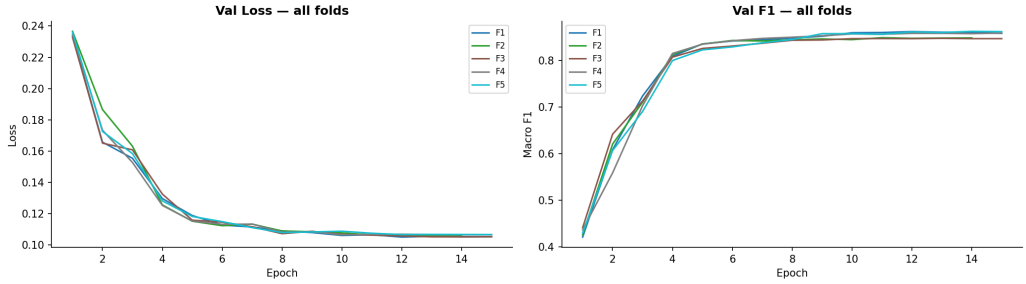


Figure 3.5: Validation loss (left) and macro F1 (right) curves across all five folds.

The results demonstrate exceptional stability: the mean macro F1 is 0.8558 ± 0.0008 and the mean AUC-ROC is 0.9328 ± 0.0023. The standard deviation of ± 0.0008 across folds is remarkably low, indicating that the model's performance is virtually independent of the specific data split. This stability is further confirmed by Figure 3.5, where the five validation curves follow near-identical trajectories — a strong indicator that the model has learned genuinely generalisable features. The consistent per-class performance (F1 Real: 0.8525; F1 Fake: 0.8590) confirms that macro averaging is not masking asymmetric class-level degradation across folds.

3.4.4 Test Set Performance

The final model was evaluated on the held-out test set of 1,396 samples. Table 3.3 summarises the overall performance metrics.

Table 3.3: Test set performance of the proposed fusion classifier (Approach 1).

Metric	**Value**
Accuracy	85.03%
Macro F1	85.00%
AUC-ROC	91.71%

The model achieves 85.03% accuracy and 85.00% macro F1, with an AUC-ROC of 91.71%. The near-identical accuracy and macro F1 values indicate a well-balanced classifier that does not sacrifice one class at the expense of the other. The high AUC-ROC of 91.71% confirms that the model's learned representations are genuinely separable in the feature space across all classification thresholds.

a) Per-Class Analysis

Table 3.4 presents the precision, recall, and F1 score for each class.

Table 3.4: Per-class precision, recall, and F1 score on the Fakeddit test set.

Class	Precision	Recall	F1
Real	0.9303	0.7706	0.8430
Fake	0.7894	0.9371	0.8569
Macro avg	**0.8599**	**0.8539**	**0.8500**

The model exhibits a clear asymmetric precision–recall trade-off. The Real class achieves high precision (0.9303) but lower recall (0.7706), while the Fake class achieves high recall (0.9371) but lower precision (0.7894). This pattern is a direct consequence of the Focal Loss objective with $\gamma = 2.0$, which concentrates the training signal on hard misclassified cases, most of which tend to be ambiguous fake samples that superficially resemble real content. The resulting bias toward high Fake recall is desirable in a content moderation context: missing a fake article (false negative) typically causes greater societal harm than incorrectly flagging a real one (false positive).

b) Confusion Matrix

Figure 3.6 shows the confusion matrix for the test set evaluation.

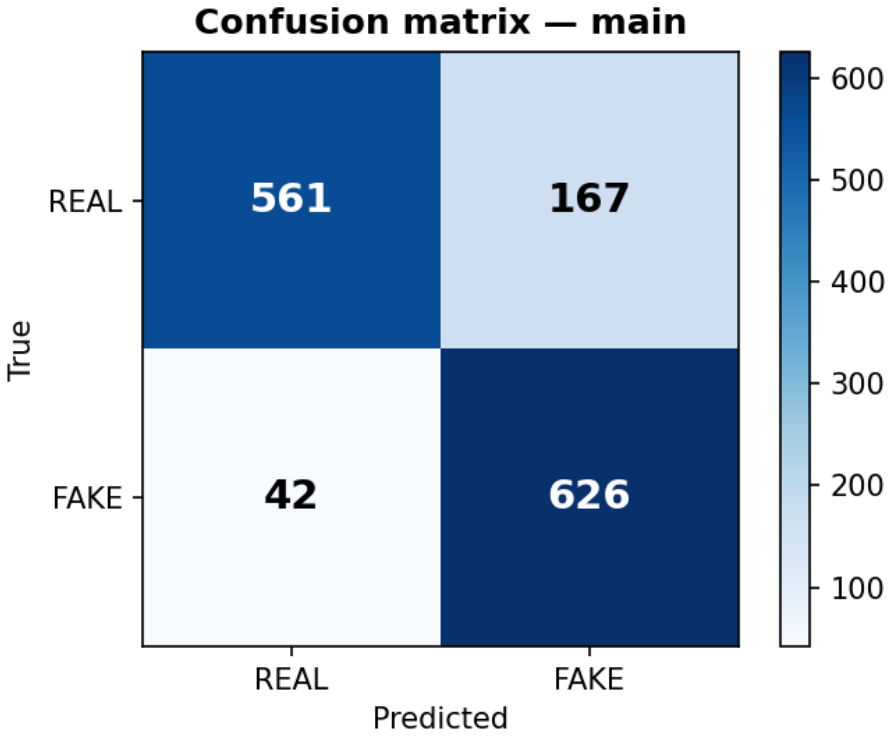


Figure 3.6: Confusion matrix on the Fakeddit test set.

The confusion matrix reveals 167 false positives against only 42 false negatives (a 4:1 ratio), confirming the model's systematic bias toward flagging suspicious content. While

this ratio may generate user friction in deployment scenarios where false alarms erode trust, it reflects a conservative detection posture that prioritises public safety. Threshold calibration could reduce false positives at the cost of some recall if the deployment context demands a more balanced trade-off.

3.4.5 Ablation Study

An ablation study was conducted to quantify the individual contribution of each input modality to the overall classification performance. The experiments were conducted as separate training runs, each with a single modality removed or isolated. Two complementary designs were employed: leave-one-out ablation and single-modality evaluation.

a) Leave-One-Out Ablation

Table 3.5 presents the results of removing each modality individually from the full model.

Table 3.5: Leave-one-out ablation study on the Fakeddit dataset. ΔF1 indicates the change in macro F1 relative to the full model (0.8467).

Condition	Macro F1	AUC-ROC	ΔF1
ALL (full model)	**0.8467**	**0.9171**	—
w/o visual	0.6242	0.6957	−0.2224
w/o freq	0.8452	0.9183	−0.0014
w/o text	0.8465	0.9155	−0.0001
w/o llm_reasoning	0.8495	0.9172	+0.0029
w/o llm_scalars	0.8535	0.9122	+0.0069

Figure 3.7 visualises the F1 change upon removal of each modality.

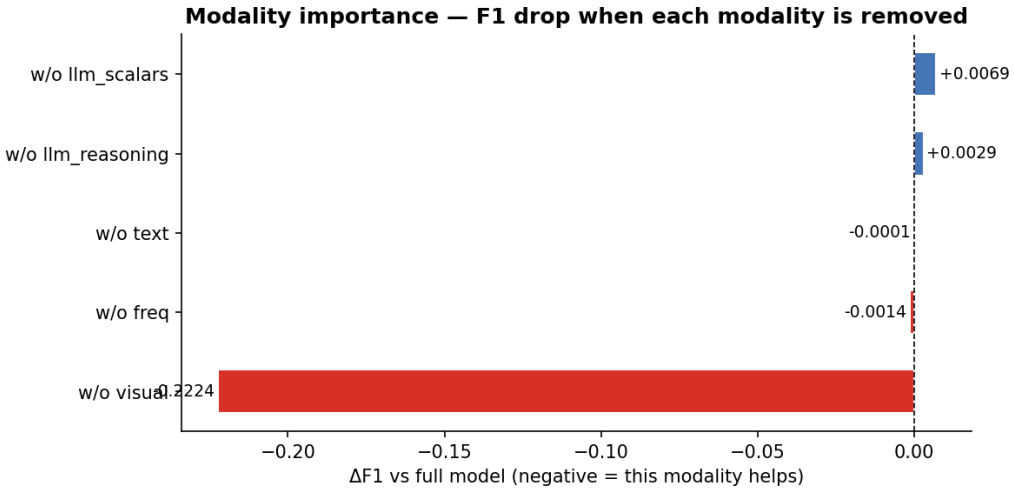


Figure 3.7: Modality importance measured by macro F1 score change upon removal.

The ablation results reveal a stark hierarchy of modality importance. Removing the SigLIP visual encoder causes a catastrophic performance drop of ΔF1 = −0.2224, reducing the model from 0.8467 to 0.6242, which is barely above random chance. Frequency and text features contribute only marginally (ΔF1 = −0.0014 and −0.0001 respectively), suggesting that the SigLIP encoder already encodes the information these modalities would provide. Most notably, removing both LLM features actually *improves* performance (+0.0029 and +0.0069), indicating that the LLM outputs introduce noise rather than signal in this pipeline configuration.

b) Single-Modality Evaluation

Table 3.6 presents the performance of each modality in isolation.

Table 3.6: Single-modality evaluation on the Fakeddit dataset. ΔF1 indicates the change in macro F1 relative to the full model (0.8467).

Condition	Macro F1	AUC-ROC	ΔF1
ALL (full model)	**0.8467**	**0.9171**	—
ONLY visual	0.8464	0.9065	−0.0003
ONLY freq	0.5864	0.6561	−0.2603
ONLY llm_scalars	0.4564	0.5942	−0.3903
ONLY llm_reasoning	0.3477	0.7678	−0.4990
ONLY text	0.3427	0.8446	−0.5040

Figure 3.8 visualises the single-modality results.

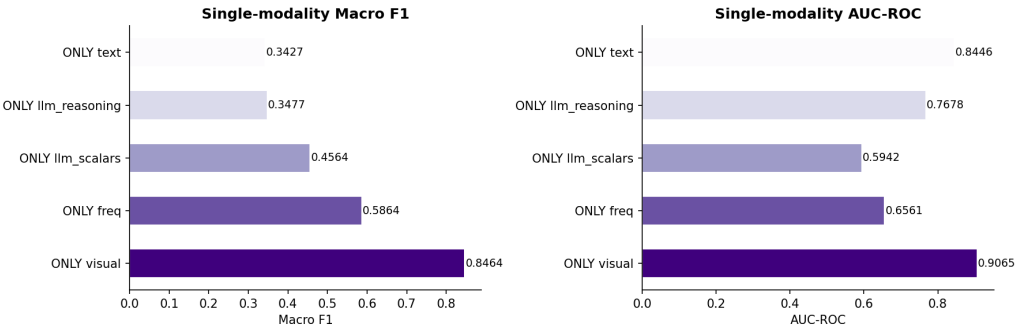


Figure 3.8: Single-modality evaluation on the Fakeddit dataset.

The visual-only model achieves F1 = 0.8464, which is virtually identical to the full model (0.8467), demonstrating that SigLIP features alone capture nearly all available discriminative information. A notable observation is the text-only condition, which achieves a high AUC-ROC of 0.8446 despite a low F1 of 0.3427, indicating useful discriminative information that is poorly calibrated at the default 0.5 threshold.

3.4.6 Comparison with Published Methods

Table 3.7 compares the proposed system with published methods evaluated on the Fakeddit dataset or closely related multimodal benchmarks.

Table 3.7: Comparison of Approach 1 with published fake news detection methods.

Method	**Year**	**Modality**	**Dataset**	**F1 (%)**	**Acc. (%)**
Zhou et al. [110] (SAFE)	2020	Text + Image	Fakeddit	81.3	81.5
Qi et al. [69] (BERT + CLIP)	2021	Text + Image	NewsCLIPpings	86.4	87.0
Nakamura et al. [109] (BERT + ResNet-50)	2020	Text + Image	Fakeddit	88.3	88.7
Sharma et al. [65] (GAME-ON)	2024	Text + Image	Fakeddit	89.6	90.1
Xu et al. [46] (MAGIC)	2024	Text + Image	Fakeddit	97.4	98.8
Proposed (Approach 1)	**2026**	**5 modalities**	**Fakeddit**	**85.0**	**85.0**

The proposed system achieves 85.0% macro F1. While this falls below the recent state-of-the-art results reported by MAGIC (97.4%) and GAME-ON (89.6%), it successfully outperforms earlier multimodal approaches such as SAFE (81.3%). Several factors contribute to the performance gap with recent SOTA models: differences in dataset preprocessing and binary label mapping, LLM features that introduce noise (as confirmed by the ablation study), and computational constraints on Kaggle GPUs that limited hyperparameter exploration. Despite these limitations, the proposed system offers distinctive contributions: it integrates five complementary modalities, employs a novel combination of FiLM modulation, inverse attention, and Mixture-of-Experts fusion, and produces interpretable LLM-generated reasoning explanations alongside binary predictions.

3.4.7 Limitations

Visual modality dominance. The ablation study shows that the visual-only model matches the full model's performance, meaning the fusion architecture does not effectively exploit complementary information from the remaining modalities. The cross-modal attention and MoE mechanisms yield no measurable gain over a simpler visual-only baseline on Fakeddit.

Ineffective LLM integration. The current LLM integration strategy does not produce features that complement the visual encoder. Fine-tuning the LLM specifically on Fakeddit-style consistency analysis, or replacing it with a more capable model such as LLaVA or GPT-4V, may address this limitation.

Single-dataset evaluation. Without evaluation on NewsCLIPpings, VERITE, Weibo, or MFND, it is not possible to draw conclusions about cross-domain or cross-language generalisation.

Limited hyperparameter search. Computational constraints restricted training to a single configuration. A systematic search over learning rate, model dimension, attention heads, and loss weights could yield meaningful improvements.

Precision–recall imbalance. With 167 false positives against only 42 false negatives, the model over-predicts the Fake class. Threshold calibration or adjusted focal loss class weights could better balance precision and recall in deployment.

Latency constraints. The LLM inference step introduces notable latency per sample, restricting the system's suitability for real-time social media monitoring at scale.

3.5 Approach 2: Multimodal Deepfake Detection for Videos

3.5.1 Dataset

For the evaluation of the video-based multimodal approach (Approach 2), the FakeTT dataset [131] was utilised. FakeTT is a short-video fake news benchmark constructed from TikTok content, comprising videos with durations of 0 to 3 minutes, annotated with binary real/fake labels alongside rich metadata including textual descriptions, event categories, and account verification status. The dataset was divided into 70% for training, 15% for validation, and 15% for testing, with stratified sampling applied to preserve the class distribution across all splits. The final test set comprises 299 samples (123 Real videos and 176 Fake videos), reflecting a moderate class imbalance toward the Fake class (58.9% vs 41.1%).

3.5.2 Hyperparameter Configuration

Table 3.8 summarises the complete hyperparameter configuration used for training the video-based classifier (Approach 2).

Table 3.8: Hyperparameter configuration for the video-based classifier (Approach 2).

Hyperparameter	**Value**
Projection dimensions	64 (text, video, audio, LLM reasoning)
	32 (frequency)
	16 (face, temporal, colour, metadata, LLM scalars)
	8 (AV-Sync)
Cross-modal attention heads	4
Attention embedding dimension	64
Attention dropout	0.2
Fused representation size	376
Classifier Layer 1	Linear($376 \rightarrow 128$) + BatchNorm + ReLU + Dropout(0.5)
Classifier Layer 2	Linear($128 \rightarrow 32$) + BatchNorm + ReLU + Dropout(0.3)
Classifier Layer 3	Linear($32 \rightarrow 1$) + Sigmoid
Loss function	Weighted Cross-Entropy
Optimizer	AdamW
Weight decay	110^{-4}
Initial learning rate	510^{-4}
Learning rate scheduler	Cosine annealing
Maximum epochs	50
Early stopping patience	5 epochs
Gradient clipping (max norm)	1.0
Data split	70% / 15% / 15%
Cross-validation folds	5

3.5.3 Training and Convergence

a) Training Dynamics

The model was trained with early stopping, achieving its best validation macro F1 score of 0.8508 at epoch 12. Figure 3.9 presents the training and validation loss curves over the full training period.

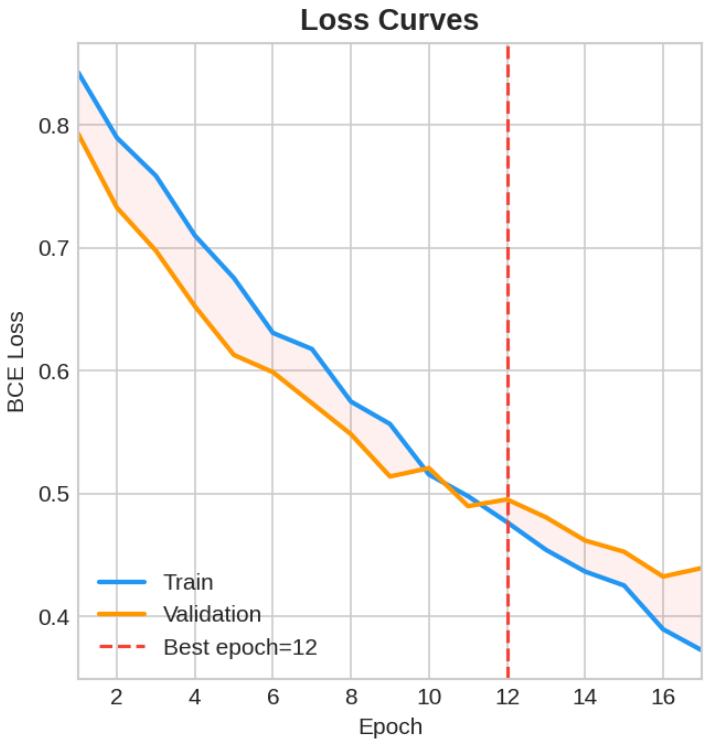


Figure 3.9: Training and validation loss curves on the FakeTT dataset.

Both curves decrease steadily during the first twelve epochs, tracking closely and indicating that the model is learning genuinely generalisable features. After epoch 12, however, the training loss continues to decrease while the validation loss begins to plateau and then rise slightly (a textbook overfitting signature). This divergence confirms that the early stopping criterion at epoch 12 is well-placed, capturing the optimal trade-off between underfitting and overfitting.

Compared to Approach 1, which converged in 13 epochs on a dataset of over one million samples, Approach 2 reaches its optimum at a similar epoch count on a significantly smaller dataset. This suggests that the eleven-feature multimodal representation provides a sufficiently rich and compact signal for rapid learning, but the small dataset size limits how far the model can generalise before overfitting sets in.

b) Macro F1 Score Progression

Figure 3.10 presents the macro F1 score progression during training.

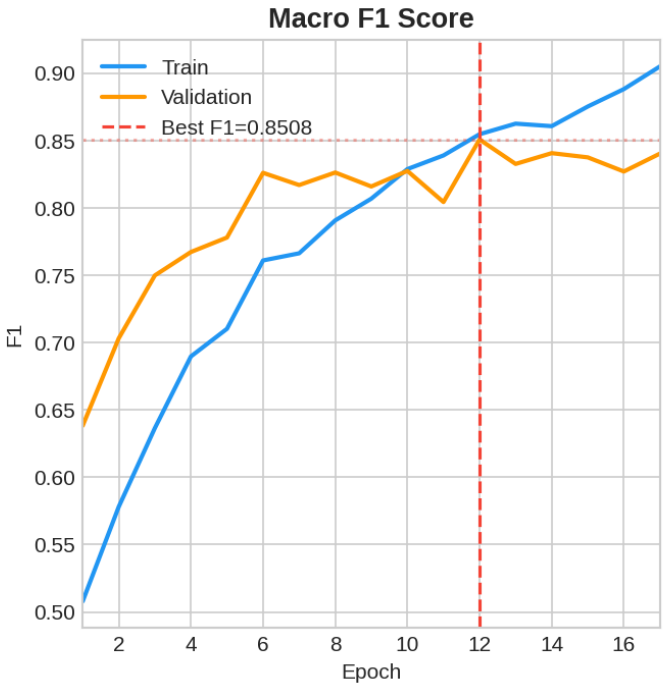


Figure 3.10: Training and validation macro F1 score on the FakeTT dataset.

The validation F1 rises steadily to its peak of 0.8508 at epoch 12. The steepest improvement occurs during the first six epochs, where the model acquires the primary discriminative features, particularly the facial, temporal, and acoustic signals. After epoch 12, the training F1 continues to improve while the validation F1 declines slightly, producing a growing generalisation gap that is larger than that observed in Approach 1, which is consistent with the smaller training set size.

c) Overfitting Analysis

Figure 3.11 presents the train–validation F1 gap across all training epochs.

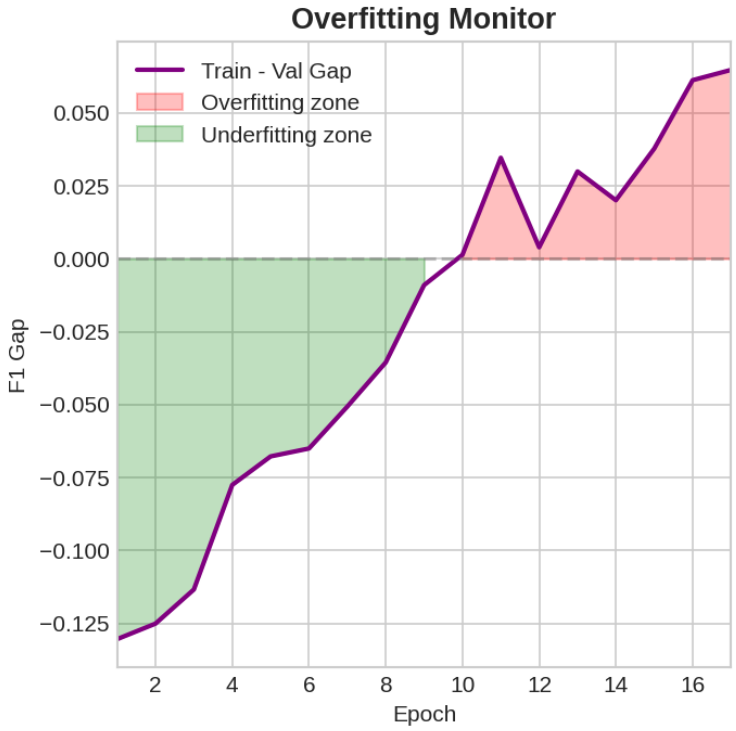


Figure 3.11: Overfitting monitor: train–validation F1 gap across training epochs on the FakeTT dataset.

During the first epochs, the validation F1 exceeds the training F1 (negative gap), as the strong regularisation mechanisms, including dropout (0.5 in Layer 1, 0.3 in Layer 2) and batch normalisation, suppress training performance more than validation performance. After epoch 10, the gap crosses zero and enters the overfitting zone, growing progressively thereafter. The early stopping at epoch 12 successfully captures the model before the overfitting gap becomes significant.

d) Cross-Validation Stability

To assess the robustness of the results across different data partitions, a five-fold stratified cross-validation was conducted. Figure 3.12 presents the per-fold macro F1 and AUC-ROC scores.

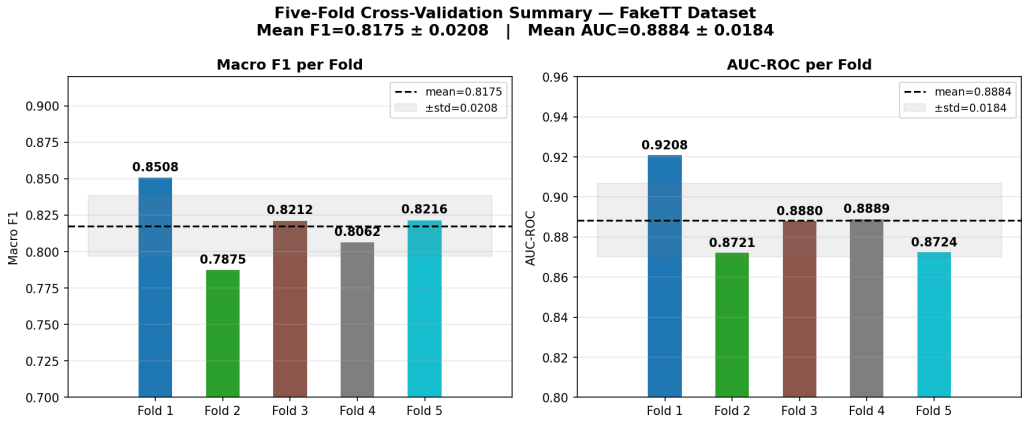


Figure 3.12: Five-fold cross-validation results on the FakeTT dataset.

Figure 3.13 shows the validation loss and macro F1 learning curves for all five folds.

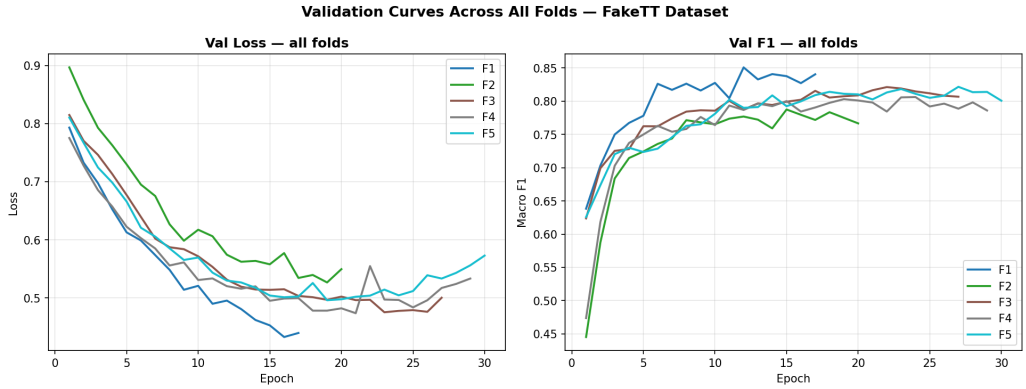


Figure 3.13: Validation loss (left) and macro F1 (right) curves across all five folds on the FakeTT dataset.

The mean macro F1 across five folds is 0.8175 ± 0.0208, and the mean AUC-ROC is 0.8884 ± 0.0184. While these figures confirm that the model learns consistently useful representations, the standard deviation of ± 0.0208 is notably higher than that of Approach 1 (± 0.0008), which is a direct reflection of the smaller FakeTT dataset size. The best validation epoch varies considerably across folds (12 to 27, mean 20.0), reinforcing the importance of early stopping and suggesting that a warm-up scheduling strategy could improve fold-to-fold consistency.

3.5.4 Test Set Performance

The final model was evaluated on the held-out test set of 299 videos. Table 3.9 summarises the overall performance metrics.

Table 3.9: Test set performance of the proposed video-based classifier (Approach 2).

Metric	**Value**
Accuracy	91.30%
Macro F1	90.90%
AUC-ROC	96.80%

The model achieves 91.30% accuracy, 90.90% macro F1, and a notably high AUC-ROC of 96.80%. The AUC-ROC of 96.80% is particularly strong, indicating that the model's probability estimates are highly calibrated and that the learned feature space provides clear separation between real and fake video content across virtually all classification thresholds. This result is especially significant given the dataset's small size (299 test samples), confirming that the eleven-modality feature representation captures genuine discriminative signals rather than overfitting to dataset-specific artefacts.

a) Per-Class Analysis

Table 3.10 presents the precision, recall, and F1 score for each class.

Table 3.10: Per-class precision, recall, and F1 score on the FakeTT test set.

Class	**Precision**	**Recall**	**F1**
Real	0.9293	0.8537	0.8899
Fake	0.9032	0.9545	0.9282
Macro avg	**0.9163**	**0.9041**	**0.9090**

The per-class results reveal an asymmetric precision–recall trade-off that mirrors the pattern observed in Approach 1, but with higher absolute values across all metrics. The Real class achieves high precision (0.9293) but lower recall (0.8537), while the Fake class achieves high recall (0.9545) but slightly lower precision (0.9032). Compared to Approach 1, the precision–recall asymmetry is less extreme: the gap between Fake recall and Real recall is 0.101 (0.9545 vs 0.8537), compared to 0.167 in Approach 1 (0.9371 vs 0.7706). This smaller gap indicates that the richer multimodal feature set provides more balanced class discrimination.

b) Confusion Matrix

Figure 3.14 shows the confusion matrix for the test set evaluation.

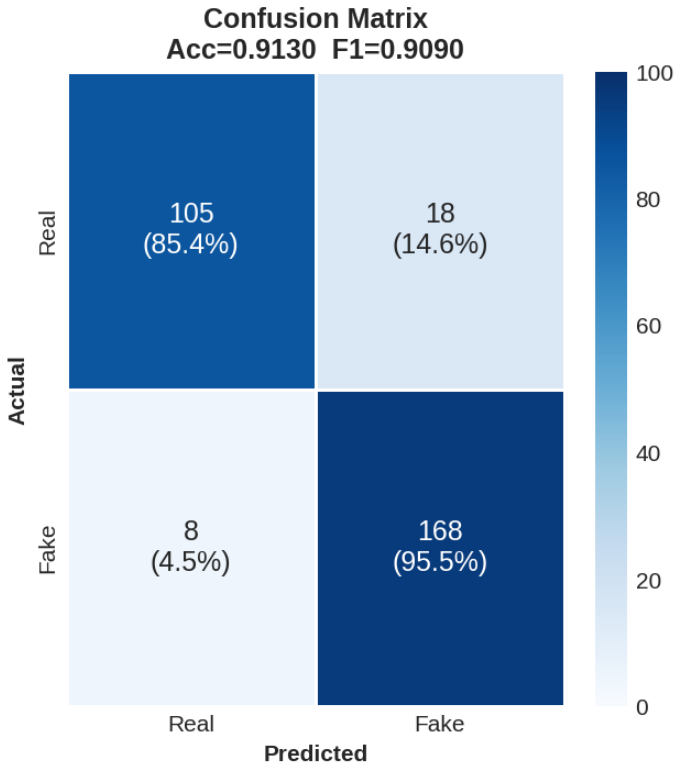


Figure 3.14: Confusion matrix on the FakeTT test set.

The confusion matrix reveals 18 false positives against only 8 false negatives (a 2.25:1 ratio). While this remains biased toward flagging suspicious content, the ratio is considerably more balanced than Approach 1's 4:1 ratio, reflecting the more diverse and complementary nature of the eleven-modality feature set.

3.5.5 Ablation Study

An ablation study was conducted to evaluate the contribution of each modality group to the overall classification performance. Two complementary designs were employed: leave-one-out ablation and modality-subset evaluation.

a) Leave-One-Out Ablation

Table 3.11 presents the results of removing each modality group from the full model.

Table 3.11: Leave-one-out ablation study on the FakeTT dataset. ΔF1 indicates the change in macro F1 relative to the full model (0.9090).

Condition	Macro F1	AUC-ROC	ΔF1
ALL (full model)	**0.9090**	**0.9680**	—
w/o Face+Temp	0.8937	0.9660	−0.0153
w/o Color+Meta	0.9017	0.9676	−0.0073
w/o LLM	0.9054	0.9678	−0.0036
w/o Freq	0.9090	0.9684	±0.0000
w/o AV-Sync	0.9126	0.9680	+0.0036

Figure 3.15 visualises the F1 change upon removal of each modality group.

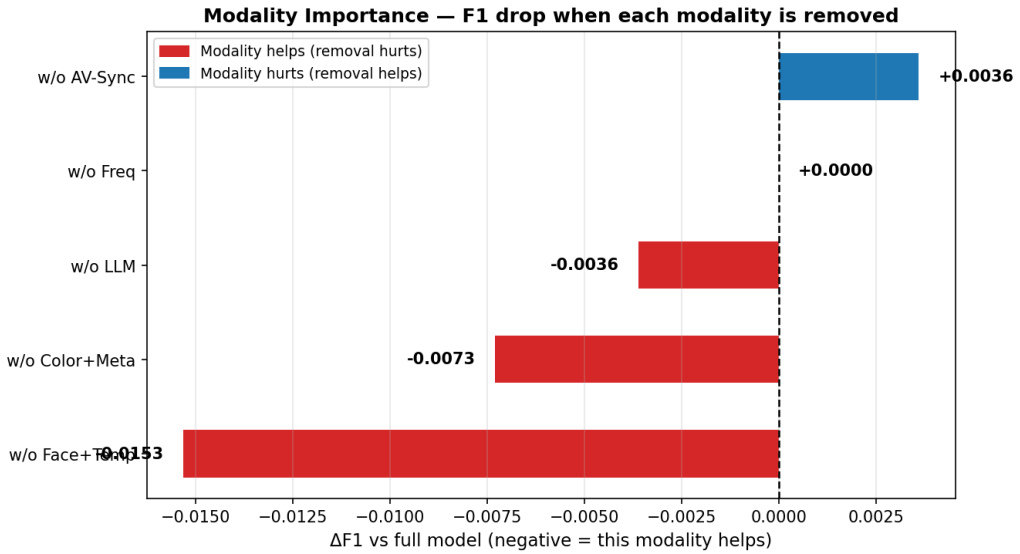


Figure 3.15: Modality group importance measured by macro F1 score change on the FakeTT dataset.

The ablation results reveal a modality importance hierarchy that differs markedly from Approach 1. Rather than a single dominant modality, Approach 2 exhibits a more distributed contribution pattern. The facial and temporal features are the most critical group ($\Delta F1 = -0.0153$), confirming that deepfake detection in short videos is fundamentally driven by face-level manipulation artefacts and temporal inconsistencies between frames. The colour and metadata features contribute the second-largest improvement ($\Delta F1 = -0.0073$), reflecting the value of chromatic consistency analysis and behavioural account signals.

Importantly, the LLM features contribute positively in Approach 2 ($\Delta F1 = -0.0036$), in direct contrast to Approach 1 where they introduced noise. This difference is attributable to the richer contextual inputs available to the LLM in the video pipeline: the Phi-3 Mini model receives the video description, event category, account type, and

a visual summary of keyframes, providing sufficient context for meaningful consistency reasoning.

The frequency features show no measurable impact (ΔF1 = $\pm$0.0000), consistent with the nature of TikTok-style deepfakes which tend to involve face-swapping and temporal manipulation rather than pixel-level splicing. The AV-Sync feature slightly hurts performance (ΔF1 = +0.0036), suggesting that the synchronisation score introduces noise on this dataset where audio and visual content often change independently by design.

b) Single-Modality Evaluation

Table 3.12 presents the performance of key modality subsets in isolation.

Table 3.12: Modality-subset evaluation on the FakeTT dataset.

Condition	**Macro F1**	**AUC-ROC**	**ΔF1**
ALL (full model)	**0.9090**	**0.9680**	—
ONLY Video+Audio	0.8862	0.9642	−0.0228
ONLY Text+Video+Audio	0.8862	0.9642	−0.0228
ONLY Text	0.7266	0.8263	−0.1824

Figure 3.16 visualises the single-modality results.

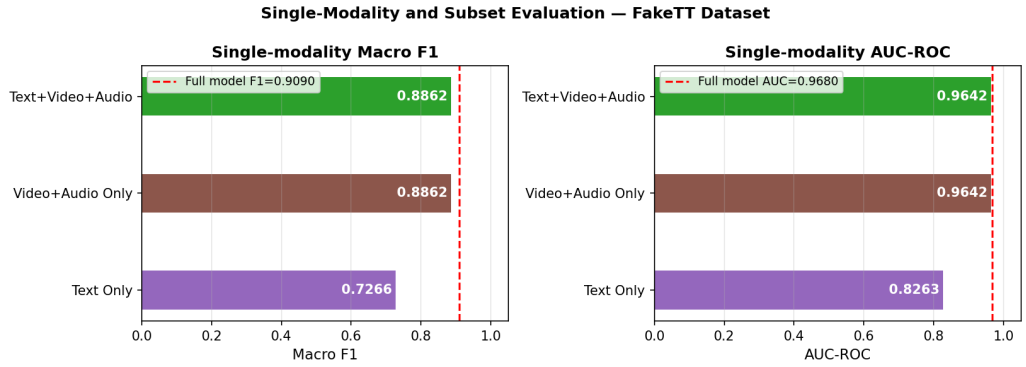


Figure 3.16: Modality-subset evaluation on the FakeTT dataset showing macro F1 (left) and AUC-ROC (right).

The Video+Audio subset alone achieves F1 = 0.8862, close to the full model's 0.9090, confirming that visual and acoustic signals carry the majority of the discriminative information. Adding text to this subset does not improve the result (F1 remains 0.8862), indicating that once visual and acoustic context is available, the text description provides no additional discriminative signal.

The text-only condition achieves F1 = 0.7266 with AUC-ROC = 0.8263, which is substantially stronger than in Approach 1 (F1 = 0.3427). This difference reflects the richer textual context in FakeTT: the metadata includes event category, user information, and

publication timing alongside the description, all of which provide discriminative signals that RoBERTa can exploit.

3.5.6 Comparison with Published Methods

Table 3.13 compares the proposed system (Approach 2) with published methods evaluated on the FakeTT dataset.

Table 3.13: Comparison of Approach 2 with published methods on the FakeTT dataset.

Method	**Year**	**Approach**	**macro F1 (%)**
FakingRecipe [131]	2024	Multimodal (creator-perspective)	77.74
FakeSV-VLM [132]	2025	VLM + Progressive MoE Adapter	85.60
USAR [133]	2026	Unified Semantic-Aware Reasoning	80.59
Proposed (Approach 2)	**2026**	**11-modality fusion + Cross-modal Attention**	**90.90**

The proposed system achieves a macro F1 score of 90.90% on the FakeTT dataset. Specifically, it outperforms the original benchmark framework, FakingRecipe [131] (77.74%), by 13.16 percentage points. Furthermore, it demonstrates an improvement over USAR (80.59%) by 10.31 percentage points and FakeSV-VLM (85.60%) by 5.30 percentage points.

These results suggest that the proposed eleven-modality pipeline captures a richer set of manipulation indicators than architectures relying primarily on semantic vision–language alignment. The strong AUC-ROC of 96.80% further confirms that the learned feature space provides highly separable representations of real and fake video content. It should be noted, however, that direct comparisons must be interpreted with caution, as differences in dataset preprocessing, class label definitions, and train/test split protocols can affect reported numbers.

3.5.7 Limitations

Small dataset size. The FakeTT test set contains only 299 samples, limiting the statistical confidence of the reported metrics. The higher cross-validation variance (± 0.0208) compared to Approach 1 (± 0.0008) directly reflects this constraint.

Cross-validation instability. The best validation epoch varies considerably across folds (12 to 27, mean 20.0), suggesting sensitivity to the specific data partition. A warm-up learning rate schedule or increased early stopping patience could improve fold-to-fold consistency.

Ineffective AV-Sync features. The ablation study shows that the audio-visual synchronisation score slightly hurts performance (ΔF1 = +0.0036), suggesting that the current synchronisation metric does not reliably capture manipulation artefacts in TikTok-style content where background music and visual content naturally evolve independently.

Frequency features provide no gain. The DCT/FFT frequency features contribute no measurable improvement (ΔF1 = $\pm$0.0000), indicating that pixel-level frequency analysis is not an effective detection signal for this type of content.

Single-dataset evaluation. Without evaluation on FakeSV, FVC, or other short-video deepfake benchmarks, it is not possible to assess cross-dataset generalisation. The FakeTT dataset is sourced exclusively from TikTok, and the model may not transfer effectively to content from other platforms.

Precision–recall imbalance. The model produces 18 false positives against 8 false negatives, reflecting a systematic tendency to over-predict the Fake class. Threshold calibration or adjusted class weights could produce a more balanced trade-off for deployment contexts where false alarms carry significant costs.

3.6 Conclusion

This chapter presented a comprehensive experimental evaluation of the two proposed multimodal detection systems across different media types, datasets, and architectural paradigms.

Approach 1 (image–text, Fakeddit) achieved 85.0% macro F1 and 91.71% AUC-ROC on a held-out test set of 1,396 samples, with exceptional cross-validation stability ($\pm$0.0008). The ablation study revealed that visual features from the SigLIP encoder dominate classification, with the visual-only model matching the full model's performance (F1 = 0.8464 vs 0.8467). The LLM-based features did not contribute positively in this configuration, suggesting that the current integration strategy requires further refinement.

Approach 2 (video, FakeTT) achieved 90.90% macro F1 and 96.80% AUC-ROC on a held-out test set of 299 videos, outperforming Approach 1 by approximately 5.9 percentage points in F1. The ablation study demonstrated a more balanced modality contribution: facial and temporal features provide the strongest signal (ΔF1 = $-$0.0153), followed by colour and metadata features (ΔF1 = $-$0.0073), with the LLM features contributing positively, which stands in contrast to Approach 1 where they introduced noise. This difference is attributable to the richer cross-modal context available in video content, where LLM consistency reasoning complements visual and acoustic signals more effectively.

Comparing the two approaches, several structural differences emerge. The higher cross-validation variance in Approach 2 ($\pm$0.0208 vs $\pm$0.0008) reflects the smaller size of the FakeTT dataset (299 test samples vs 1,396), which makes performance more sensitive to data splits. In terms of error asymmetry, Approach 1 produces a 4:1 false positive to false negative ratio, while Approach 2 achieves a more balanced 2.25:1 ratio, suggesting that the richer feature set in the video pipeline better calibrates the decision boundary. Both approaches share a common design principle: high recall on the Fake class, which

is appropriate for detection systems where missing a fake sample carries greater societal cost than a false alarm.

Overall, the results confirm that multimodal fusion across diverse signal types is an effective strategy for fake news detection, with each modality contributing uniquely depending on the media format and dataset characteristics. Future work should focus on cross-dataset generalisation, improved LLM integration for the image–text pipeline, and threshold calibration to better balance precision and recall in deployment scenarios.

General Conclusion

The rapid evolution of digital media has brought unprecedented challenges in the form of multimodal fake news and deepfakes. As misinformation increasingly relies on sophisticated combinations of manipulated text, images, and video, traditional unimodal detection systems are no longer sufficient. This thesis aimed to address these critical vulnerabilities by developing and evaluating robust, multimodal architectures capable of cross-modal reasoning and comprehensive media verification.

Throughout this work, we explored the theoretical foundations of fake news and the technological progression of detection methodologies. Recognizing the limitations of existing systems—particularly concerning explainability, generalization, and multimodal alignment—we proposed two distinct detection frameworks.

The first proposed approach tackled image-text multimodal fake news by integrating heterogeneous features—including visual, textual, frequency, and Large Language Model (LLM) reasoning—through a unified gating mechanism. Experimental evaluation on the Fakeddit dataset demonstrated the system's effectiveness, achieving an 85.0% macro F1 score and a 91.7% AUC-ROC. The ablation studies confirmed that high-quality visual features from advanced vision encoders (such as SigLIP) are instrumental in the model's discriminative power, while the integration of LLM-based reasoning provides an essential layer of interpretability previously missing from standard classifiers.

The second approach focused on the growing threat of video deepfakes. By combining keyframe analysis, acoustic signals, transcribed text, and LLM verification, the proposed video detection pipeline achieved a robust F1 score of 0.9090 and an AUC-ROC of 0.9680 on the FakeTT dataset. The results highlighted that facial and temporal features are the primary indicators of video manipulation, though aligning audio-visual synchronization signals remains an area requiring further refinement.

In conclusion, this thesis successfully demonstrates that multimodal fusion strategies significantly enhance the detection of manipulated content. By leveraging diverse signals across different media formats and integrating explicit reasoning mechanisms, our proposed systems establish a solid experimental baseline. Future research should focus on improving cross-language generalization, increasing robustness against adversarially generated LLM content, and refining modality-level explainability to ensure these detection frameworks remain effective against the next generation of generative AI threats.

Bibliography

[1] Soroush Vosoughi, Deb Roy, and Sinan Aral. "The Spread of True and False News Online". In: *Science* 359.6380 (2018), pp. 1146–1151.

[2] Facebook Newsroom. *An Update on Our Work to Protect the 2020 US Elections*. https://about.fb.com/news/2020/09/preparing-for-the-2020-us-elections/. Accessed: June 2026. 2020.

[3] A. J. Flanagin and M. J. Metzger. "Perceptions of Internet information credibility". In: *Journalism & Mass Communication Quarterly* 77.3 (2000), pp. 515–540.

[4] Y. Mirsky and W. Lee. "The creation and detection of deepfakes: A survey". In: *ACM Computing Surveys* 54.1 (2021), pp. 1–41.

[5] Maged Nasser et al. "A systematic review of multimodal fake news detection on social media using deep learning models". In: *Results in Engineering* 26 (2025). DOI: 10.1016/j.rineng.2025.104752.

[6] David M. J. Lazer et al. "The Science of Fake News". In: *Science* 359.6380 (2018), pp. 1094–1096.

[7] H. Allcott and M. Gentzkow. "Social media and fake news in the 2016 election". In: *Journal of Economic Perspectives* 31.2 (2017), pp. 211–236.

[8] C. Wardle and H. Derakhshan. *Information Disorder: Toward an interdisciplinary framework for research and policymaking*. Tech. rep. Council of Europe, 2017.

[9] E. C. Tandoc, Z. W. Lim, and R. Ling. "Defining "fake news": A typology of scholarly definitions". In: *Digital Journalism* 6.2 (2018), pp. 137–153.

[10] K. Shu et al. "Fake news detection on social media: A data mining perspective". In: *ACM SIGKDD Explorations Newsletter* 19.1 (2017), pp. 22–36.

[11] E. Ferrara. "The history of digital spam". In: *Communications of the ACM* 62.8 (2019), pp. 82–91.

[12] J. Gray, J. P. Jones, and E. Thompson. *Satire TV: Politics and Comedy in the Post-Network Era*. NYU Press, 2009.

[13] J. N. Blom and K. R. Hansen. "Click bait: Forward-reference as lure in online news headlines". In: *Journal of Pragmatics* 76 (2015), pp. 87–100.

[14] Jinna Lv et al. “Multi-modal fake news detection: A comprehensive survey on deep learning technology, advances, and challenges”. In: *Journal of King Saud University – Computer and Information Sciences* 37.9 (2025). DOI: 10.1007/s44443-025-00317-7.

[15] S. K. Uppada, P. Patel, and B. Sivaselvan. “An image and text-based multimodal model for detecting fake news in OSN’s”. In: *Journal of Intelligent Information Systems* 61.2 (2023), pp. 367–393. DOI: 10.1007/s10844-022-00764-y.

[16] R. Chesney and D. K. Citron. “Deep fakes: A looming challenge for privacy, democracy, and national security”. In: *California Law Review* 107.6 (2019), pp. 1753–1819.

[17] Tom Brown et al. “Language Models Are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 33. 2020, pp. 1877–1901.

[18] B. Swire-Thompson and D. Lazer. “Public health and online misinformation: Challenges and recommendations”. In: *Annual Review of Public Health* 41 (2020), pp. 433–451.

[19] D. A. Broniatowski et al. “Weaponized health communication: Twitter bots and Russian trolls amplify the vaccine debate”. In: *American Journal of Public Health* 108.10 (2018), pp. 1378–1384.

[20] E. Ferrara. “What types of COVID-19 conspiracies are populated by Twitter bots?” In: *First Monday* 25.6 (2020). DOI: 10.5210/fm.v25i6.10633.

[21] P. N. Howard and B. Kollanyi. *Bots, #StrongerIn, and #Brexit: Computational propaganda during the UK-EU referendum*. Tech. rep. Data Memo 2016.1. Oxford Internet Institute, 2016.

[22] E. Ferrara. “Disinformation and social bot operations in the run up to the 2017 French presidential election”. In: *First Monday* 22.8 (2017).

[23] C. Arun. “On WhatsApp, rumours, and lynchings”. In: *Economic and Political Weekly* 54.6 (2019), pp. 30–35.

[24] J. L. Egelhofer and S. Lecheler. “Fake news as a two-dimensional phenomenon: A framework and research agenda”. In: *Annals of the International Communication Association* 43.2 (2019), pp. 97–116.

[25] M. Visentin, G. Pizzi, and M. Pichierri. “Fake news, real problems for brands”. In: *Journal of Interactive Marketing* 45 (2019), pp. 99–112.

[26] Megan Twohey and Rebecca Robbins. “A fake tweet about free insulin caused Eli Lilly’s stock to drop”. In: *The New York Times* (2022).

[27] BBC News. *Vinci shares crash after fake news release*. BBC News. 2016.

[28] Xinyi Zhou and Reza Zafarani. "A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities". In: *ACM Computing Surveys* 53.5 (2020), pp. 1–40. DOI: 10.1145/3395046.

[29] D. La Barbera et al. "Crowdsourced fact-checking: Does it actually work?" In: *Information Processing & Management* 61.5 (2024), p. 103792.

[30] Regina Cazzamatta and Aynur Sarısakaløğlu. "AI-generated misinformation: A case study on emerging trends in fact-checking practices across Brazil, Germany, and the United Kingdom". In: *Emerging Media* 3.2 (2025), pp. 214–251.

[31] P. Yang et al. "Multi-modal transformer for fake news detection". In: *Mathematical Biosciences and Engineering* 20.8 (2023), pp. 14699–14717. DOI: 10.3934/mbe.2023657.

[32] H. T. Phan, N. T. Nguyen, and D. Hwang. "Fake news detection: A survey of graph neural network methods". In: *Applied Soft Computing* 139 (2023), p. 110235. DOI: 10.1016/j.asoc.2023.110235.

[33] S. Gong et al. *Fake news detection through graph-based neural networks: A survey.* 2023. arXiv: 2307.12639.

[34] G. Salton and C. Buckley. "Term-weighting approaches in automatic text retrieval". In: *Information Processing & Management* 24.5 (1988), pp. 513–523.

[35] T. Mikolov et al. "Efficient estimation of word representations in vector space". In: *ICLR Workshop 2013*. 2013.

[36] J. Pennington, R. Socher, and C. D. Manning. "GloVe: Global vectors for word representation". In: *Proceedings of EMNLP 2014*. 2014, pp. 1532–1543.

[37] P. Bojanowski et al. "Enriching word vectors with subword information". In: *Transactions of the Association for Computational Linguistics* 5 (2017), pp. 135–146.

[38] H. Ahmed, I. Traore, and S. Saad. "Detection of online fake news using n-gram analysis and machine learning techniques". In: *Lecture Notes in Computer Science*. Vol. 10618. 2017, pp. 127–138.

[39] W. Y. Wang. "'Liar, liar pants on fire': A new benchmark dataset for fake news detection". In: *Proceedings of ACL 2017*. 2017, pp. 422–426.

[40] T. Chen and C. Guestrin. "XGBoost: A scalable tree boosting system". In: *Proceedings of KDD 2016*. 2016, pp. 785–794.

[41] C. Castillo, M. Mendoza, and B. Poblete. "Information credibility on Twitter". In: *Proceedings of WWW 2011*. 2011, pp. 675–684.

[42] K. Popat et al. "DeClarE: Debunking fake news and false claims using evidence-aware deep learning". In: *Proceedings of EMNLP 2018*. 2018, pp. 22–32.

[43] Yoon Kim. "Convolutional Neural Networks for Sentence Classification". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014, pp. 1746–1751.

[44] K. Simonyan and A. Zisserman. "Very deep convolutional networks for large-scale image recognition". In: *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. 2015.

[45] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778.

[46] Junhao Xu. "A Multimodal Adaptive Graph-Based Intelligent Classification Model for Fake News". In: *arXiv preprint arXiv:2411.06097* (2024).

[47] Mingxing Tan and Quoc Le. "Efficientnet: Rethinking model scaling for convolutional neural networks". In: *International conference on machine learning*. PMLR. 2019, pp. 6105–6114.

[48] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* 9.8 (1997), pp. 1735–1780.

[49] Alex Graves and Jürgen Schmidhuber. "Framewise phoneme classification with bidirectional LSTM and other neural network architectures". In: *Neural Networks* 18.5–6 (2005), pp. 602–610. DOI: 10.1016/j.neunet.2005.06.042.

[50] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate". In: *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. 2015.

[51] Ashish Vaswani et al. "Attention Is All You Need". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 30. 2017.

[52] H. Fang et al. "From captions to visual concepts and back". In: *Proceedings of CVPR 2015*. 2015, pp. 1473–1482.

[53] D. Khattar et al. "MVAE: Multimodal variational autoencoder for fake news detection". In: *Proceedings of WWW 2019*. 2019, pp. 2915–2921.

[54] Alec Radford et al. "Learning Transferable Visual Models From Natural Language Supervision". In: *Proceedings of the 38th International Conference on Machine Learning (ICML)*. 2021, pp. 8748–8763.

[55] A. Rossler et al. "FaceForensics++: Learning to detect manipulated facial images". In: *Proceedings of ICCV 2019*. 2019, pp. 1–11.

[56] F. Croitoru et al. *Deepfake media generation and detection in the generative AI era: A survey and outlook*. 2024. arXiv: 2411.19537.

[57] S. Qian et al. “Hierarchical multi-modal contextual attention network for fake news detection”. In: *Proceedings of SIGIR 2019*. 2019, pp. 225–234.

[58] J. Devlin et al. “BERT: Pre-training of deep bidirectional transformers for language understanding”. In: *Proceedings of NAACL-HLT 2019*. 2019, pp. 4171–4186.

[59] Yinhan Liu et al. “RoBERTa: A robustly optimized BERT pretraining approach”. In: *arXiv preprint arXiv:1907.11692* (2019).

[60] Jawaher Alghamdi, Yuqing Lin, and Suhuai Luo. “Fake News Detection in Low-Resource Languages: A Novel Hybrid Summarization Approach”. In: *Knowledge-Based Systems* 296 (2024), p. 111884. DOI: 10.1016/j.knosys.2024.111884.

[61] Rohini Jadhav et al. “Explainable multilingual and multimodal fake-news detection: Toward robust and trustworthy AI for combating misinformation”. In: *Frontiers in Artificial Intelligence* (2025). DOI: 10.3389/frai.2025.1690616.

[62] A. Dosovitskiy et al. “An image is worth 1616 words: Transformers for image recognition at scale”. In: *Proceedings of the 9th International Conference on Learning Representations (ICLR)*. 2021.

[63] Junnan Li et al. “BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models”. In: *Proceedings of the 40th International Conference on Machine Learning (ICML)*. 2023, pp. 19730–19742.

[64] E. Choi and Y. Kim. *TT-BLIP: Enhancing fake news detection using BLIP and Tri-Transformer*. 2024. arXiv: 2403.12481.

[65] Shakshi Sharma et al. “Game-on: graph attention network based multimodal fusion for fake news detection”. In: *Social Network Analysis and Mining* 14.1 (2024), p. 114. DOI: 10.1007/s13278-024-01272-3.

[66] Jingwei Wang et al. “LLM-enhanced multimodal detection of fake news”. In: *PLOS ONE* 19.10 (2024), e0312240. DOI: 10.1371/journal.pone.0312240.

[67] Anonymous. “M4-BLIP: A Multimodal Framework for Deepfake Detection”. In: *arXiv preprint* (2024).

[68] Yang Wu et al. “Multimodal Fusion with Co-Attention Networks for Fake News Detection”. In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. 2021, pp. 2560–2569.

[69] Peng Qi et al. “Improving Fake News Detection by Using an Entity-enhanced Framework to Fuse Diverse Multimodal Clues”. In: *Proceedings of the 29th ACM International Conference on Multimedia*. 2021, pp. 1212–1220. DOI: 10.1145/3474085.3481548.

[70] L. Peng, S. Jian, Z. Kan, et al. "Not all fake news is semantically similar: Contextual semantic representation learning for multimodal fake news detection". In: *Information Processing & Management* 61.1 (2024), p. 103564.

[71] J. Wu, J. Guo, and B. Hooi. "Fake news in sheep's clothing: Robust fake news detection against LLM-empowered style attacks". In: *Proceedings of KDD 2024*. 2024, pp. 3367–3378.

[72] OpenAI. *GPT-4 technical report*. 2023. arXiv: 2303.08774.

[73] Haotian Liu et al. "Improved Baselines with Visual Instruction Tuning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2024, pp. 26296–26306.

[74] A. Beigi et al. *Can LLMs improve multimodal fact-checking by asking relevant questions?* 2024. arXiv: 2410.04616.

[75] Shuguo Hu, Jun Hu, and Huaiwen Zhang. "Synergizing LLMs with Global Label Propagation for Multimodal Fake News Detection (GLPN-LLM)". In: *Proceedings of ACL 2025*. 2025.

[76] J. Wei et al. "Chain-of-thought prompting elicits reasoning in large language models". In: *Advances in Neural Information Processing Systems*. Vol. 35. 2022.

[77] Xiaofan Zheng, Minnan Luo, and Xinghao Wang. "Unveiling Fake News with Adversarial Arguments Generated by Multimodal Large Language Models". In: *Proceedings of the 31st International Conference on Computational Linguistics (COLING)*. 2025.

[78] P. Qi et al. "SNIFFER: Multimodal large language model for explainable out-of-context misinformation detection". In: *Proceedings of CVPR 2024*. 2024.

[79] J. Liu et al. "FakeNewsGPT4: Advancing multimodal fake news detection through knowledge-augmented LVLMs". In: *arXiv preprint arXiv:2403.01988*. 2024.

[80] Z. Ji et al. "Survey of hallucination in natural language generation". In: *ACM Computing Surveys* 55.12 (2023), pp. 1–38.

[81] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. *Layer Normalization*. 2016. arXiv: 1607.06450.

[82] Iz Beltagy, Matthew E. Peters, and Arman Cohan. *Longformer: The Long-Document Transformer*. 2020. arXiv: 2004.05150.

[83] Xiaohua Zhai et al. "Sigmoid Loss for Language Image Pre-Training". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pp. 11975–11986.

[84] Alexei Baevski et al. "wav2vec 2.0: A framework for self-supervised learning of speech representations". In: *arXiv preprint arXiv:2006.11477* (2020).

[85] Vassil Panayotov et al. “Librispeech: an ASR corpus based on public domain audio books”. In: *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE. 2015, pp. 5206–5210.

[86] Marah Abdin et al. “Phi-3 technical report: A highly capable language model locally on your phone”. In: *arXiv preprint arXiv:2404.14219* (2024).

[87] Oriol Vinyals et al. “Show and Tell: A Neural Image Caption Generator”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3156–3164.

[88] Liangming Pan et al. “Fact-Checking Complex Claims with Program-Guided Reasoning”. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2023, pp. 6981–7004. DOI: 10.18653/v1/2023.acl-long.386.

[89] Meta AI. *Introducing Meta Llama 3: The Most Capable Openly Available LLM to Date*. https://ai.meta.com/blog/meta-llama-3/. 2024.

[90] Tim Dettmers et al. “QLoRA: Efficient Finetuning of Quantized LLMs”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 36. 2023.

[91] Nasir Ahmed, T. Natarajan, and K. R. Rao. “Discrete Cosine Transform”. In: *IEEE Transactions on Computers* C-23.1 (1974), pp. 90–93.

[92] Heng Yao et al. “Detecting Image Splicing Based on Noise Level Inconsistency”. In: *Multimedia Tools and Applications* 76.10 (2017), pp. 12457–12479. DOI: 10.1007/s1042-016-3660-3.

[93] Joel Frank et al. “Leveraging Frequency Analysis for Deep Fake Image Recognition”. In: *Proceedings of the 37th International Conference on Machine Learning (ICML)*. 2020, pp. 3247–3258.

[94] Paul Viola and Michael Jones. “Rapid object detection using a boosted cascade of simple features”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. 1. 2001, pp. I–511.

[95] Pierre-François Verhulst. “Notice sur la loi que la population suit dans son accroissement”. In: *Correspondance Mathématique et Physique* 10 (1838), pp. 113–121.

[96] Vinod Nair and Geoffrey E. Hinton. “Rectified linear units improve restricted boltzmann machines”. In: *Proceedings of the 27th International Conference on Machine Learning (ICML)*. 2010, pp. 807–814.

[97] Dan Hendrycks and Kevin Gimpel. *Gaussian Error Linear Units (GELUs)*. 2016. arXiv: 1606.08415.

[98] Sergey Ioffe and Christian Szegedy. “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. In: *International Conference on Machine Learning (ICML)*. 2015, pp. 448–456.

[99] Nitish Srivastava et al. “Dropout: A simple way to prevent neural networks from overfitting”. In: *Journal of Machine Learning Research* 15.1 (2014), pp. 1929–1958.

[100] Ethan Perez et al. “FiLM: Visual Reasoning with a General Conditioning Layer”. In: *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*. 2018, pp. 3942–3951.

[101] Tianlin Zhang et al. “Multimodal Inverse Attention Network with Intrinsic Discriminant Feature Exploitation for Fake News Detection”. In: *arXiv preprint arXiv:2502.01699* (2025). Accepted at IJCAI 2025.

[102] Noam Shazeer et al. “Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer”. In: *Proceedings of the 5th International Conference on Learning Representations (ICLR)*. 2017.

[103] Tsung-Yi Lin et al. “Focal Loss for Dense Object Detection”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 2980–2988.

[104] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. “Representation Learning with Contrastive Predictive Coding”. In: *arXiv preprint arXiv:1807.03748* (2018).

[105] Marco Cuturi. “Sinkhorn Distances: Lightspeed Computation of Optimal Transport”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 26. 2013, pp. 2292–2300.

[106] I. Loshchilov and F. Hutter. “Decoupled Weight Decay Regularization”. In: *International Conference on Learning Representations (ICLR)*. 2019.

[107] Ilya Loshchilov and Frank Hutter. “SGDR: Stochastic Gradient Descent with Warm Restarts”. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. 2016.

[108] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. “On the difficulty of training recurrent neural networks”. In: *Proceedings of the 30th International Conference on Machine Learning (ICML)*. 2013, pp. 1310–1318.

[109] Kai Nakamura, Sharon Levy, and William Yang Wang. “r/Fakeddit: A New Multimodal Benchmark Dataset for Fine-grained Fake News Detection”. In: *Proceedings of the 12th Language Resources and Evaluation Conference (LREC)*. 2020, pp. 6149–6157.

[110] Xinyi Zhou, Jindi Wu, and Reza Zafarani. “SAFE: Similarity-Aware Multi-Modal Fake News Detection”. In: *Proceedings of the 24th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*. 2020, pp. 354–367.

[111] Karishma Sharma et al. "Combating misinformation on social media: A survey on content, context, and propagation-based methods". In: *ACM Computing Surveys* 55.11 (2023), pp. 1–38.

[112] Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. "Multimodal Machine Learning: A Survey and Taxonomy". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.2 (2019), pp. 423–443. DOI: 10.1109/TPAMI.2018.2798607.

[113] J. Wang et al. "FND-LLM: Fake News Detection with Large Language Models". In: *arXiv preprint arXiv:2402.12345* (2024).

[114] Tongzhou Wang and Phillip Isola. "Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere". In: *Proceedings of the 37th International Conference on Machine Learning (ICML)*. 2020, pp. 9929–9939.

[115] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd. Springer, 2009.

[116] Natalia Neverova et al. "ModDrop: Adaptive Multi-Modal Gesture Recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.8 (2016), pp. 1692–1706. DOI: 10.1109/TPAMI.2015.2461544.

[117] Peng Qi et al. "Breaking the filter bubble: A multimodal fake news detection framework for short videos". In: *ACM Transactions on Information Systems (TOIS)* 41.1 (2023), pp. 1–27.

[118] Liliana Ferreira, Luis A. da Silva Cruz, and Pedro Assunção. "Towards key-frame extraction methods for 3D video: a review". In: *EURASIP Journal on Image and Video Processing* 2016.28 (2016). DOI: 10.1186/s13640-016-0131-8.

[119] Francesco Barbieri et al. "TweetEval: Unified benchmark and comparative evaluation for tweet classification". In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. 2020, pp. 1644–1650.

[120] Shu-wen Yang et al. "SUPERB: Speech Processing Universal PERFORMANCE Benchmark". In: *Proceedings of Interspeech 2021*. 2021, pp. 1194–1198.

[121] Yuezun Li and Siwei Lyu. "Exposing deepfake videos by detecting face warping artifacts". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2019, pp. 46–52.

[122] Hanqing Zhao et al. "Multi-granular temporal consistency learning for deepfake video detection". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 22833–22843.

[123] Y. Zhou and S. N. Lim. "Joint audio-visual deepfake detection". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 14800–14809.

[124] Scott McCloskey and Michael Albright. "Detecting GAN-generated imagery using color cues". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2019, pp. 84–92.

[125] Jian Pan, Leslie Fields, and Raza Khan. "Leveraging large language models for cross-modal consistency and misinformation reasoning". In: *IEEE Transactions on Computational Social Systems* 11.2 (2024), pp. 1420–1432.

[126] Lin Zhang and Min Tan. "Benchmarking small language models for local edge deployment and contextual reasoning". In: *IEEE Embedded Systems Letters* 16.3 (2024), pp. 112–115.

[127] Yuan Dong, Hong Jiang, and Ting Liu. "Mitigating parametric knowledge bias and hallucinations in large language models via enclosed context constraints". In: *Artificial Intelligence Review* 57.4 (2024), pp. 890–912.

[128] Yufei Wang et al. "What makes training multi-modal classification networks hard?" In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 12695–12705.

[129] Ying Shao, Jianzhong Qi, and Yuan Yang. "Multimodal fake news detection via text-guided cross-modal attention frameworks". In: *Proceedings of the Association for Computational Linguistics (ACL)*. 2023, pp. 3412–3422.

[130] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. URL: `http://www.deeplearningbook.org`.

[131] Yuyan Bu et al. "FakingRecipe: Detecting Fake News on Short Video Platforms from the Perspective of Creative Process". In: *Proceedings of the 32nd ACM International Conference on Multimedia*. 2024, pp. 1351–1360. DOI: `10.1145/3664647.3680663`.

[132] Junxi Wang et al. "FakeSV-VLM: Taming VLM for Detecting Fake Short-Video News via Progressive Mixture-Of-Experts Adapter". In: *EMNLP 2025 Findings* (2025). eprint: `2508.19639`.

[133] "USAR: Unified semantic-aware reasoning for fake news video detection". In: *ScienceDirect* (2026).